

ARTICLES FOR FACULTY MEMBERS

Marine Big Data

TITLE/ AUTHOR	Global spatial risk assessment of sharks under the footprint of fisheries / Queiroz, N., Humphries, N.E., Couto, A. et al
SOURCE	Nature Volume 572, Pages 461–466 (Aug 2019) https://doi.org/10.1038/s41586-019-1444-4 (Database: Nature)
TITLE/ AUTHOR	Marine biofilms constitute a bank of hidden microbial diversity and functional potential / Zhang, W., Ding, W., Li, YX. et al.
SOURCE	Nature Communication Volume 10, Article no. 517 (Jan 2019) https://doi.org/10.1038/s41467-019-08463-z (Database: Nature)
TITLE/ AUTHOR	Big Data Driven Marine Environment Information Forecasting: A Time Series Prediction Network / J. Wen, J. Yang, B. Jiang, H. Song and H. Wang
SOURCE	IEEE Transactions on Fuzzy Systems Volume 29, No. 1, Pages 4-18 (Jan. 2021) https://doi.org/10.1109/TFUZZ.2020.3012393 (Database: IEEE)
TITLE/ AUTHOR	Internet of Things in Marine Environment Monitoring: A Review. Sensors / Xu, G., Shi, Y., Sun, X., & Shen, W.
SOURCE	Sensors Volume 19, No. 7, Pages 1711 (April 2019) http://dx.doi.org/10.3390/s19071711 (Database: MDPI AG)

ARTICLES FOR FACULTY MEMBERS

Marine Big Data

TITLE/ AUTHOR	Internet of Underwater Things and Big Marine Data Analytics—A Comprehensive Survey / M. Jahanbakht, W. Xiang, L. Hanzo and M. Rahimi Azghadi
SOURCE	IEEE Communications Surveys & Tutorials Volume 23, No. 2, Pages 904-956, Secondquarter 2021 https://doi.org/10.1109/COMST.2021.3053118 (Database: ResearchGate)
TITLE/ AUTHOR	Privacy-Preserving Blockchain-Based Federated Learning for Marine Internet of Things / Z. Qin, J. Ye, J. Meng, B. Lu and L. Wang
SOURCE	IEEE Transactions on Computational Social Systems Volume 9, No. 1, Pages 159-173 (Feb. 2022) https://doi.org/10.1109/TCSS.2021.3100258 (Database: IEEE)

TITLE/ AUTHOR	Global spatial risk assessment of sharks under the footprint of fisheries / Queiroz, N., Humphries, N.E., Couto, A. et al.
SOURCE	Nature Volume 572, Pages 461–466 (Aug 2019) https://doi.org/10.1038/s41586-019-1444-4 (Database: Nature)

Global spatial risk assessment of sharks under the footprint of fisheries

A list of authors and their affiliations appears in the online version of the paper.

Effective ocean management and the conservation of highly migratory species depend on resolving the overlap between animal movements and distributions, and fishing effort. However, this information is lacking at a global scale. Here we show, using a big-data approach that combines satellite-tracked movements of pelagic sharks and global fishing fleets, that 24% of the mean monthly space used by sharks falls under the footprint of pelagic longline fisheries. Space-use hotspots of commercially valuable sharks and of internationally protected species had the highest overlap with longlines (up to 76% and 64%, respectively), and were also associated with significant increases in fishing effort. We conclude that pelagic sharks have limited spatial refuge from current levels of fishing effort in marine areas beyond national jurisdictions (the high seas). Our results demonstrate an urgent need for conservation and management measures at high-seas hotspots of shark space use, and highlight the potential of simultaneous satellite surveillance of megafauna and fishers as a tool for near-real-time, dynamic management.

Industrialized fishing is a major source of mortality for large marine animals (marine megafauna)^{1–6}. Humans have hunted megafauna in the open ocean for at least 42,000 years⁷, but international fishing fleets that target large, epipelagic fishes did not spread into the high seas until the 1950s⁸. Prior to this, the high seas constituted a spatial refuge that was largely free from exploitation, as fishing pressure was concentrated on continental shelves^{3,8}. Pelagic sharks are among the widest-ranging vertebrates, with some species exhibiting annual migrations on the ocean-basin scale⁹, long term trans-ocean movements¹⁰ and/or fine-scale site fidelity to preferred shelf and open ocean areas^{5,9,11}. These behaviours could cause extensive spatial overlap with different fisheries that exploit regions ranging from coastal areas to the deep ocean. On average, large pelagic sharks account for 52% of all of the identified shark catch worldwide, from both shark-targeting fisheries and as bycatch¹². Regional declines in abundance of pelagic sharks have previously been reported^{13,14}, but it is unclear whether exposure to high levels of fishing effort extends across ocean-wide population ranges and overlaps areas of the high seas in which sharks are most abundant^{5,13}. The conservation of pelagic sharks—the management of which on the high seas is currently limited^{12,15,16}—would benefit greatly from a clearer understanding of the spatial relationships between the habitats of sharks and active fishing zones. However, obtaining unbiased estimates of the distributions of sharks and fishing effort is complicated by the fact that most data on pelagic sharks come from catch records and other fishery-dependent sources^{4,15,16}.

Here we provide a global estimate of the extent of overlap in the use of space between sharks and industrial fisheries. This estimate is based on analysis of the movements of pelagic sharks tagged with satellite transmitters in the Atlantic, Indian and Pacific Oceans, together with the movements of fishing vessels that are monitored globally by the automatic identification system (AIS), which was developed as a vessel safety and anti-collision system (Methods). Our study focuses on 23 species of large pelagic sharks that occupy oceanic and/or neritic habitats, which span a broad distribution from cold-temperate to tropical waters (Supplementary Table 1). All of these species face some level of fishing pressure from coastal, shelf and/or high-seas fisheries: the International Union for the Conservation of Nature (IUCN) Red List assesses almost two-thirds of these species as being endangered (26%) or vulnerable (39%), and a further quarter as near-threatened (26%) (Supplementary Table 2). Although regional-fisheries management

organizations are tasked with the management of sharks in the high seas, little or no management is in place for most species^{3,5,12–18}.

Movement patterns of sharks and fishing vessels

The 11 shark species (or taxa groups) that accounted for 96% of the 1,804 satellite tags that were deployed are among the largest of shark species: blue sharks (*Prionace glauca*); shortfin mako sharks (*Isurus oxyrinchus*); tiger sharks (*Galeocerdo cuvier*); salmon sharks (*Lamna ditropis*); whale sharks (*Rhincodon typus*); white sharks (*Carcharodon carcharias*); oceanic whitetip sharks (*Carcharhinus longimanus*); porbeagle sharks (*Lamna nasus*); silky sharks (*Carcharhinus falciformis*); bull sharks (*Carcharhinus leucas*); and hammerhead sharks (*Sphyrna* spp.) (Supplementary Tables 3–5). Movement patterns indicated that multiple species aggregated within the same major oceanographic features (Fig. 1), such as the Gulf Stream (blue sharks, shortfin mako sharks, tiger sharks, white sharks and porbeagle sharks), the California Current (blue sharks, shortfin mako sharks, white sharks and salmon sharks) and the East Australian Current (blue sharks, shortfin mako sharks, tiger sharks, white sharks and porbeagle sharks), (Extended Data Fig. 1; see ‘Supplementary results and discussion, section 2.1’ in the Supplementary Information). The global relative density map (Fig. 2a) reveals distribution patterns of pelagic sharks and the locations of space-use hotspots (defined here as areas with ≥ 75 th percentile of weighted daily location density) (Methods). Major space-use hotspots of tracked pelagic sharks in the Atlantic Ocean were in the Gulf Stream and its western approaches, the Caribbean Sea, the Gulf of Mexico and around oceanic islands such as the Azores (Fig. 2a, Supplementary Table 6). In the Indian Ocean, space-use hotspots were evident in the Agulhas Current, Mozambique Channel, the South Australian Basin and northwest Australia, and in the Pacific Ocean, space-use hotspots were in the California Current, Galapagos Islands and around New Zealand. Although, as expected, tagging sites occurred in some space-use hotspots (as tagging rates are inherently higher in hotspots), we also identified space-use hotspots in which no tagging sites occurred in the North Atlantic Ocean (outer Gulf Stream, Charlie Gibbs Fracture Zone, western European shelf edge and the Bay of Biscay), the Indian Ocean (southern Madagascar, the Crozet and Amsterdam Islands, and the South Australian Basin) and the Pacific Ocean (Alaska Current, outer California Current, the white shark ‘café’ area, halfway between Baja California and Hawaii¹¹, North Equatorial Current, Clipperton Island

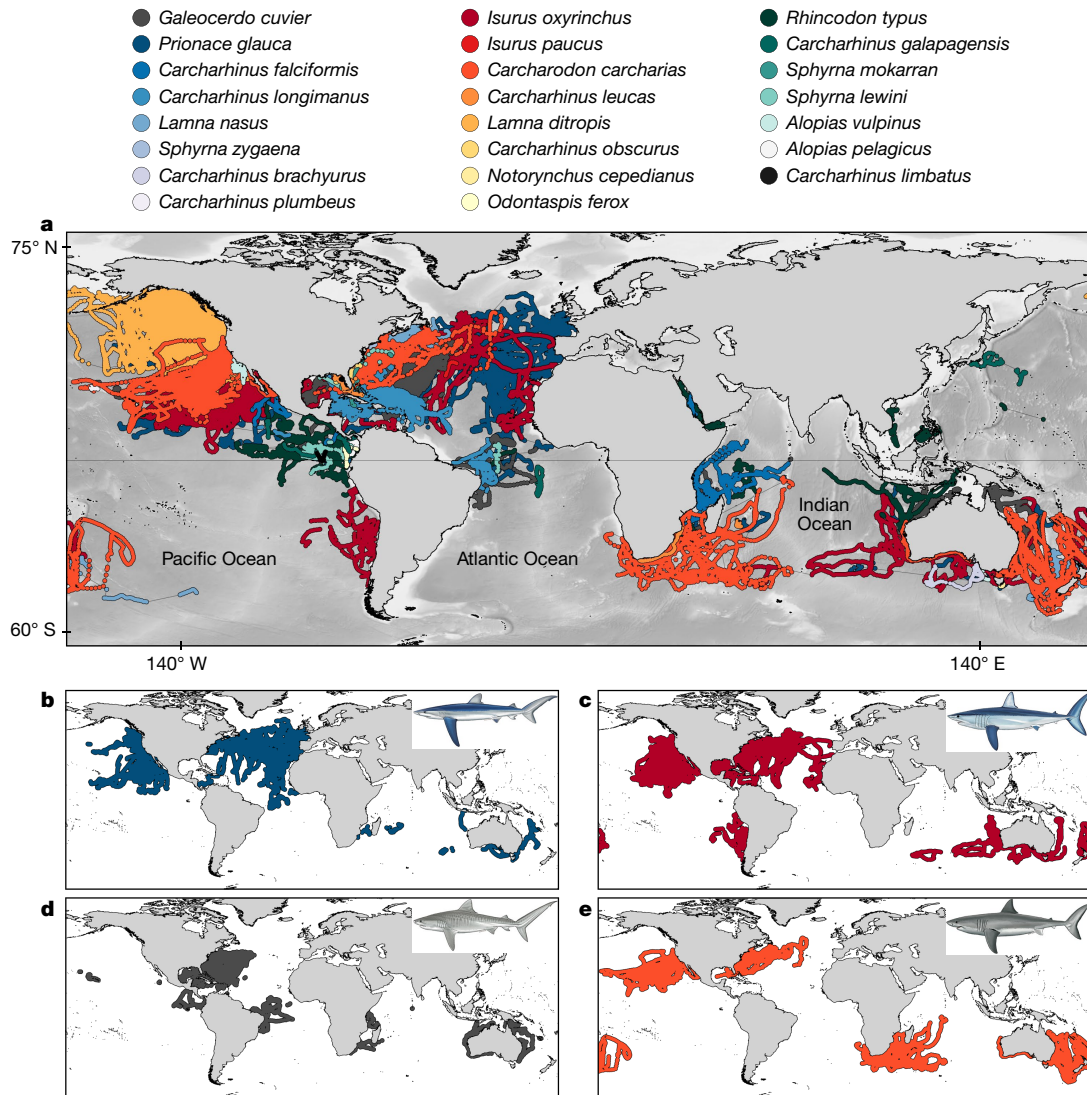


Fig. 1 | Movements of oceanic and neritic pelagic sharks. a, Daily state-space model locations, estimated from locations obtained via satellite transmitters deployed on 1,681 sharks from 23 species between 2002 and 2017. **b–e**, Extent of the space use areas of individual shark species are

illustrated for the species with the greatest numbers of tags deployed across multiple ocean regions: blue shark (**b**), shortfin mako shark (**c**), tiger shark (**d**) and white shark (**e**). Shark images created by M. Dando.

and Kermadec Islands) (Extended Data Fig. 1). There was consistency between our fine-scale identification of hotspots of space use by sharks, and the coarse-scale species richness hotspots of oceanic predators estimated from fishery-dependent catch data⁴ (see ‘Supplementary results and discussion, section 2.1’ in the Supplementary Information).

To determine the extent to which hotspots of space use by sharks fall under the footprint of global industrialized fisheries, we mapped the movements of fishing vessels that carry AIS transmitters (estimated to be fitted on 50–75% of active vessels that are over 24 m in length)^{19–23}. First, we mapped the mean annual and mean monthly fishing effort (in days) of AIS-equipped fishing vessels that use various types of fishing gear^{19,20} for the period 2012–2016 (Extended Data Fig. 2, Methods), and then mapped the estimated global fishing effort of drifting pelagic longline and purse seine vessels separately (as these two types of gear catch the majority of pelagic sharks^{12,15}) (Fig. 2b, Extended Data Fig. 2). The global distribution map of longline fishing effort identifies several large-scale areas that are used heavily, such as the North Atlantic Ocean, southwest Indian Ocean, and the central equatorial and northwest regions of the Pacific Ocean (Fig. 2b, Extended Data Figs. 1, 2). There were also areas in which industrial longline activity appeared sparse; for example, the central and southwest North Atlantic Ocean, northeast Pacific Ocean and northern Indian Ocean. We focused our

detailed analysis of shark overlap with that of longline fishing effort, as it is this gear that catches most pelagic sharks globally¹⁵ and because most of the other types of gear used by the AIS-equipped fishing vessels represented in the dataset do not target or generate abundant bycatch of pelagic sharks^{19,20} (see ‘Supplementary results and discussion, section 2.2’ in the Supplementary Information). The number of longline-fishing-effort days estimated for the Atlantic Ocean using AIS data was positively correlated with the number of observed baited longline hooks (observed hooks) that were deployed in the Atlantic Ocean (Spearman’s $r = 0.182$, $P = 0.008$; $n = 241$) (Methods), which confirms that longline-fishing-effort days based on AIS data are indicative of actual fishing effort^{19,20}.

Spatial overlap of sharks and fishing effort

To explore the spatial heterogeneities of sharks and vessels, we used generalized additive models to determine how the relative density of location estimates for sharks and the distribution of longline-fishing effort were affected by environmental covariates (Methods, Supplementary Table 7). Distributions of the density of pelagic sharks and of the fishing effort of pelagic longline vessels were best explained by the same drivers: both demonstrated strong relationships with habitat types that are characterized by surface and subsurface temperature

gradients (fronts²⁴ and thermoclines) and/or high primary productivity (Extended Data Fig. 3, Supplementary Table 8, see 'Supplementary results and discussion, section 2.3' in the Supplementary Information). The similar environmental drivers that we identified between shark density and fishing effort predict a high spatial overlap between sharks and fishers: sharks are known to aggregate in biologically productive features (such as fronts) to enhance their foraging opportunities^{5,6,24}, a behaviour that fishers exploit to increase their chances of making higher catches of commercially valuable epipelagic fishes (including sharks)^{5,6}.

We calculated the spatial overlap of tracked sharks with longline-fishing effort for a mean month within the datasets (Methods). Overlap was defined as the spatial co-occurrence of sharks and fishing effort within a $1^\circ \times 1^\circ$ grid cell in an average month (Methods) (for grid-cell size analysis, see Extended Data Fig. 4, Supplementary Table 9, 'Supplementary results and discussion, section 2.4' in the Supplementary Information). The overlap between the space use of tracked sharks and fishing effort was dominated by pelagic longline gear (Fig. 2; compare the longline distribution in Fig. 2b with the distribution of fishing effort of all AIS-equipped fishing vessels in Extended Data Fig. 2a). Globally, the distribution of longline-fishing activity in the dataset overlapped 24% of the mean monthly space use of tracked sharks at the $1^\circ \times 1^\circ$ scale (mean $\pm$ s.d. monthly overlap, $23.7\% \pm 32.7$; median, 4.5%; $n = 1,681$ tracks). This estimate is unlikely to be biased by the fact that a majority of our tags were deployed in the northwest Atlantic Ocean or northeast Pacific Ocean, because there was relatively low AIS-monitored longline-fishing effort in both regions (Figs. 1a, 2a–c). Across the four regions in which the majority of sharks were tracked, mean monthly spatial overlap of the 11 most-frequently tracked species or taxa groups with longline-fishing effort was 8% (eastern Pacific Ocean), 24% (Oceania), 37% (North Atlantic Ocean) and 38% (southwest Indian Ocean) (Supplementary Table 10). Differences in the overlap patterns between space use and fishing effort between ocean regions—and for species within regions—were not driven by the numbers of tags that we deployed (see 'Supplementary results and discussion, section 2.1' in the Supplementary Information). Overlap varied across species and oceans, which reflects the heterogeneous distributions of both space use by sharks and longline fishing activity (Extended Data Figs. 6, 7). For example, monthly spatial overlap with fishing effort averaged across all oceans ranged from 49% for the blue shark to 1.3% for the salmon shark. Between oceans, the overlap of space use by blue sharks (the pelagic shark that is most-commonly caught by open-ocean longline fleets¹⁷) with fishing effort was 76% in the North Atlantic Ocean, which decreased to 14% in the eastern Pacific Ocean (median overlap values are given in Supplementary Tables 10, 11).

To estimate the potential exposure of sharks in different regions of the oceans to longline-fishing effort, we calculated the mean monthly fishing effort to which individual sharks were exposed in each grid cell that the shark occupied during a corresponding month, standardized to account for variations in the durations of individual tracks; this is hereafter referred to as the fishing exposure index (FEI) (Methods). As expected, across all oceans and species sharks were exposed to highly variable longline-fishing effort (Supplementary Table 10). Given this, we tested whether the FEI was indicative of actual sharks captured and landed by fisheries. We compared the median monthly individual-species FEI for shark species in the North Atlantic Ocean (the ocean for which we had the most species and tracks) with official records from the Food and Agriculture Organization of the United Nations (FAO) regarding the mean annual landings of these species in the North Atlantic Ocean (Methods). We found a significant positive relationship between these FAO landings data and the individual-species mean FEI (linear regression, $r^2 = 0.45$, $n = 8$ species (or taxa group), $F = 6.72$, $F_{0.05(1),1.7} = 5.59$, $P < 0.05$) (Extended Data Fig. 5), which implies that the index reflects fishing-induced shark mortality.

Hotspots of spatial overlap of the relative density of sharks with longline-fishing effort were evident, for example, in the Gulf Stream and stretching eastward to the Azores, on the western European shelf

edge, in the west African upwelling, in the California Current (and white shark café¹¹), in the Agulhas Current, on the southern Great Barrier Reef, and in New Zealand shelf waters (Fig. 2c). This demonstrates that, globally, high levels of fishing effort are focused on extensive hotspots of space use by sharks (Supplementary Tables 6, 12). Nonetheless, substantial areas of the high seas that are used by pelagic sharks may exist that are largely free from the AIS-monitored fishing activity of longline and purse seine vessels, which could be targeted for shark conservation measures (see 'Supplementary results and discussion, section 2.7' in the Supplementary Information). Identifying such areas can only be addressed using fishery-independent distributions (as are presented here). However, a general characteristic of large areas with low levels of longline-fishing activity was lower densities of sharks (<75 th percentile of relative density) (Fig. 2a), which indicates that sharks were not remaining in these areas but were rather moving through them—potentially as part of foraging excursions or migrations for reproduction^{9,11}. The lower relative density of sharks in these areas suggests lower productivity—which is supported by our modelling results (model 1) (Extended Data Fig. 3)—and, consequently, poorer fishing opportunities (which may explain the apparent low levels of fishing effort in these areas). The results also show that very few large hotspots of space use by pelagic sharks occurred in areas that are free of AIS-equipped fishing vessels, particularly those with longline and purse seine gear (Fig. 2c, Extended Data Fig. 2c, d).

Determining the spatial risk to sharks from fishing

The extent of spatial overlap between the relative density distribution of sharks and longline-fishing effort indicates which species are more at risk from fishing, and how this risk is distributed (Fig. 3). Because we demonstrate that a large proportion of shark mortality related to fishing (as represented by landings) is related to longline-fishing effort in areas of shark space use, it follows that sharks that have both a high overlap with fishing effort and a high FEI (greater susceptibility) will be at a greater risk of capture than those with low overlap and low FEI (Fig. 3). We found the main commercially valuable pelagic sharks were grouped within the highest potential risk zone in the North Atlantic Ocean and eastern Pacific Ocean (blue sharks and shortfin mako sharks) (Fig. 3a, b; see 'Supplementary results and discussion, section 2.5' in the Supplementary Information for significance tests, and for results for the other species). In the North Atlantic Ocean, shortfin mako sharks and blue sharks were at a significantly greater risk compared to other tracked sharks, because their mean monthly space use overlap (62% (median, 71%) and 76% (median, 81%), respectively) co-occurred with a high mean FEI (Fig. 3a, Extended Data Figs. 6, 7, Supplementary Table 10). However, exposure risk varied between oceans because, although the spatial overlap with fishing effort of blue sharks remained relatively high in the southwest Indian Ocean, and for blue sharks and shortfin mako sharks in Oceania (mean, 18–47%; median, 11–33%; Supplementary Table 10), individual-species FEI means were lower in areas of overlap in these regions (Fig. 3a, c, d).

Among sharks that are generally considered to be of less commercial value (including tiger sharks and bull sharks), we found that risk from longlines was high in some—but not all—regions. Bull sharks used spatially limited near-shore habitats in tropical regions within the southwest Indian Ocean: in these areas, they were at increased risk owing to high mean overlap with fishing effort (94%) and high mean FEI (Fig. 3c, Supplementary Table 10). This greater susceptibility could lead to high localized catches, which (if replicated elsewhere) could explain why bull sharks are one of the ten most-commonly traded species in the Hong Kong fin market²⁵. By contrast, tiger sharks were exposed to lower overlap and lower mean FEI in all ocean areas except for Oceania, where they were within the highest potential-risk zone (Fig. 3a–d, see 'Supplementary results and discussion, sections 2.5, 2.6' in the Supplementary Information).

High risk was evident for internationally protected sharks under the Convention on International Trade in Endangered Species (CITES) Appendix II and regional-fisheries management organization

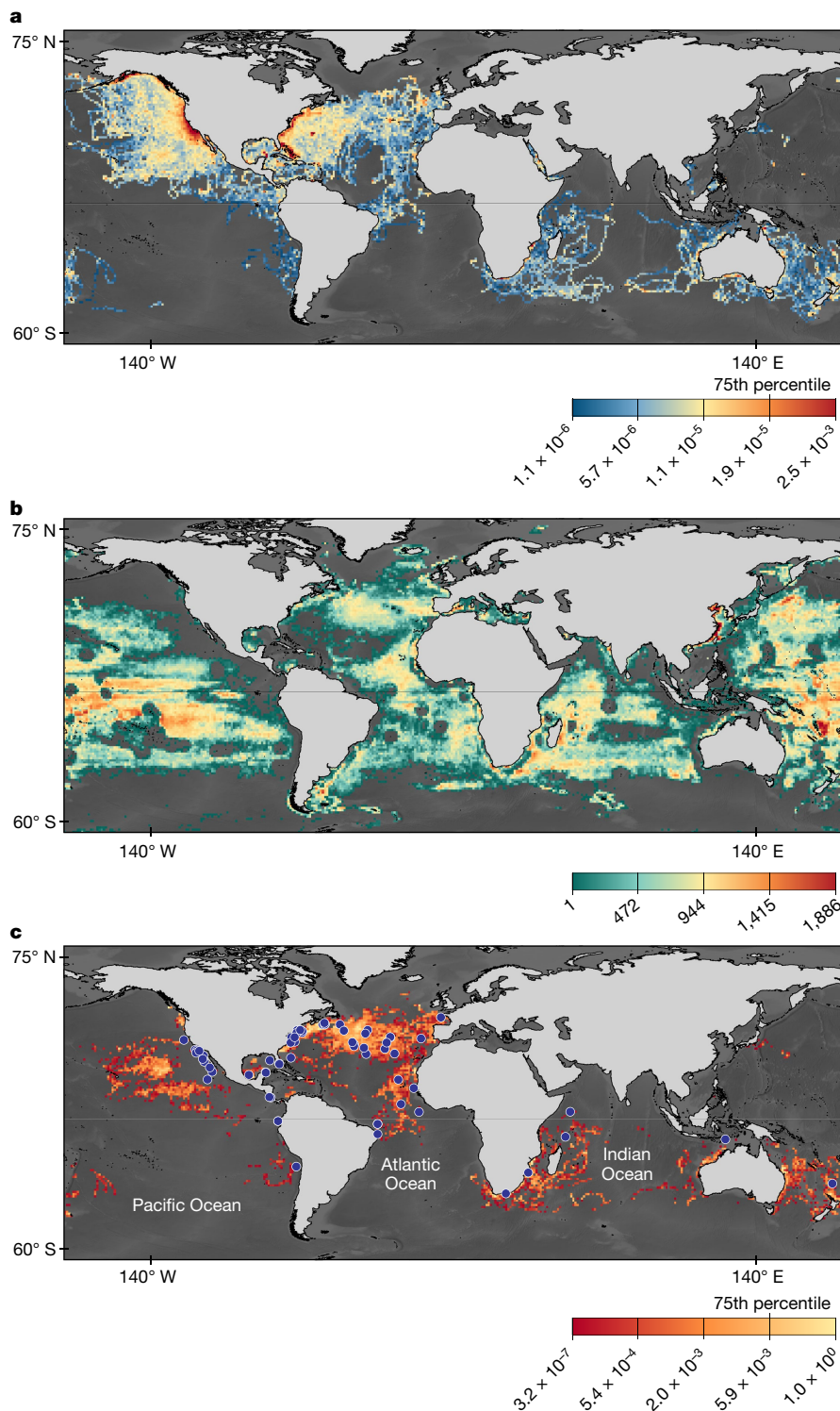


Fig. 2 | Spatial distributions and overlap of sharks and longline fishing vessels. **a**, Relative density of sharks. Distribution of the mean monthly weighted, normalized location density of tracked sharks in $1^\circ \times 1^\circ$ grid cells (shark hotspots were defined by cells with ≥ 75 th percentile of relative density). **b**, Number of days fishing. Mean annual distribution of fishing effort (mean days per grid cell) of AIS-tracked longline fishing vessels in 2012–2016 (see Methods). **c**, FEI. Distribution of the mean monthly overlap and level of fishing effort (in days) that sharks were exposed to in overlapping areas for all species within $1^\circ \times 1^\circ$ grid cells (see Methods). Hotspots of spatial overlap of shark density and fishing effort were defined as cells with ≥ 75 th percentile of mean FEI. Blue circles denote locations at which tagged sharks were caught by commercial fishers, mainly using pelagic longlines and coastal nets.

regulations. The porbeagle shark (global status of endangered on the IUCN Red List) and white shark (global status of vulnerable on the IUCN Red List) have low population sizes compared to historic levels (Supplementary Table 2). In the North Atlantic Ocean and Oceania, we found that the porbeagle shark occurred in the highest-risk zone (Fig. 3a, d), which indicates a high potential for mortality as a result of incidental bycatch. White sharks were in the highest-risk zone in all oceans in which they were tracked, with a mean spatial overlap with fishing effort that ranged from 15% (median 13%) in the east Pacific Ocean to 64% (median 65%) in the southwest Indian Ocean—except for the North Atlantic Ocean, in which their mean FEI was just below the average FEI for all species (Fig. 3, Supplementary Table 10). Our

results show a high risk for porbeagle sharks and white sharks from longline fishing across broad regions, and highlight the need for continued protection—including sufficient coverage by scientific observers on vessels, to underpin accurate data reporting—so that stock rebuilding can continue²⁶. For the northeast Atlantic porbeagle shark, this rebuilding is estimated to take a further 15–30 years to reach sustainable levels even if fishing mortality is reduced to zero¹⁸.

Decreasing the grid-cell size in spatial analyses can lead to concomitant decreases in the estimates of percentages of spatial overlaps^{19,20}, which could potentially affect the patterns of species risk exposure that we found. However, a grid-cell size analysis showed that the patterns of species occurrence within the high- or low-risk zones remained

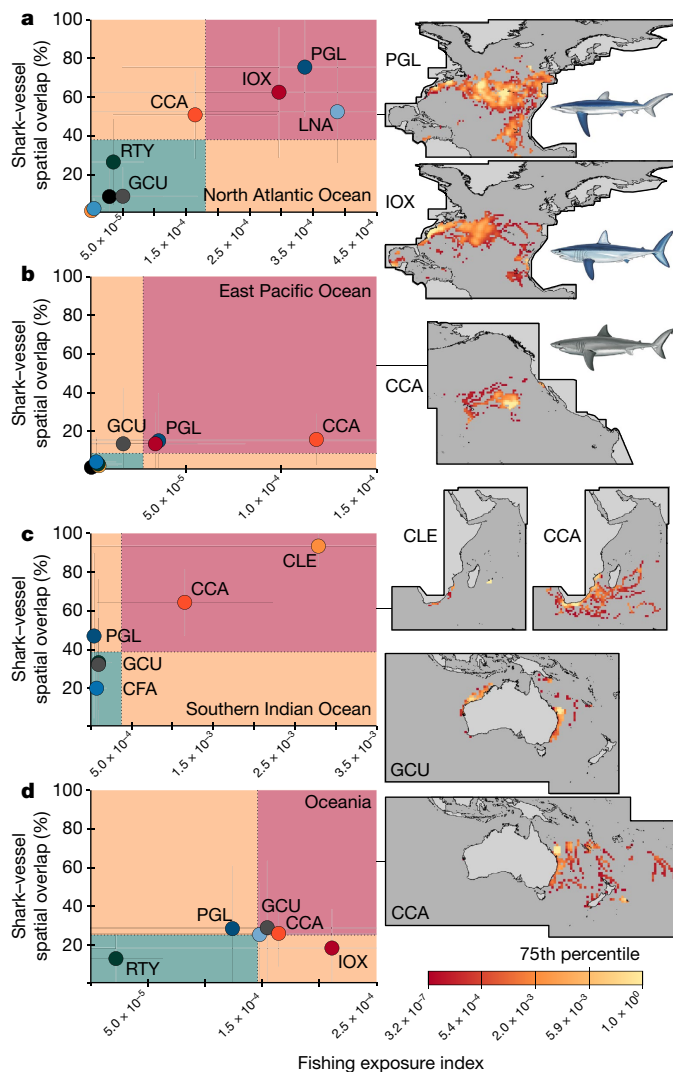


Fig. 3 | Estimated exposure risk of sharks to capture by longline fishing activity. **a–d**, Plots (left) showing the spatial overlap of sharks and longline fishing effort against species mean monthly FEI indicate species that are subject to high overlap and FEI (higher-risk red zone on plot denotes a higher than average overlap and FEI) and those with lower overlap and FEI (lower-risk green zone denotes a lower than average overlap and FEI) for the North Atlantic Ocean (**a**), eastern Pacific Ocean (**b**) and southern Indian Ocean (**c**), and for the Oceania region (**d**). Lines separating the coloured zones are fixed at the average values of spatial overlap (y axis) and FEI (x axis) for all species combined. For each ocean, the amount of fishing effort to which individual shark species were exposed (mean FEI; see Methods) is given in the right panel. CCA, *C. carcharias*; CFA, *C. falciformis*; CLE, *C. leucas*; GCU, *G. cuvier*; IOX, *I. oxyrinchus*; LNA, *L. nasus*; PGL, *P. glauca*; RTY, *R. typus*. Error bars denote ± 1 s.d. of the mean.

consistent, irrespective of the spatial scale at which they were observed (Extended Data Fig. 4) or the subset of tracking years that were analysed (Extended Data Figs. 8, 9, see ‘Supplementary results and discussion, section 2.4’ in the Supplementary Information).

Temporal variation in risk

The highest levels of exposure risk of sharks to longline fisheries were not constant through time, but instead varied seasonally as space use by sharks and fishing vessels shifted in relation to each other (Fig. 4, Extended Data Fig. 10). Overall, for species with sufficient data (Fig. 4) the mean monthly overlap of species space use with fishing effort combined with mean FEI showed that sharks spent 4–6 months per year in the lowest-risk zone and 2–6 months in the highest-risk zone, with

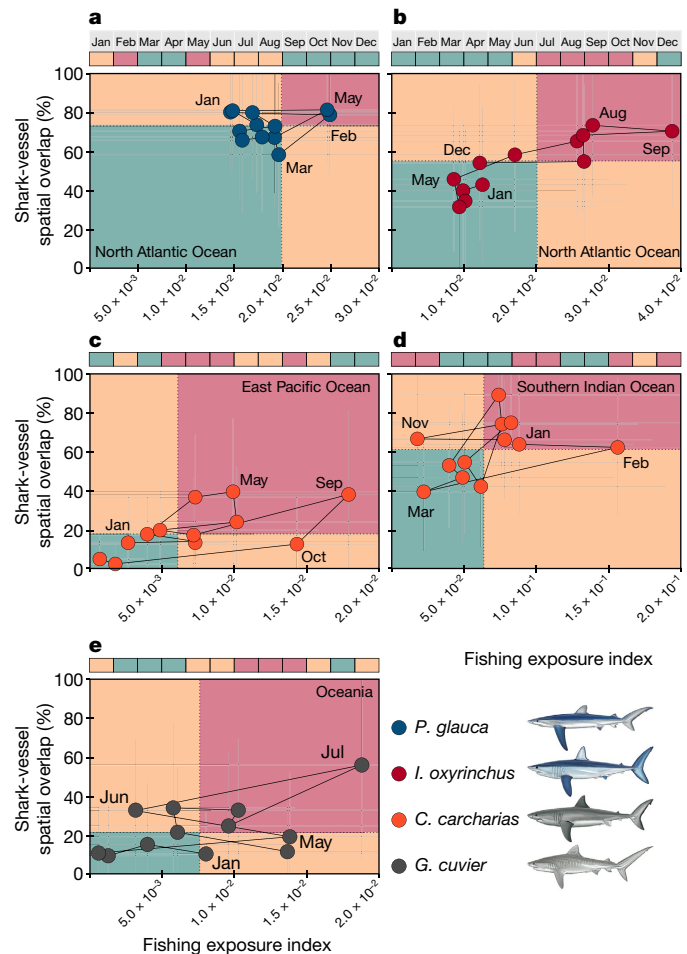


Fig. 4 | Temporal changes in the exposure risk to sharks of longline fishing. **a–e**, Monthly mean spatial overlap of sharks and longline fishing effort versus monthly mean FEI for all individuals of that species for the four most data-rich species in a relative year: blue sharks (**a**), shortfin mako sharks (**b**), white sharks (**c**, **d**) and tiger sharks (**e**). Lines separating the coloured zones are fixed at the respective individual-species average values of spatial overlap (y axis) and FEI (x axis) in a relative year. Horizontal bars denote months with different fishing-exposure risks (red, highest risk; green, lowest risk). Error bars denote ± 1 s.d. of the mean.

differing patterns of changing risk from fishing evident across species (Fig. 4). For example, the highest risk for white sharks in the southwest Indian Ocean and blue sharks in the North Atlantic Ocean occurred at discrete times in the year. For white sharks in the Indian Ocean, this pattern arises from long-range seasonal movements (that take place in December to February, June, July, and October) into annually persistent areas with high mean FEI. For blue sharks, the pattern appears to be driven by sharks and longline vessels co-occurring maximally in boreal winter and summer: lower exposure risk occurs during boreal spring and autumn, as sharks migrate north before returning south⁵. Longline fisheries also make this seasonal south–north–south movement, but lag behind movements of blue sharks (and thus exhibit lower mean overlap and FEI during these times) (Extended Data Fig. 10a–d). Similarly, annual risk patterns of white sharks in the eastern Pacific and tiger sharks from around Australia were driven by migratory behaviour, with the highest risk occurring for three consecutive months in boreal (for white sharks) and austral (for tiger sharks) spring as sharks arrive in areas with higher-than-average exposure to longline-fishing effort (Fig. 4c, e). By contrast, shortfin mako sharks in the North Atlantic Ocean were exposed to high mean overlap (about 60%) and high mean FEI continually through the boreal summer and autumn (July to October), principally owing to their occupation of a space-use

hotspot that is located where the Gulf Stream and Labrador Current converge, which results in persistent high overlap with high levels of longline-fishing effort (Fig. 4b, Extended Data Fig. 10e–h). Tracking of shortfin mako sharks and vessels indicates that fishery-induced mortality within this hotspot is therefore likely to be high. This was confirmed by the high overall return rate of satellite tags (19.3%) attached to shortfin mako sharks in the Atlantic Ocean ($n = 119$ tags; tracking duration: mean $\pm$ s.d., 161.5 ± 156.9 days; median, 109 days), which were returned to us after sharks were captured by longline fleets in the Atlantic Ocean during the study. To our knowledge, this is the highest species-specific return rate for sharks that has yet been recorded in an ocean-scale (as opposed to a regional-scale) study^{27,28} (Fig. 2c, Supplementary Table 13, see ‘Supplementary results and discussion, section 2.6’ in the Supplementary Information).

Discussion

Our results show that globally important habitat areas for threatened pelagic sharks overlap considerably with industrial fishing activity, in both space and time. Given the high levels of fishing effort in hotspots of space use of many species for substantial portions of the year, and the very-few tracked hotspots that are free from exploitation, our study reveals that the exposure risk of sharks to fisheries in the high seas is spatially extensive—for some species, this exposure risk stretches nearly across their entire ocean-scale population ranges. Overall, the patterns that we observed suggest a future in which sharks experience only limited spatial refuge from industrial longline-fishing effort, which is currently centred on ecologically important hotspots of space use for oceanic sharks. Our distribution maps are a first, but essential, underpinning for a conservation blueprint for pelagic sharks in the high seas. Our study highlights the scale of the overlap between fishing effort and hotspots of space use by sharks, and argues for more-effective and timely monitoring, reporting and management of pelagic sharks as a result. To enhance the recovery of vulnerable species, one solution is the designation of large-scale marine protected areas²⁹ around ecologically important space-use hotspots of pelagic sharks²⁴—notwithstanding the need for more-complete reporting of catch data, with enforcement to support stricter conventional management bycatch prohibitions, quotas or minimum sizes^{5,16}. This study also outlines shark space-use hotspot locations at which AIS-monitored fishing effort appears at present to be relatively low; these locations are ones in which shark conservation could be maximized, while minimizing the effect on fishing activity that is not directed at sharks (‘Supplementary results and discussion, sections 2.6, 2.7’ in the Supplementary Information). Although it would be challenging to develop a legally binding treaty for managing high-seas fauna²¹, burgeoning technology for global surveillance and enforcement now offers additional options for a step change in ocean management^{6,30}.

Satellite monitoring of marine megafauna^{1,5,11,31}, oceanographic features (eddies and fronts)^{6,24} and the distribution of global fishing vessels^{19,20} could provide signals of shifting space use by wide-ranging sharks and other marine megafauna owing to environmental changes, which—in turn—could inform the designation of new temporary time–area closures to industrial fishing⁶ and the tracking of fishers’ displacement activities²¹. The potential of near-real-time, synoptic measurements of marine megafauna, fishing activity and the marine environment (particularly given the remoteness and vast extent of the high seas) suggests technology-led conservation measures will be crucial tools for reversing the observed declines in iconic ocean predators³ such as pelagic sharks^{12–14,30}. In the future, conservation technology could develop towards the incorporation of adaptive management strategies^{6,30} that are actionable in real time to assess risks in the overlap between fishing vessels and sharks across the global ocean.

Online content

Any methods, additional references, Nature Research reporting summaries, source data, extended data, supplementary information, acknowledgements, peer review information; details of author contributions and competing interests; and

statements of data and code availability are available at <https://doi.org/10.1038/s41586-019-1444-4>.

Received: 4 October 2018; Accepted: 10 July 2019;

Published online 24 July 2019.

- Hays, G. C. et al. Key questions in marine megafauna movement ecology. *Trends Ecol. Evol.* **31**, 463–475 (2016).
- Lewison, R. L. et al. Global patterns of marine mammal, seabird, and sea turtle bycatch reveal taxa-specific and cumulative megafauna hotspots. *Proc. Natl Acad. Sci. USA* **111**, 5271–5276 (2014).
- McCauley, D. J. et al. Marine defaunation: animal loss in the global ocean. *Science* **347**, 1255641 (2015).
- Worm, B., Sandow, M., Oschlies, A., Lotze, H. K. & Myers, R. A. Global patterns of predator diversity in the open oceans. *Science* **309**, 1365–1369 (2005).
- Queiroz, N. et al. Ocean-wide tracking of pelagic sharks reveals extent of overlap with longline fishing hotspots. *Proc. Natl Acad. Sci. USA* **113**, 1582–1587 (2016).
- Scales, K. L. et al. Fisheries bycatch risk to marine megafauna is intensified in Lagrangian coherent structures. *Proc. Natl Acad. Sci. USA* **115**, 7362–7367 (2018).
- O’Connor, S., Ono, R. & Clarkson, C. Pelagic fishing at 42,000 years before the present and the maritime skills of modern humans. *Science* **334**, 1117–1121 (2011).
- Tickler, D., Meeuwig, J. J., Palomares, M.-L., Pauly, D., Zeller, D. Far from home: distance patterns of global fishing fleets. *Sci. Adv.* **4**, eaar3279 (2018).
- Lea, J. S. E. et al. Repeated, long-distance migrations by a philopatric predator targeting highly contrasting ecosystems. *Sci. Rep.* **5**, 11202 (2015).
- Guzman, H. M., Comez, C. G., Hearn, A. & Eckert, S. A. Longest recorded trans-Pacific migration of a whale shark (*Rhincodon typus*). *Mar. Biodivers. Rec.* **11**, 8 (2018).
- Block, B. A. et al. Tracking apex marine predator movements in a dynamic ocean. *Nature* **475**, 86–90 (2011).
- Worm, B. et al. Global catches, exploitation rates, and rebuilding options for sharks. *Mar. Policy* **40**, 194–204 (2013).
- Baum, J. K. et al. Collapse and conservation of shark populations in the Northwest Atlantic. *Science* **299**, 389–392 (2003).
- Ferretti, F., Worm, B., Britten, G. L., Heithaus, M. R. & Lotze, H. K. Patterns and ecosystem consequences of shark declines in the ocean. *Ecol. Lett.* **13**, 1055–1071 (2010).
- Oliver, S., Braccini, M., Newman, S. J. & Harvey, E. S. Global patterns in the bycatch of sharks and rays. *Mar. Policy* **54**, 86–97 (2015).
- Campana, S. E. Transboundary movements, unmonitored fishing mortality, and ineffective international fisheries management pose risks for pelagic sharks in the Northwest Atlantic. *Can. J. Fish. Aquat. Sci.* **73**, 1599–1607 (2016).
- International Commission for the Conservation of Atlantic Tunas (ICCAT). *Report of the 2017 ICCAT Shortfin Mako Assessment Meeting (Madrid, Spain)*. <https://www.iccat.int/en/assess.html> (2017).
- International Commission for the Conservation of Atlantic Tunas. *Report of the Standing Committee on Research and Statistics (SCRS)*. doc. no. PLE 104/2017 <https://www.iccat.int/en/scrs.html> (2017).
- Kroodsma, D. A. et al. Tracking the global footprint of fisheries. *Science* **359**, 904–908 (2018).
- Kroodsma, D. A. et al. Response to Comment on “Tracking the global footprint of fisheries”. *Science* **361**, eaat7789 (2018).
- McCauley, D. J. et al. Ending hide and seek at sea. *Science* **351**, 1148–1150 (2016).
- Shepperson, J. L. et al. A comparison of VMS and AIS data: the effect of data coverage and vessel position recording frequency on estimates of fishing footprints. *ICES J. Mar. Sci.* **75**, 988–998 (2018).
- Sala, E. et al. The economics of fishing the high seas. *Sci. Adv.* **4**, eaat2504 (2018).
- Scales, K. L. et al. On the front line: frontal zones as priority at-sea conservation areas for mobile marine vertebrates. *J. Appl. Ecol.* **51**, 1575–1583 (2014).
- Fields, A. T. et al. Species composition of the international shark fin trade assessed through a retail-market survey in Hong Kong. *Conserv. Biol.* **32**, 376–389 (2018).
- Curtis, T. H. et al. Seasonal distribution and historic trends in abundance of white sharks, *Carcharodon carcharias*, in the western North Atlantic Ocean. *PLoS ONE* **9**, e99240 (2014).
- Kohler, N. E. & Turner, P. A. Shark tagging: a review of conventional methods and studies. *Environ. Biol. Fishes* **60**, 191–224 (2001).
- Byrne, M. E. et al. Satellite telemetry reveals higher fishing mortality rates than previously estimated, suggesting overfishing of an apex marine predator. *Proc. R. Soc. Lond. B* **284**, 20170658 (2017).
- O’Leary, B. C. et al. Addressing criticisms of large-scale marine protected areas. *Bioscience* **68**, 359–370 (2018).
- Hays, G. C. et al. Translating marine animal tracking data into conservation policy and management. *Trends Ecol. Evol.* **34**, 459–473 (2019).
- Sequeira, A. M. M. et al. Convergence of marine megafauna movement patterns in coastal and open oceans. *Proc. Natl Acad. Sci. USA* **115**, 3072–3077 (2018).

Publisher’s note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

© The Author(s), under exclusive licence to Springer Nature Limited 2019

Nuno Queiroz^{1,2}, Nicolas E. Humphries², Ana Couto¹, Marisa Vedor^{1,3}, Ivo da Costa¹, Ana M. M. Sequeira^{4,5}, Gonzalo Mucientes¹, António M. Santos^{1,3}, Francisco J. Abascal⁶, Debra L. Abercrombie⁷, Katya Abrantes⁸, David Acuña-Marrero⁹, André S. Afonso^{10,11}, Pedro Afonso^{12,13,14}, Darrell Anders¹⁵, Gonzalo Araujo¹⁶, Randall Arauz^{17,18,19}, Pascal Bach²⁰, Adam Barnett⁸, Diego Bernal²¹, Michael L. Berumen²², Sandra Bessudo Lion^{19,23}, Natalia P. A. Bezerra¹⁰, Antonin V. Blaison²⁰, Barbara A. Block²⁴, Mark E. Bond²⁵, Ramón Bonfil²⁶, Russell W. Bradford²⁷, Camrin D. Braun²⁸, Edward J. Brooks²⁹, Annabelle Brooks^{29,30}, Judith Brown³¹, Barry D. Bruce²⁷, Michael E. Byrne^{32,33}, Steven E. Campana³⁴, Aaron B. Carlisle³⁵, Demian D. Chapman²⁵, Taylor K. Chapple²⁴, John Chisholm³⁶, Christopher R. Clarke³⁷, Eric G. Clua³⁸, Jesse E. M. Cochran²², Estelle C. Crochelet^{39,40}, Laurent Dagorn²⁰, Ryan Daly^{41,42}, Daniel Devia Cortés⁴³, Thomas K. Doyle^{44,45}, Michael Drew⁴⁶, Clinton A. J. Duffy⁴⁷, Thor Erikson⁴⁸, Eduardo Espinoza^{19,49}, Luciana C. Ferreira⁵⁰, Francesco Ferretti²⁴, John D. Filmlalter^{20,42}, G. Chris Fischer⁵¹, Richard Fitzpatrick⁸, Jorge Fontes^{12,13,14}, Fabien Forget²⁰, Mark Fowler⁵², Goldsmowth⁵³, Austin J. Gallagher^{54,55}, Enrico Gennari^{42,56,57}, Simon D. Goldsborough⁵⁸, Matthew J. Gollock⁵⁹, Jonathan R. Green⁶⁰, Johan A. Gustafson⁶¹, Tristan L. Guttridge⁶², Hector M. Guzman⁶³, Neil Hammerschlag^{55,64}, Luke Harman⁴⁴, Fábio H. V. Hazin¹⁰, Matthew Heard⁴⁶, Alex R. Hearn^{19,65,66}, John C. Holdsworth⁶⁷, Bonnie J. Holmes⁶⁸, Lucy A. Howey⁶⁹, Mauricio Hoyos^{19,70}, Robert E. Hueter⁷¹, Nigel E. Hussey⁷², Charlie Huvener⁴⁶, Dylan T. Irion⁷³, David M. P. Jacoby⁷⁴, Oliver J. D. Jewell^{75,76}, Ryan Johnson⁷⁷, Lance K. B. Jordan⁶⁹, Salvador J. Jorgensen⁷⁸, Warren Joyce⁵², Clare A. Keating Daly⁴¹, James T. Ketchum^{19,70}, A. Peter Klimley^{19,79}, Alison A. Kock^{42,80,81,82}, Pieter Koen⁸³, Felipe Ladino²³, Fernanda O. Lana⁸⁴, James S. E. Lea^{37,85}, Fiona Llewellyn⁵⁹, Warrick S. Lyon⁵³, Anna MacDonnell⁵², Bruno C. L. Macena^{10,13}, Heather Marshall^{21,86}, Jaime D. McAllister⁸⁷, Rory McAuley^{88,89}, Michael A. Meyer¹⁵, John J. Morris⁷¹, Emily R. Nelson⁵⁵, Yannis P. Papastamatiou²⁵, Toby A. Patterson²⁷, Cesar Peñaherrera-Palma^{19,90}, Julian G. Pepperell⁹¹, Simon J. Pierce⁹², Francois Poisson²⁰, Lina Maria Quintero²³, Andrew J. Richardson⁹³, Paul J. Rogers⁵⁸, Christoph A. Rohner⁹², David R. L. Rowat⁹⁴, Melita Samoilys⁹⁵, Jayson M. Semmens⁸⁷, Marcus Sheaves⁹⁶, George Shillinger^{19,24,96}, Mahmood Shivji³², Sarika Singh¹⁵, Gregory B. Skomal³⁶, Malcolm J. Smale⁹⁷, Laureenne B. Snyders¹⁵, German Soler^{19,23,87}, Marc Soria²⁰, Kilian M. Stehfest⁸⁷, John D. Stevens²⁷, Simon R. Thorrold⁹⁸, Mariana T. Tolotti²⁰, Alison Towner^{57,76}, Paulo Travassos¹⁰, John P. Tyminski⁷¹, Frederic Vandepierre^{12,13,14}, Jeremy J. Vaudso³², Yuuki Y. Watanabe^{99,100}, Sam B. Weber¹⁰¹, Bradley M. Wetherbee^{32,102}, Timothy D. White²⁴, Sean Williams²⁹, Patricia M. Zárte^{19,103}, Robert Harcourt¹⁰⁴, Graeme C. Hays¹⁰⁵, Mark G. Meekan⁵⁰, Michele Thums⁵⁰, Xabier Irigoien^{106,107}, Victor M. Eguiluz¹⁰⁸, Carlos M. Duarte²², Lara L. Sousa^{2,109}, Samantha J. Simpson^{2,110}, Emily J. Southall² & David W. Sims^{2,110,111*}

¹Centro de Investigação em Biodiversidade e Recursos Genéticos/Research Network in Biodiversity and Evolutionary Biology, Campus Agrário de Vairão, Universidade do Porto, Vairão, Portugal. ²Marine Biological Association of the United Kingdom, Plymouth, UK. ³Departamento de Biologia, Faculdade de Ciências da Universidade do Porto, Porto, Portugal. ⁴UWA Oceans Institute, Indian Ocean Marine Research Centre, University of Western Australia, Crawley, Western Australia, Australia. ⁵School of Biological Sciences, University of Western Australia, Crawley, Western Australia, Australia. ⁶Spanish Institute of Oceanography, Santa Cruz de Tenerife, Spain. ⁷Abercrombie and Fish, Port Jefferson Station, Jefferson, NY, USA. ⁸Marine Biology and Aquaculture Unit, College of Science and Engineering, James Cook University, Cairns, Queensland, Australia. ⁹School of Natural and Computational Sciences, Massey University, Palmerston North, New Zealand. ¹⁰Universidade Federal Rural de Pernambuco (UFRPE), Departamento de Pesca e Aquicultura, Recife, Brazil. ¹¹MARE, Marine and Environmental Sciences Center, Instituto Politécnico de Leiria, Peniche, Portugal. ¹²MARE, Laboratório Marítimo da Guia, Faculdade de Ciências da Universidade de Lisboa, Cascais, Portugal. ¹³Institute of Marine Research (IMAR), Departamento de Oceanografia e Pescas, Universidade dos Açores, Horta, Portugal. ¹⁴Oceanos - Departamento de Oceanografia e Pescas, Universidade dos Açores, Horta, Portugal. ¹⁵Department of Environmental Affairs, Oceans and Coasts Research, Cape Town, South Africa. ¹⁶Large Marine Vertebrates Research Institute Philippines, Jagna, Philippines. ¹⁷Fins Attached Marine Research and Conservation, Colorado Springs, CO, USA. ¹⁸Programa Restauración de Tortugas Marinas PRETOMA, San José, Costa Rica. ¹⁹MigraMar, Olema, CA, USA. ²⁰Institut de Recherche pour le Développement, UMR MARBEC (IRD, Ifremer, Université de Montpellier, CNRS), Sète, France. ²¹Biology Department, University of Massachusetts Dartmouth, Dartmouth, MA, USA. ²²Red Sea Research Center, Division of Biological and Environmental Science and Engineering, King Abdullah University of Science and Technology, Thuwal, Saudi Arabia. ²³Fundación Malpelo y Otros Ecosistemas Marinos, Bogotá, Colombia. ²⁴Hopkins Marine Station of Stanford University, Pacific Grove, CA, USA. ²⁵Department of Biological Sciences, Florida International University, North Miami, FL, USA. ²⁶Océanos Vivos, Mexico City, Mexico. ²⁷CSIRO Oceans and Atmosphere, Hobart, Tasmania, Australia. ²⁸Massachusetts Institute of Technology–Woods Hole Oceanographic

Institution Joint Program in Oceanography/Applied Ocean Science and Engineering, Cambridge, MA, USA. ²⁹Shark Research and Conservation Program, Cape Eleuthera Institute, Eleuthera, Bahamas. ³⁰University of Exeter, Exeter, UK. ³¹South Atlantic Environmental Research Institute, Stanley, Falkland Islands. ³²Department of Biological Sciences, The Guy Harvey Research Institute, Nova Southeastern University, Dania Beach, FL, USA. ³³School of Natural Resources, University of Missouri, Columbia, MO, USA. ³⁴Life and Environmental Sciences, University of Iceland, Reykjavik, Iceland. ³⁵School of Marine Science and Policy, University of Delaware, Lewes, DE, USA. ³⁶Massachusetts Division of Marine Fisheries, New Bedford, MA, USA. ³⁷Marine Research Facility, Jeddah, Saudi Arabia. ³⁸PSL, Labex CORAIL, CRIOTE USR3278 EPHE-CNRS-UPVD, Papetō'ai, French Polynesia. ³⁹Agence de Recherche pour la Biodiversité à la Réunion (ARBRE), Saint-Denis, Réunion, Marseille, France. ⁴⁰Institut de Recherche pour le Développement, UMR 228 ESPACE-DEV, Saint-Denis, Réunion, Marseille, France. ⁴¹Save Our Seas Foundation–D'Arros Research Centre (SOSF-DRC), Geneva, Switzerland. ⁴²South African Institute for Aquatic Biodiversity (SAIAB), Grahamstown, South Africa. ⁴³Department of Fisheries Evaluation, Fisheries Research Division, Instituto de Fomento Pesquero (IFOP), Valparaíso, Chile. ⁴⁴School of Biological, Earth and Environmental Sciences, University College Cork, Cork, Ireland. ⁴⁵MaREI Centre, Environmental Research Institute, University College Cork, Cork, Ireland. ⁴⁶College of Science and Engineering, Flinders University, Adelaide, South Australia, Australia. ⁴⁷Department of Conservation, Auckland, New Zealand. ⁴⁸Geological Sciences, South African Institute for Aquatic Biodiversity (SAIAB), Durban, South Africa. ⁴⁹Dirección Parque Nacional Galápagos, Puerto Ayora, Ecuador. ⁵⁰Australian Institute of Marine Science, Indian Ocean Marine Research Centre (UWA), Crawley, Western Australia, Australia. ⁵¹OCEARCH, Park City, UT, USA. ⁵²Bedford Institute of Oceanography, Dartmouth, Nova Scotia, Canada. ⁵³National Institute of Water and Atmospheric Research, Wellington, New Zealand. ⁵⁴Beneath the Waves, Herndon, VA, USA. ⁵⁵Rosentiel School of Marine and Atmospheric Science, University of Miami, Miami, FL, USA. ⁵⁶Oceans Research, Mossel Bay, South Africa. ⁵⁷Department of Ichthyology and Fisheries Science, Rhodes University, Grahamstown, South Africa. ⁵⁸SARDI Aquatic Sciences, Adelaide, South Australia, Australia. ⁵⁹Zoological Society of London, London, UK. ⁶⁰Galapagos Whale Shark Project, Puerto Ayora, Galapagos Islands, Ecuador. ⁶¹Griffith Centre for Coastal Management, Griffith University School of Engineering, Griffith University, Gold Coast, Queensland, Australia. ⁶²Bimini Biological Field Station, South Bimini, Bahamas. ⁶³Smithsonian Tropical Research Institute, Panama City, Panama. ⁶⁴Leonard and Jayne Abess Center for Ecosystem Science and Policy, University of Miami, Coral Gables, FL, USA. ⁶⁵Galapagos Science Center, San Cristóbal, Ecuador. ⁶⁶Universidad San Francisco de Quito, Quito, Ecuador. ⁶⁷Blue Water Marine Research, Tutukaka, New Zealand. ⁶⁸University of Queensland, Brisbane, Queensland, Australia. ⁶⁹Microwave Telemetry, Columbia, MD, USA. ⁷⁰Pelagios-Kakunja, La Paz, Mexico. ⁷¹Mote Marine Laboratory, Center for Shark Research, Sarasota, FL, USA. ⁷²Biological Sciences, University of Windsor, Windsor, Ontario, Canada. ⁷³Cape Research and Diver Development, Simon's Town, South Africa. ⁷⁴Institute of Zoology, Zoological Society of London, London, UK. ⁷⁵Centre for Sustainable Aquatic Ecosystems, Harry Butler Institute, Murdoch University, Perth, Western Australia, Australia. ⁷⁶Dyer Island Conservation Trust, Kleinbaai, South Africa. ⁷⁷Blue Wilderness Research Unit, Scottburgh, South Africa. ⁷⁸Monterey Bay Aquarium, Monterey, CA, USA. ⁷⁹University of California Davis, Davis, CA, USA. ⁸⁰Cape Research Centre, South African National Parks, Steenberg, South Africa. ⁸¹Shark Spotters, Fish Hoek, South Africa. ⁸²Institute for Communities and Wildlife in Africa, Department of Biological Sciences, University of Cape Town, Cape Town, South Africa. ⁸³Veterinary Services, Western Cape Department of Agriculture, Elsenvurg, South Africa. ⁸⁴Departamento de Biologia Marinha, Universidade Federal Fluminense (UFF), Niterói, Brazil. ⁸⁵Department of Zoology, University of Cambridge, Cambridge, UK. ⁸⁶Atlantic White Shark Conservancy, Chatham, MA, USA. ⁸⁷Fisheries and Aquaculture Centre, Institute for Marine and Antarctic Studies, University of Tasmania, Hobart, Tasmania, Australia. ⁸⁸Department of Fisheries, Government of Western Australia, Perth, Western Australia, Australia. ⁸⁹Minderoo Foundation, Flourishing Oceans Initiative, Perth, Western Australia, Australia. ⁹⁰Pontificia Universidad Católica del Ecuador Sede Manabí, Portoviejo, Ecuador. ⁹¹Pepperell Research and Consulting, Noosa, Queensland, Australia. ⁹²Marine Megafauna Foundation, Truckee, CA, USA. ⁹³Conservation and Fisheries Department, Ascension Island Government, Georgetown, Ascension Island, UK. ⁹⁴Marine Conservation Society Seychelles, Victoria, Seychelles. ⁹⁵CORDIO, Mombasa, Kenya. ⁹⁶Upwell, Monterey, CA, USA. ⁹⁷Department of Zoology and Institute for Coastal and Marine Research, Nelson Mandela University, Port Elizabeth, South Africa. ⁹⁸Biology Department, Woods Hole Oceanographic Institution, Woods Hole, MA, USA. ⁹⁹National Institute of Polar Research, Tachikawa, Tokyo, Japan. ¹⁰⁰Department of Polar Science, SOKENDAI (The Graduate University for Advanced Studies), Tachikawa, Tokyo, Japan. ¹⁰¹Centre for Ecology and Conservation, University of Exeter, Penryn, UK. ¹⁰²Department of Biological Sciences, University of Rhode Island, Kingston, RI, USA. ¹⁰³Department of Oceanography and Environment, Fisheries Research Division, Instituto de Fomento Pesquero (IFOP), Valparaíso, Chile. ¹⁰⁴Department of Biological Sciences, Macquarie University, Sydney, New South Wales, Australia. ¹⁰⁵School of Life and Environmental Sciences, Deakin University, Geelong, Victoria, Australia. ¹⁰⁶AZTI - Marine Research, Pasaia, Spain. ¹⁰⁷IKERBASQUE, Basque Foundation for Science, Bilbao, Spain. ¹⁰⁸Instituto de Física Interdisciplinar y Sistemas Complejos, Consejo Superior de Investigaciones Científicas, University of the Balearic Islands, Palma de Mallorca, Spain. ¹⁰⁹Wildlife Conservation Research Unit, Department of Zoology, University of Oxford, Tubney, UK. ¹¹⁰Ocean and Earth Science, National Oceanography Centre Southampton, University of Southampton, Southampton, UK. ¹¹¹Centre for Biological Sciences, University of Southampton, Southampton, UK. *e-mail: dws@mba.ac.uk

METHODS

No statistical methods were used to predetermine sample size. The experiments were not randomized and investigators were not blinded to allocation during experiments and outcome assessment.

Study animals and tagging. From 2002 to 2017, we tagged 1,804 pelagic sharks with satellite transmitters at multiple tagging sites in the Atlantic Ocean, Indian Ocean and Pacific Ocean (Extended Data Fig. 1), including 649 in the North Atlantic Ocean, 588 in the eastern Pacific Ocean, 151 in Oceania and 153 in the southwest Indian Ocean: 60% of deployments occurred between 2010 and 2017 (Extended Data Fig. 1, Supplementary Tables 3–5). The number of tagged individuals varied among species, and ranged from 1 to 280. Two types of satellite-transmitter tag (position-only advanced research and global observation satellite (ARGOS) transmitter and pop-off satellite archival transmitter (PSAT)) were used. Sharks were captured with baited hooks (longlines, rod-and-line angling or with handlines), in purse seine during commercial fishing operations, or tagged free-swimming in the water. Tags were attached to the first dorsal fin or in the dorsal musculature. All animal handling procedures were approved by institutional ethical review committees and completed by trained personnel (see Supplementary Information for details). Data were provided by the data owners to the senior author and quality-checked before archiving in a database. Poor-quality data were reported for 123 tags (72 ARGOS and 51 PSAT), owing to (for example) early tag failure, premature tag pop-off or a high percentage of locations estimated with high spatial error (for example, raw computed geolocations over land). All of these resulted in poor state-space model fits, which led to short or unreliable track reconstructions. Analyses were therefore restricted to the remaining 1,681 tracks from 1,066 ARGOS and 615 PSAT tags on sharks from 23 species, ranging in total duration per species from 20 to 57,037 days with a median of 4.1 years total track time per species (Supplementary Table 3). The number of sharks tracked within each region is given in Supplementary Table 14.

Track processing. Movements of PSAT-tagged sharks were estimated using either satellite-relayed data from each tag or from archival data after the tags were physically recovered. Data were provided as: (i) raw shark positions that were previously reconstructed using software provided by the tag manufacturers (for example, Wildlife Computers or Microwave Telemetry), in which daily maximal rate-of-change in light intensity was used to estimate local time of midnight or midday for longitude calculations, and day-length estimation for determining latitude^{32,33}, or (ii) filtered positions in which a state-space model (SSM) (unscented Kalman filter with sea-surface temperature, UKFSST)³⁴ had been applied to correct the raw geolocation estimates and obtain the most-probable track. In the first case, raw positions were corrected using the UKFSST SSM (UKFSST R package) in addition to a bathymetric correction applied to the initial Kalman position estimates (analyzepsat R add-on). A daily time-series of locations was estimated using a continuous-time correlated random walk (CTCRW) Kalman filter³⁵ (crawl R package). UKFSST geolocations were parameterized with s.d. constants, which produces the smallest mean deviation from concurrent ARGOS positions³⁶. In the latter case, the CTCRW SSM was applied to produce regular time-series.

For ARGOS transmitter tags, data were provided as raw ARGOS (Doppler frequency shift) position estimates. Location class Z data—assigned for a failed attempt at obtaining a position—were discarded from the dataset. The remaining raw position estimates (location classes 3, 2, 1, 0, A and B) were analysed point-to-point with a speed filter of 3 m s^{-1} to remove outlier locations. Subsequently, the CTCRW SSM was applied to each individual track, which produced a single position estimate per day using model parameters implemented in the crawl R package³⁵.

Shark tracking data from the Tagging of Pacific Predators (TOPP) program were downloaded from the Animal Tracking Network hosted by the Integrated Ocean Observing System (<https://ioos.noaa.gov/project/atn/>, downloaded September 2017). Both ARGOS and light-based geolocation data in the Animal Tracking Network had already been filtered with a Bayesian-based SSM³⁷. In brief, the SSM was fitted to each track individually, using the WinBUGS software that conducts Bayesian statistical analyses using Markov chain Monte Carlo sampling³⁸. For each track, two Markov chain Monte Carlo chains each of length 10,000 were run and a sample of 2,000 from the joint posterior probability distribution was obtained by discarding the first 5,000 iterations and retaining every 5th of the remaining iterations. SSM fits were posteriorly inspected for obvious problems (for example, unrealistic movements¹¹). Because two different SSMs were applied to data used in this study, we tested for possible biases in the spatial density analysis (see ‘Spatial density analysis’) by comparing $1^\circ \times 1^\circ$ density grid maps obtained with both UKFSST and Bayesian-based filtered tracks using a subset of 83 ARGOS-linked tracks in the North Atlantic Ocean (blue shark, $n = 27$; shortfin mako shark, $n = 42$; white shark, $n = 3$; and oceanic whitetip shark, $n = 11$). Differences in spatial grid density between the two methods were negligible (Supplementary Fig. 1). Thus, tracks with daily locations were reconstructed for 1,681 individuals, totalling 281,724 tracking days (Supplementary Table 3).

Spatial density analysis. To obtain unbiased estimates of shark spatial density, gaps between consecutive dates in the raw tracking data were interpolated to one position per day. Long temporal gaps without tag-reported location data in a reconstructed track can result in extensive interpolated movements driven by the underlying random walk model rather than the movement pattern of the shark¹¹. Although the frequency of long temporal gaps without data (> 20 days) in our dataset was low (Supplementary Table 15), any tracks with gaps without data that exceeded 20 days were nonetheless split into segments before interpolation, thus avoiding the inclusion of unrepresentative interpolated location estimates⁵. Similarly, location estimates derived for periods without data that exceeded 20 days were also discarded from TOPP data¹¹.

To account for biases in spatial location density associated with (i) variable track lengths and (ii) shorter tracks near the tagging location, we broadly followed a previously published basic time weighting procedure¹¹. In this study, each daily location estimate of an individual was weighted by the inverse of the number of all individuals with location estimates for the same relative day of their track:

$$w_{it} = 1/n_t \text{ for } i \in I \quad (1)$$

in which w_{it} is the weight for the t th location estimate of the track of the i th individual, n_t is the number of total individuals with a t th location estimate, and I is the set of individuals of all species. We calculated weights for all individuals, irrespective of species, to estimate the global relative spatial density of pelagic sharks (that is, Fig. 2a). Periods with gaps without data that exceeded 20 days were not included when weighting the locations. To minimize bias in estimates of spatial density patterns when sample sizes were lower, a modified version of a previously published weighting procedure¹¹ was implemented, such that location weights after a threshold day of the relative track were fixed equal to the weight on the day corresponding to the 85th percentile of track lengths. Under this weighting scheme, individual location estimates closer to the tagging location tended to receive a lower weight than later locations as—owing to tag failure—transmissions of satellite locations are more likely earlier in the track of an individual shark. Therefore, longer tracks received a higher total weight than shorter tracks because of the lower number of long tracks and, consequently, the lower value of n_t towards the end of the track. Calculated spatial densities were therefore more representative of the actual distributions, and less affected by tag loss, failure or a spatial bias towards deployment location.

The weights for all individuals (equation (1)) were normalized so that they summed to unity. Therefore, within the study area, all individuals contributed equally to the described global spatial density patterns:

$$D_{it} = \sum_{i \in I} \sum_{t=1}^T w_{it}$$

in which D_{it} is the relative density contribution of the t th location estimate for individual i , and T_i is the number of location estimates for individual i . The relative density contributions for all location estimates for all individuals (D_{it}) were then summed within each grid cell of the study area for each month of a relative year, which gave 12 spatial relative density maps to compare with monthly longline-fishing effort. The mean annual D_{it} per grid cell for a relative year was calculated from the 12 monthly relative densities per grid cell to provide the global relative density of tracked sharks mapped in Fig. 2a. Hammerhead sharks (3 species) and mako sharks (2 species) were analysed as taxa groups: *Sphyrna* spp. and *Isurus* spp., respectively. The spatial coverage of $1^\circ \times 1^\circ$ grid cells occupied by sharks per ocean region was between 53% (eastern Pacific Ocean) and 25% (Oceania) of total grid cells (Supplementary Table 15). Spatial relative densities of locations were also calculated for each of the ten most data-rich species separately at a $1^\circ \times 1^\circ$ grid-cell size (Extended Data Figs. 6, 7). We followed the same procedure as for the all-species spatial density calculation, but instead weighted by the inverse of the number of total individuals of a single species on the same relative day of their track, and with the weights for each species normalized to one.

To examine how the broad spatial distribution of sharks between years may have changed, we re-calculated the relative density contributions for all location estimates for all individuals (D_{it}) together within each of eight consecutive two-year classes starting in 2002 (Extended Data Fig. 8). Each daily location within a class was weighted by the inverse of the number of individuals with location estimates for the same relative day of the two years (for example, 1 January 2012 is the relative day 1 of all tracks in each of two years that were active on that date). Similar to the weighting scheme applied to the main data, periods with gaps without data that exceeded 20 days were not included when weighting the locations. After the 85th percentile of the track length, daily weights were fixed as for the all-species spatial density. Total weights for all individuals within each two-year class were normalized to one. In addition, owing to a mismatch in the years of data availability between sharks and fishing vessels, exposure risk (overlap and FEI) was

re-calculated for the period between 2012 and 2016, which was common to both sharks and longline-fishing vessels (Extended Data Fig. 9). Relative density of all individuals (D_{it}) was re-calculated on the basis of the weighting scheme, considering only individuals with tracks that were within the 2012–2016 period.

Fishing vessel geolocation data. The AIS was developed as a vessel safety and anti-collision system with global coverage, rather than to track fishing vessels for fishery management purposes^{19–23}. However, its global coverage of the locations of many thousands of ships through time enables the analysis of the distribution of fishing effort^{19–23}. Here, fishing-effort (hours of fishing) data, gridded at 0.01° by flag state, and estimated gear type were obtained from Global Fishing Watch (GFW) (available at <https://globalfishingwatch.org/datasets-and-code/fishing-effort/>). GFW used raw AIS vessel tracking data obtained from ORBCOMM via their AIS-enabled satellite constellation (<https://www.orbcomm.com/eu/networks/satellite-ais>) to calculate fishing effort and derive the gridded data, as previously described in detail^{19,20}. In brief, GFW uses two neural network algorithms to categorize different types of fishing gear (for example, drifting longlines or purse seines), in addition to estimating the spatiotemporally resolved locations at which fishing gears were most likely to be deployed by individual vessels^{19,20}. We used the GFW gridded fishing-effort data in the years 2012 to 2016 for all gear types, and for estimated drifting pelagic longlines and purse seines. The GFW gear-type classification algorithms are being continuously refined to correct for acknowledged contamination of some gear types with others in some regions^{19,20} (for example, drifting longlines with bottom-set longlines off New Zealand). For each gear type in this study, we summed the number of hours fishing in a month (expressed as days, in which 24 h of fishing effort = 1 day) within each $1^\circ \times 1^\circ$ grid cell to provide 12 monthly global fishing-effort maps. The mean annual fishing effort per grid cell in a relative year was calculated from the 12 monthly fishing-effort maps. Global distributions of fishing effort for all gear types, longlines and purse seines were mapped separately and overlaid by shark spatial relative density of locations for all individuals (D_{it}) to determine spatial overlap intensity (the fishing effort to which sharks were exposed; see ‘Shark and vessel spatial overlap and effort’). AIS data coverage increased from 2012 to 2016 as more satellite AIS receivers were launched and commenced operation^{19,20}. However, the global spatial distribution of longline-vessel fishing effort was broadly similar across years (Supplementary Fig. 2), and variation in annual maximum fishing effort displayed no increasing trend over time, which indicated that our calculated mean annual fishing effort for 2012–2016 did not overestimate the spatial overlap or fishing effort, but can be considered conservative (Supplementary Fig. 2). To test that the numbers of longline fishing days per grid cell from AIS data were representative of actual fishing effort as measured by the numbers of baited hooks deployed by longline vessels, we correlated fishing days using AIS data from the Atlantic Ocean with International Commission for the Conservation of Atlantic Tunas (ICCAT) observed hook data (downloaded from <https://iccat.int/en/accesingdb.html>). We compared the total number of observed hooks in ICCAT data at a $5^\circ \times 5^\circ$ grid-cell size (the finest spatial resolution for these ICCAT data) with the total number of fishing days in the AIS dataset, also at $5^\circ \times 5^\circ$. To calculate the fishing-effort days from AIS data in each $5^\circ \times 5^\circ$ grid cell, we summed the days in the $1^\circ \times 1^\circ$ cells that fell within each $5^\circ \times 5^\circ$ cell. Data were used from 2015, the most recent year for which we had both ICCAT hook data and comprehensive longline-fishing coverage from AIS data.

Shark and fishing-effort environment modelling. To model shark and fishing vessel distributions in relation to environmental variables, data were extracted from online databases (Supplementary Fig. 3). The environmental variables were selected on the basis of their demonstrated importance in affecting shark occurrence and included: (i) sea water temperature ($^\circ\text{C}$) (abbreviation used in models: sea-surface temperature, SST; temperature at 100 m, TEM_100), which is known to influence the presence of many pelagic shark species^{5,11}; (ii) maximum thermal gradient ($\Delta^\circ\text{C}$ per 100 km) (TGR) influences shark spatial density⁵, and was calculated here on the basis of the SST data and using maximum gradient maps by determining where, for each pixel, a geodetic-distance-corrected maximum thermal gradient was identified; (iii) sea water salinity (in psu) (SAL), an important determinant of habitat use in some sharks^{1,39}; (iv) sea-surface height above geoid (in m) (SSH), which influences shark presence⁵ and catches by fisheries⁶; (v) ocean mixed-layer depth thickness or thermocline depth (in m) (MLD), which affects the foraging behaviour of pelagic sharks⁴⁰; (vi) mass of chlorophyll *a* in sea water (mg m^{-3}) (CHL) as a proxy for productivity, which often characterizes the preferred habitats of sharks^{5,40}; (vii) mole concentration of phytoplankton, expressed as carbon in sea water concentration (mmol m^{-3}) (PHY), as a direct measure of productivity; (viii) net primary production of biomass, expressed as carbon per unit volume in sea water per day ($\text{g m}^{-3} \text{d}^{-1}$) (NPP), which quantifies productivity; and (ix) mole concentration of dissolved molecular oxygen in sea water (mmol m^{-3}) (DO), which can strongly influence shark space use¹. Environmental datasets i–v were downloaded from Copernicus Marine Environment Monitoring Service (CMEMS) Global Ocean Physics Reanalysis product (http://marine.copernicus.eu/services-portfolio/access-to-products/?option=com_csw&view=details

&product_id=GLOBAL_REANALYSIS_PHY_001_025; downloaded November 2017) and datasets vi–ix from CMEMS Global Ocean Biochemistry Hindcast product (http://marine.copernicus.eu/services-portfolio/access-to-products/?option=com_csw&view=details&product_id=GLOBAL_REANALYSIS_BIO_001_029; downloaded November 2017). CMEMS data were available for 2002 to 2014 from the surface to 5,500 m as monthly datasets. Overall averages (2002–2014) were calculated at a $1^\circ \times 1^\circ$ grid-cell resolution for surface and 100-m depth layers (with the exception of SSH and MLD) (Supplementary Fig. 3). Most of these variables and interactions are also considered important for explaining fishing patterns^{5,6}.

We developed and compared a set of generalized additive models (GAMs) with a Gaussian family and an identity link, using the log-transformed relative density of locations of all individual sharks (D_{it}) as the response variable. We used the relative density of sharks rather than presence and absence data, because our main aim was to highlight the areas in which the highest overlap with fishing effort might occur. Because we were interested in identifying areas (grid cells) with the highest overlap, and understanding how general environmental variables might influence shark density in specific locations, we considered the relative density for all 23 shark species combined without considering random effects per species. All environmental variables were standardized (mean-centred and divided by the s.d.) and collinearity was checked before inclusion in the models. Highly skewed environmental variables were logged before standardization: this included most predictors at the surface (except for SAL and SSH), as well as NPP (for sharks only) and TGR at 100 m (TGR_100). All possible combinations of 16 variables were not undertaken, owing to collinearity. Rather, we focused on testing ecologically relevant hypotheses. A description of the general hypothesis tested with each model included in the model set is given in Supplementary Table 7. It was necessary to include models with a reduced number of variables because some variables were collinear, and those variables were included in other models. Because sharks respond to surface and subsurface thermal gradients (which often support higher biological productivity)^{5,6,11,40}, we tested for interactions between MLD and SST, CHL and MLD at 100 m (MLD_100), CHL at 100 m (CHL_100) and TEM at 100 m (TEM_100), MLD and TGR at the surface, MLD and CHL_100, CHL_100 and TEM_100, and between SAL and TEM_100.

A GAM with a Tweedie distribution and log-link function provided the best modelling approach for the fishing effort data (including zeros in grid cells), as this distribution includes a family of probability distributions including normal, gamma, Poisson and compound Poisson–gamma. We considered two response variables separately: fishing effort (days of fishing per grid cell) of all AIS-equipped fishing vessels, and fishing effort of AIS-equipped longline-fishing vessels only. We did not consider presence and absence data, because our aim was to understand how environment influenced variations in fishing effort. In our model set, we included different combinations of a total of the same 16 explanatory environmental variables used for shark density modelling (Supplementary Table 7), and also a null (all terms equal to zero), intercept-only model. The dimension basis for all terms was limited to 5 (that is, $k = 5$) to assist controlling for overfitting⁴¹. We then used the Akaike’s information criterion (AIC)⁴² to compare the models in the model set for all sharks and fishing vessels. We assessed the relative strength of evidence for each model using the weights of AIC, and the goodness-of-fit of each model by calculating the percentage of deviance explained (%DE). All models were implemented in R using the mgcv package⁴³.

Shark and vessel spatial overlap and effort. The spatial overlap (in per cent) between an individual tracked shark and fishing effort was calculated as the number of grid cells that sharks and fishing effort (in days) occurred in the same $1^\circ \times 1^\circ$ grid cells in an average month, as a function of all shark grid cells occupied and standardized for shark track length. This was summarized as spatial overlap (%) = $100(n_o/n_c)$, in which n_o is the number of grid cells occupied by an individual tracked shark that overlap with grid cells with fishing effort and n_c is the total number of grid cells occupied by an individual tracked shark. The mean monthly spatial overlap of an individual shark was determined from monthly spatial overlap values, and the mean monthly spatial overlap per species was calculated by averaging the mean monthly individual spatial overlap values across all individuals of a species within each ocean region. A fixed $1^\circ \times 1^\circ$ geographical grid cell (in which $1^\circ = 110.6$ km) was chosen, because it is the approximate length of high-seas longlines (that is, 100-km long with an average of 1,200 baited hooks⁵ that attract fish over long distances^{19,20}); because it was similar to the broad light-based geolocation error field of PSAT tags ($n = 615$ sharks; 37% of the total tracks) after SSM processing that we used here, generally shown to be about $0.4\text{--}1.5^\circ$ latitude (about $45\text{--}167$ km)^{32,44–46}; and because it exceeded the upper 95% confidence intervals of the mean daily movement distances of the widest-ranging sharks that we tracked (Supplementary Table 16). In addition, the $1^\circ \times 1^\circ$ grid-cell size was suitable to reduce the effects of gaps in AIS coverage that, at smaller grid sizes, could potentially result in substantial unrecorded fishing effort per grid cell^{19–23}. To examine the effect of grid-cell size on estimates of spatial overlap^{19,20,47}, we calculated the

overlap of all sharks tracked with ARGOS transmitters, from which locations estimated from SSMs were fitted to ARGOS observations (for example, 2.4–5.5-km spatial accuracy⁴⁸), with longline-fishing effort at 2° × 2°, 1° × 1°, 0.75° × 0.75°, 0.50° × 0.50°, 0.25° × 0.25° and 0.10° × 0.10° grid-cell sizes (Extended Data Fig. 4, Supplementary Fig. 4).

An estimate of the exposure of an individual shark to fishing effort within each grid cell occupied during its observed track was termed the FEI and calculated as:

$$FEI = \frac{\sum_{i=1}^n f_i d_i}{n} \quad (3)$$

Here FEI pertains to an individual shark per month in a given year. The term f_i is the fishing effort (number of vessel days) in grid cell i occupied by a shark during its track; d_i is the relative density contribution for all location estimates for an individual shark summed in grid cell i of its track (that is, location estimates of an individual shark were weighted by the inverse of the number of total individual sharks of a single species on the same relative day of their track, and with the weights for each species normalized to one; see ‘Spatial density analysis’); and n is the number of grid cells occupied by an individual shark during its track in a given month of a given year. Individual mean FEI was calculated for an individual shark by averaging the monthly FEI values of an individual shark through time (over the duration of its observed track in monthly steps). To estimate the typical exposure within a species, individual-species mean FEI was calculated by averaging individual-shark mean FEI values for that species within each ocean region (Figs. 3, 4).

To map the mean monthly spatial variation in overlap and fishing effort (fishing exposure) within the space used by sharks (Fig. 2c), we calculated the product of D_{it} and f_i in each grid cell in each month of a relative year across individual sharks (regardless of species), and averaged across the 12 months within each grid cell. In addition, for comparing temporally matched shark–vessel spatial overlap and fishing effort in 2012–2016, we repeated the calculation above but including only those individuals (species) present within these years by multiplication of f_i with the re-calculated D_{it} for those years only (see ‘Spatial density analysis’ for details).

To test for differences in the exposure risk of sharks to fishing activity between different species within the general fishing areas designated by the FAO (Extended Data Fig. 1c), we undertook statistical analysis of exposure risk calculated for each shark as the product of the mean monthly spatial overlap and mean monthly fishing effort. Because the data were not normal (Shapiro–Wilk normality test, $P < 0.05$), a Kruskal–Wallis test was performed (with pairwise Wilcoxon rank-sum tests as a post hoc test). Because of differences in the number of tagged individuals per species, groups of >25 sharks per species were randomly selected and the Kruskal–Wallis test performed. This procedure was repeated 1,000 times and the percentage of times that significant differences were observed was recorded. Species with fewer than 25 individuals tracked were removed from the analysis. Given the lower numbers of sharks tracked in the southwest Indian Ocean and Oceania regions (Supplementary Table 14), statistical tests were restricted to the North Atlantic Ocean and eastern Pacific Ocean regions. In the Atlantic Ocean, selected species were *P. glauca* ($n = 152$), *Isurus* spp. ($n = 120$), *G. cuvier* ($n = 131$), *C. carcharias* ($n = 26$), *C. longimanus* ($n = 99$), *L. nasus* ($n = 46$), *C. leucas* ($n = 38$) and *Sphyrna* spp. ($n = 40$); in the Pacific Ocean, species were *P. glauca* ($n = 112$), *I. oxyrinchus* ($n = 113$), *L. ditropis* ($n = 172$), *R. typus* ($n = 77$) and *C. carcharias* ($n = 59$).

Shark landings. Mean annual pelagic shark landings (t) by species/taxa groups were obtained from the FAO database (www.FAO.org/fishery/statistics/global-capture-production/query/en; downloaded September 2018) and related to the median monthly FEI of each species or taxa group. Landings reported for the North Atlantic Ocean (northwest, northeast, western central and eastern central Atlantic Ocean) between 2007 and 2016 were used in the analysis, because these spanned the main period during which most sharks were tracked (70% between 2007–2017) and longline-fishing effort was monitored (2012–2016). Data were extracted for eight species or taxa groups that are regularly caught by shelf and/or high-seas fisheries in the North Atlantic Ocean, the region in which most tags were deployed. The species and taxa groups were *P. glauca*, *I. oxyrinchus*, *C. longimanus*, *C. leucas*, *L. nasus*, *G. cuvier*, *C. carcharias* and hammerheads (*Sphyrna* spp., comprising *S. lewini*, *S. mokarran* and *S. zygaena*).

Reporting summary. Further information on research design is available in the Nature Research Reporting Summary linked to this paper.

Data availability

The source code used to undertake analyses and to prepare figures, in addition to the derived data underlying Fig. 2 maps (shark relative spatial density, longline-fishing effort and shark–longline-fishing overlap and FEI) and Fig. 3 plots (spatial overlap and FEI) is freely available on GitHub (<https://github.com/GlobalSharkMovement/GlobalSpatialRisk>).

32. Wilson, R. P., Ducamp, J.-J., Rees, W. G., Culik, B. M. & Niekamp, K. in *Wildlife Telemetry* (eds Priede, I. G. & Swift, S. M.) 131–134 (Ellis Horwood, Chichester, 1992).
33. Delong, R. L., Stewart, B. S. & Hill, R. D. Documenting migrations of northern elephant seals using day length. *Mar. Mamm. Sci.* **8**, 155–159 (1992).
34. Lam, C., Nielsen, A. & Sibert, J. Improving light and temperature based geolocation by unscented Kalman filtering. *Fish. Res.* **91**, 15–25 (2008).
35. Johnson, D. S., London, J. M., Lea, M.-A. & Durban, J. W. Continuous-time correlated random walk model for animal telemetry data. *Ecology* **89**, 1208–1215 (2008).
36. Sippel, T., Holdsworth, J., Dennis, T. & Montgomery, J. Investigating behaviour and population dynamics of striped marlin (*Kajikia audax*) from the southwest Pacific Ocean with satellite tags. *PLoS ONE* **6**, e21087 (2011).
37. Jonsen, I. D., Flemming, J. M. & Myers, A. E. Robust state-space modeling of animal movement data. *Ecology* **86**, 2874–2880 (2005).
38. Lunn, D. J., Thomas, A., Best, N. & Spiegelhalter, D. WinBUGS – a Bayesian modelling framework: 472 concepts, structure, and extensibility. *Stat. Comput.* **10**, 325–337 (2000).
39. Ward-Paige, C. A., Britten, G. L., Bethea, D. M. & Carlson, J. K. Characterizing and predicting essential habitat features for juvenile coastal sharks. *Mar. Ecol. (Berl.)* **36**, 419–431 (2014).
40. Queiroz, N. et al. Convergent foraging tactics of marine predators with different feeding strategies across heterogeneous ocean environments. *Front. Mar. Sci.* **4**, 239 (2017).
41. Fisher, R., Wilson, S. K., Sin, T. M., Lee, A. C. & Langlois, T. J. A simple function for full-subsets multiple regression in ecology with *R*. *Ecol. Evol.* **8**, 6104–6113 (2018).
42. Burnham, K. P. & Anderson, D. R. Multimodel inference: understanding AIC and BIC in model selection. *Sociol. Methods Res.* **33**, 261–304 (2004).
43. Wood, S. N. Fast stable restricted maximum likelihood and marginal likelihood estimation of semiparametric generalized linear models. *J. R. Stat. Soc. Series B Stat. Methodol.* **73**, 3–36 (2011).
44. Teo, S. L. H. et al. Validation of geolocation estimates based on light level and sea surface temperature from electronic tags. *Mar. Ecol. Prog. Ser.* **283**, 81–98 (2004).
45. Nielsen, A., Bigelow, K. A., Musyl, M. K. & Sibert, J. R. Improving light-based geolocation by including sea surface temperature. *Fish. Oceanogr.* **15**, 314–325 (2006).
46. Sims, D. W. in *Sharks and Their Relatives II – Biodiversity, Adaptive Physiology and Conservation* (eds Carrier, J. C. et al.) 351–392 (CRC, Boca Raton, 2010).
47. Amoroso, R. O., Parma, A. M., Pitcher, C. R., McConaughy, R. A. & Jennings, S. Comment on “Tracking the global footprint of fisheries”. *Science* **361**, eaat6713 (2018).
48. Lowther, A. D., Lydersen, C., Fedak, M. A., Lovell, P. & Kovacs, K. M. The Argos-CLS Kalman filter: error structures and state-space modelling relative to Fastloc GPS data. *PLoS ONE* **10**, e0124754 (2015).

Acknowledgements We thank all who were involved in fieldwork and data collection (full details are given in the Supplementary Information). Data analysis was funded in part by the Marine Biological Association (MBA) and the UK Natural Environment Research Council (NERC) (NE/R00997X/1) (to D.W.S.) with additional research support from the Save Our Seas Foundation and the NERC Oceans 2025 Strategic Research Programme, in which D.W.S. was a principal investigator. D.W.S. was supported by an MBA Senior Research Fellowship, N.Q. by European Regional Development Fund (FEDER) via the Programa Operacional Competitividade e Internacionalização (COMPETE), National Funds via Fundação para a Ciência e a Tecnologia (FCT) under PTDC/MAR/100345/2008 and COMPETE FCOMP-01-0124-FEDER-010580 (to N.Q. and D.W.S.), and Norte Portugal Regional Operational Programme (NORTE 2020) under the PORTUGAL 2020 Partnership Agreement through the European Regional Development Fund (ERDF) in project MarInfo (NORTE-01-0145-FEDER-000031). Additional support was provided by an FCT Investigator Fellowship IF/01611/2013 (N.Q.), FCT Doctoral Fellowship PD/BD/52603/2014 (M.V.), PTDC/MAR-BIO/4458/2012, Xunta de Galicia - Isabel Barreto Program 2009-2012 (G.M.), Australian Research Council (ARC) grant DE170100841 and operational funds from the Australian Institute for Marine Science (AIMS) (both to A.M.M.S.). We thank Stanford University, the Tagging of Pacific Predators programme and Global Fishing Watch for making data freely available. We thank M. Dando for creating the shark images. This research contributes to the Global Shark Movement Project (GSMP).

Author contributions The GSMP is coordinated by D.W.S. N.Q. and D.W.S. conceived the study, N.Q., N. E. Humphries and D.W.S. designed the study, and all authors contributed to animal tagging, fieldwork, data collection and/or contribution of tools (full details are given in the Supplementary Information). N.Q., N. E. Humphries, A.C., M.V., I.d.C., A.M.M.S., L.L.S., S.J.S. and D.W.S. analysed the collated data. D.W.S. drafted the paper with contributions from N.Q., N. E. Humphries and A.M.M.S. All authors contributed to subsequent drafts.

Competing interests The authors declare no competing interests.

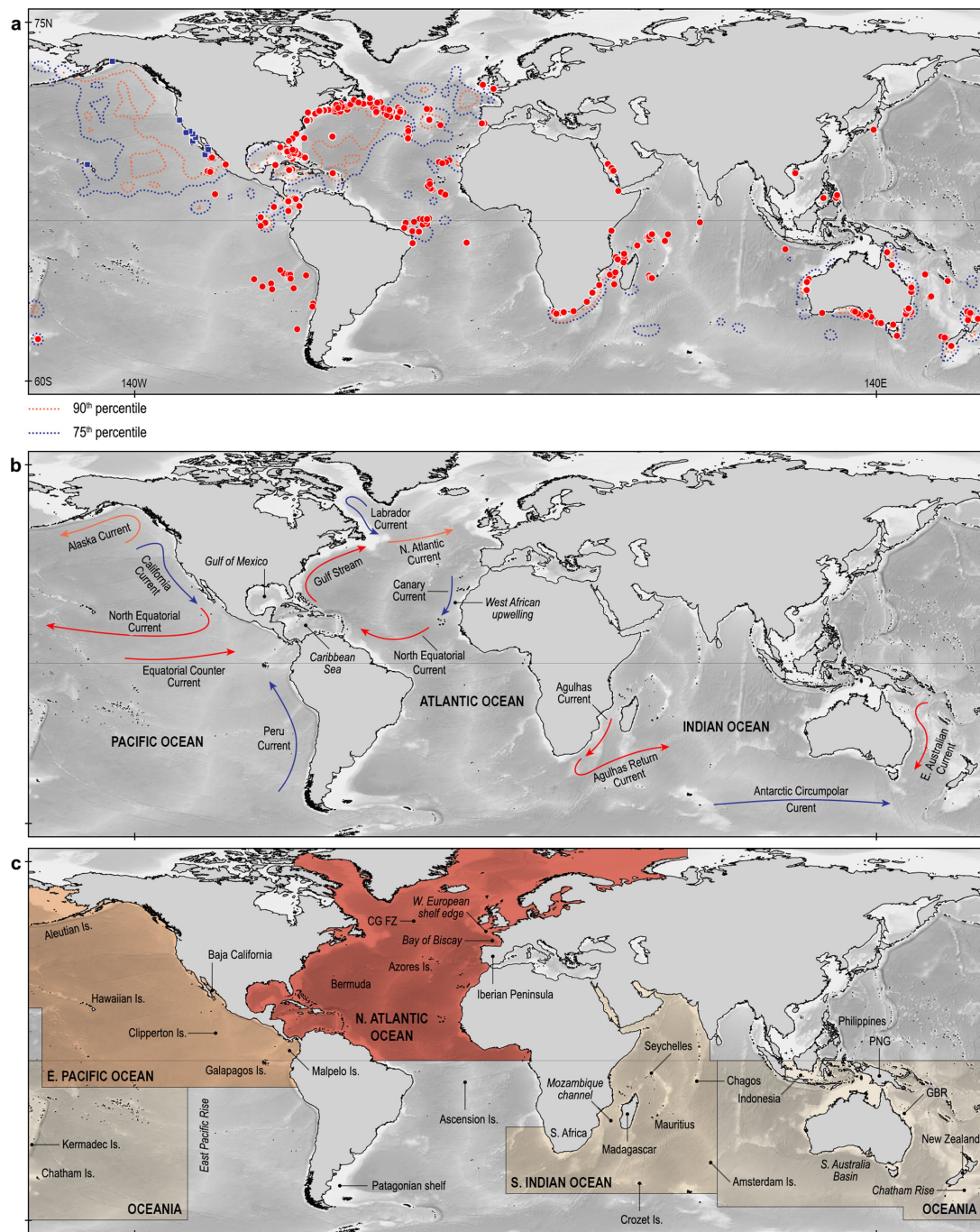
Additional information

Supplementary information is available for this paper at <https://doi.org/10.1038/s41586-019-1444-4>.

Correspondence and requests for materials should be addressed to D.W.S.

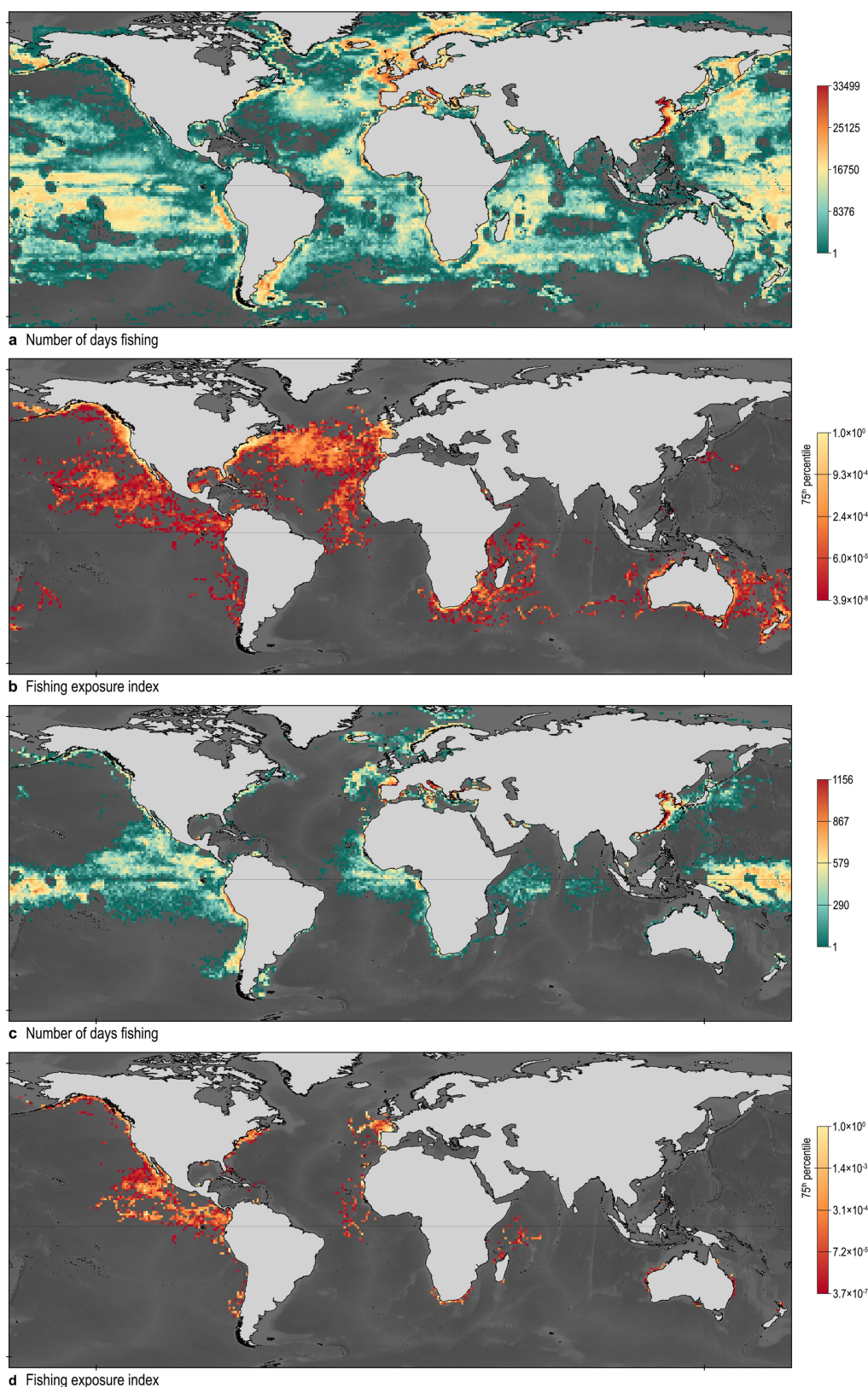
Peer review information Nature thanks Julia Baum, Brendan Godley, Ian Jonsen and David Kroodsma for their contribution to the peer review of this work.

Reprints and permissions information is available at <http://www.nature.com/reprints>.



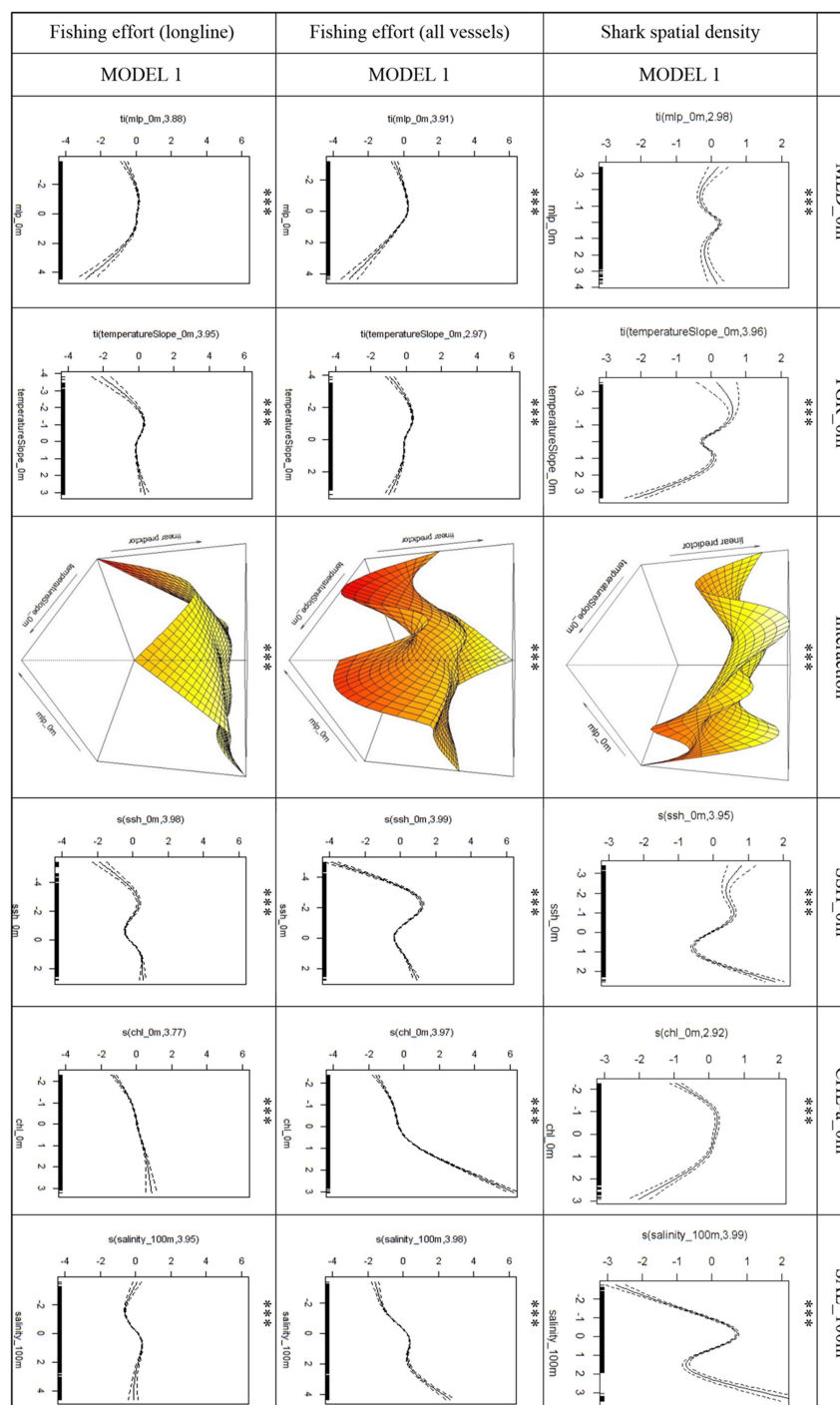
Extended Data Fig. 1 | The locations of shark tag deployment sites in relation to shark space-use hotspots, ocean currents, physical features and fishing areas. a, Red circles denote the locations in which satellite transmitters were attached and sharks released, and blue squares in the eastern Pacific Ocean denote the annual median deployment locations of tags by the TOPP program¹¹. Shark space-use hotspots are shown as the 75th (blue dotted lines) and 90th percentiles (red dotted lines) of the mean

monthly relative density of estimated shark locations within $1^\circ \times 1^\circ$ grid cells given in Fig. 2a. **b, c,** Schematic maps of major ocean currents (**b**) and physical features overlaid on FAO fishing areas (**c**). Coloured arrows in **b** denote the thermal regime of currents, with warmer colours indicating higher water temperatures. CGFZ, Charlie Gibbs Fracture Zone; GBR, Great Barrier Reef; PNG, Papua New Guinea.



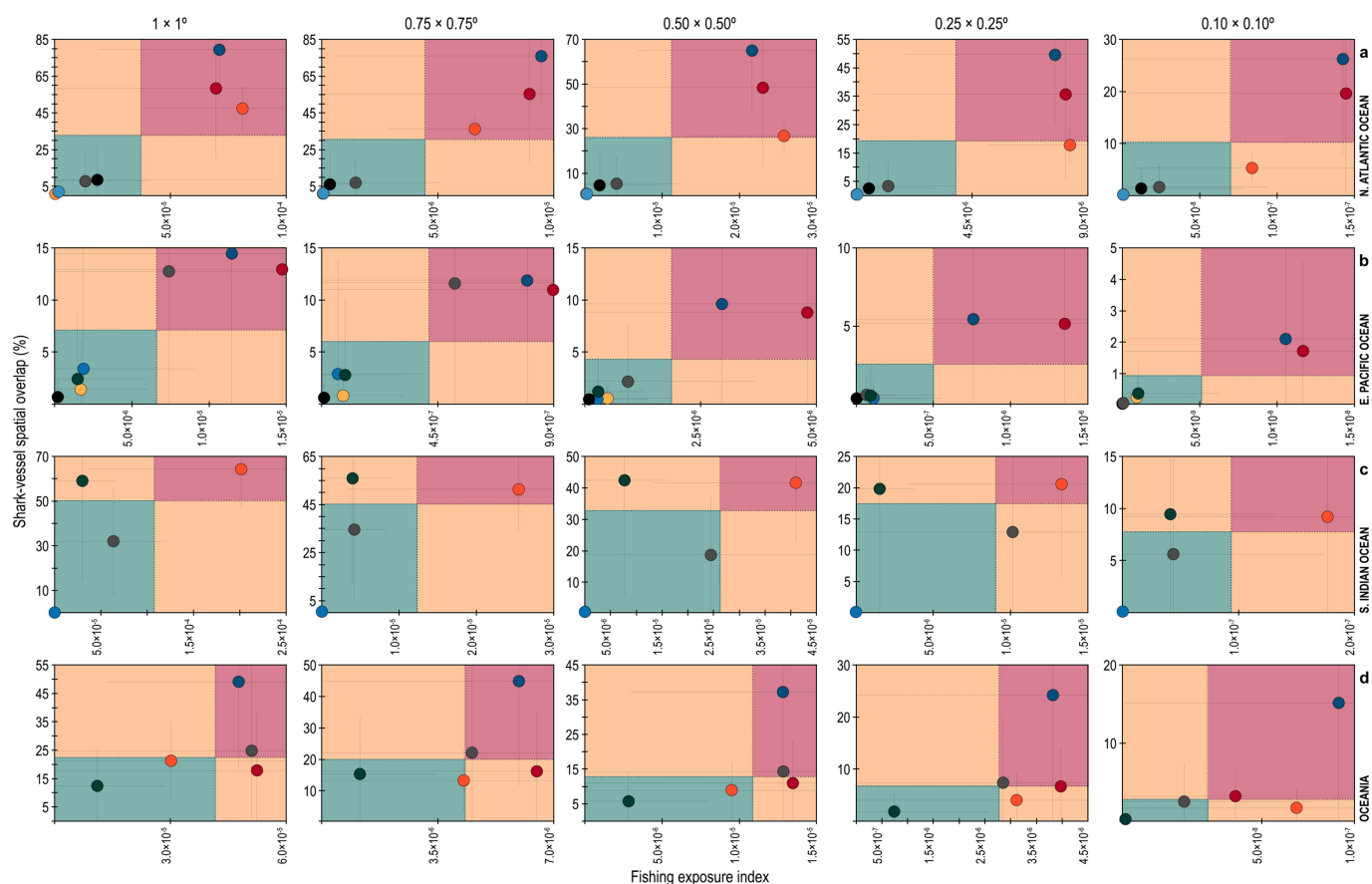
Extended Data Fig. 2 | Spatial distribution of fishing vessels and overlap with sharks. **a**, Distribution of the fishing effort of AIS-tracked vessels (mean annual days spent per grid cell) between 2012 and 2016 (Methods). **b**, Distribution of the mean monthly overlap and level of the fishing effort for all vessels (in days) to which sharks were exposed in overlapping areas, for all species, within $1^\circ \times 1^\circ$ grid cells (Methods). Spatial overlap hotspots were defined as $1^\circ \times 1^\circ$ grid cells with ≥ 75 th percentile of mean FEI. Note that the overlap pattern of sharks and all mapped AIS-equipped fishing

vessels is similar to that determined for sharks and longline-fishing vessels in Fig. 2c. **c**, Distribution of the fishing effort of AIS-equipped purse-seine vessels, using mean annual days spent per grid cell between 2012 and 2016 (Methods). **d**, Distribution of the mean monthly overlap and level of fishing effort of purse-seine vessel (in days) to which sharks were exposed in overlapping areas, for all species, within $1^\circ \times 1^\circ$ grid cells (Methods). Spatial overlap hotspots were defined as $1^\circ \times 1^\circ$ grid cells with ≥ 75 th percentile of mean FEI.



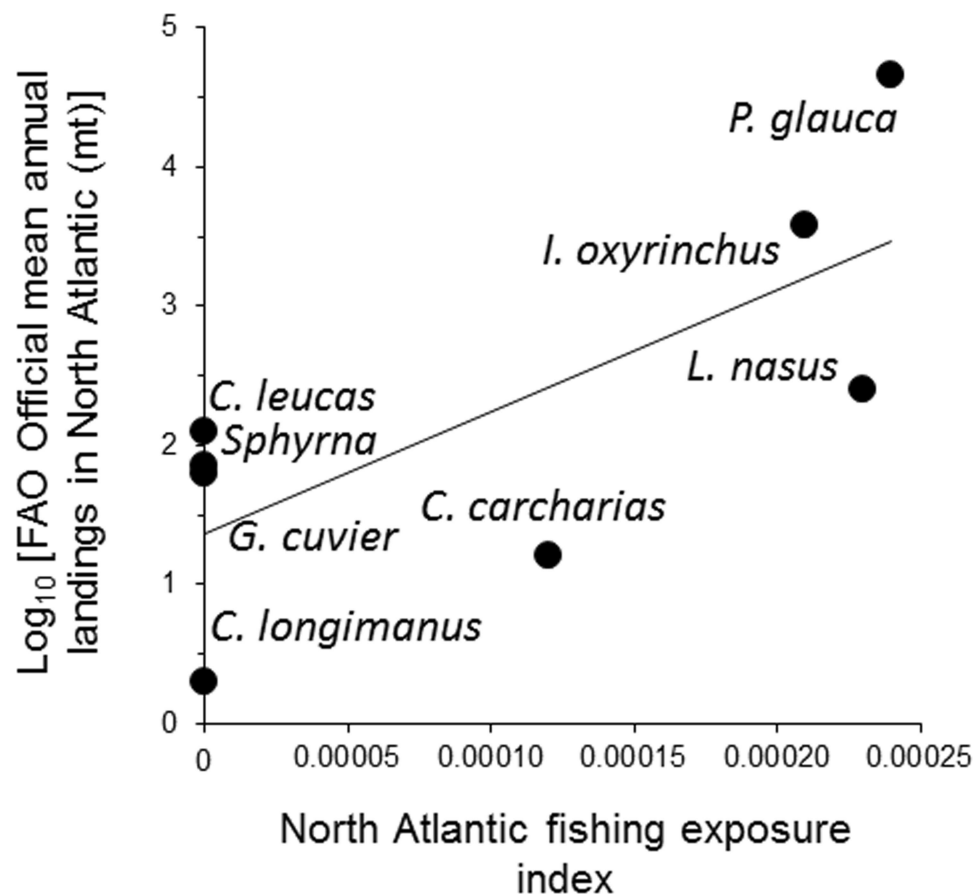
Extended Data Fig. 3 | Environmental modelling results. Estimated relationships between mean monthly relative density of all sharks (top) and fishing effort of all AIS-equipped fishing vessels (middle) and longline-fishing vessels only (bottom), with all environmental variables in

the highest-ranked model (model 1) of the GAM tested. The third column shows the interaction results between the two variables described in the first and second columns. Asterisks indicate the significance level for each smooth term included in the GAM; *** $P < 0.001$.



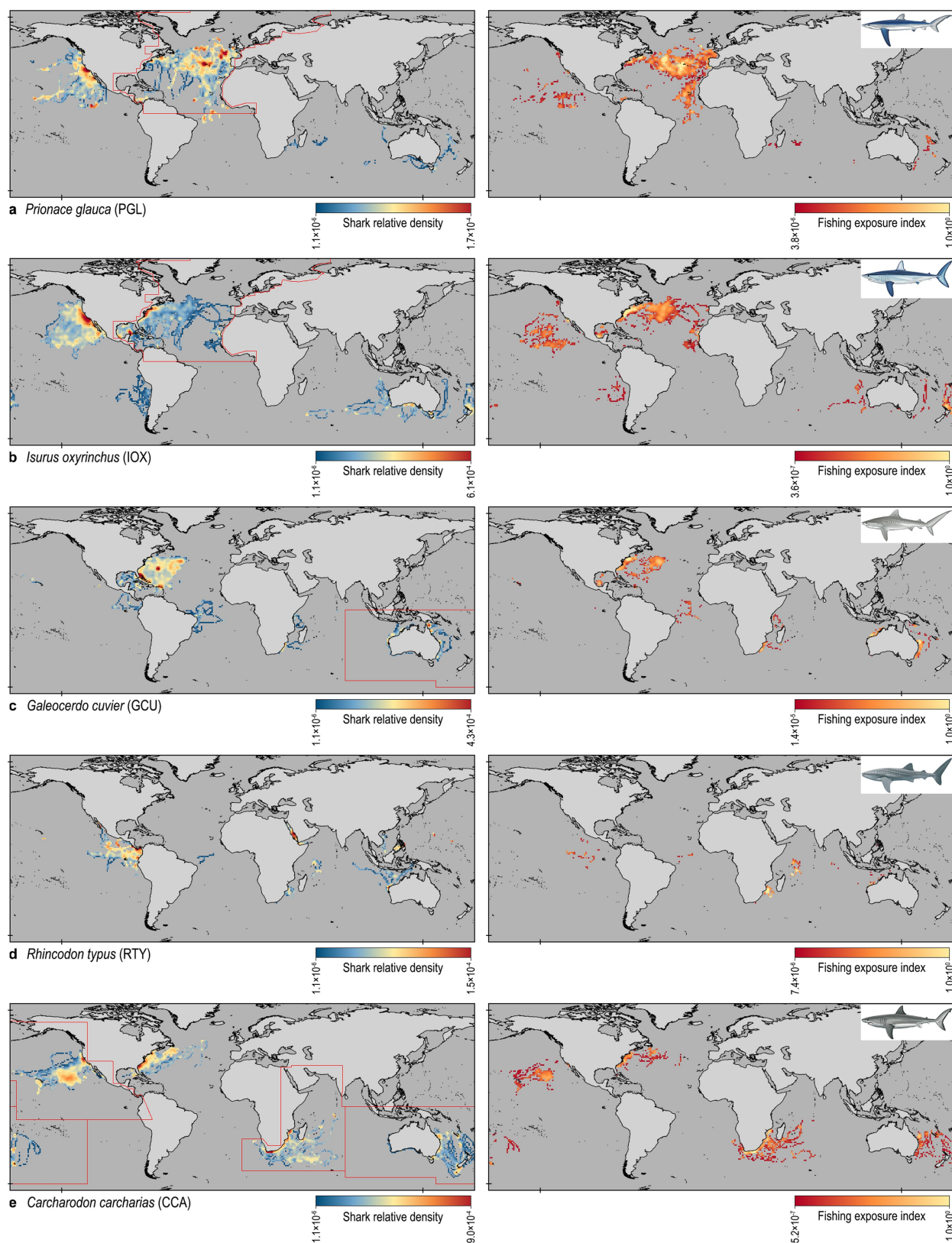
Extended Data Fig. 4 | Effect of grid-cell size on risk-exposure patterns of sharks to longline fisheries. a–d, North Atlantic Ocean (a), east Pacific Ocean (b), southwest Indian Ocean (c) and Oceania (d). Note that, regardless of the grid-cell size at which the individual-species mean spatial overlap and FEI were calculated, the species that occur in the highest- (red) and the lowest-risk zones (green) remain notably conserved, which indicates a general pattern that is not dependent on the scale at which

these data were analysed. Shark-species identification codes corresponding to marker colours are given in Fig. 3. In addition, for the North Atlantic (a), hammerhead sharks (*Sphyrna* spp.) are represented by a black circle and oceanic whitetip sharks (*C. longimanus*) by a light blue circle; for the eastern Pacific Ocean (b), the salmon shark (*L. ditropis*) is represented by a light orange circle. Error bars are ± 1 s.d. An additional comparison of $2^\circ \times 2^\circ$ with $1^\circ \times 1^\circ$ grid-cell size is given in Supplementary Fig. 4.



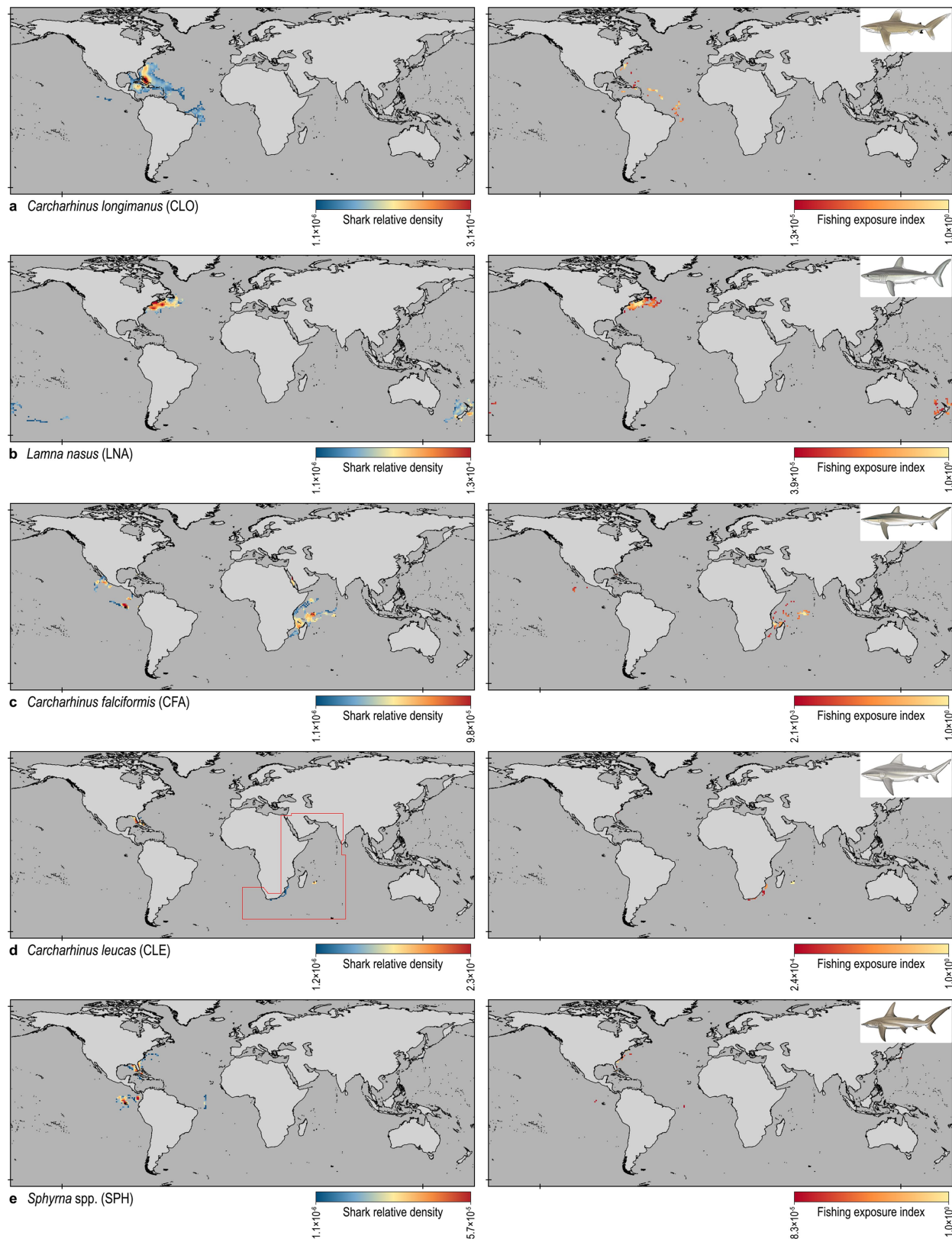
Extended Data Fig. 5 | Relationship between shark landings by fisheries in the North Atlantic Ocean, and shark density–longline FEI. Plot showing shark landings from the North Atlantic Ocean (mean, 2007–2016), extracted from the FAO total-capture production database, was dependent upon the longline-fishing effort in the North Atlantic Ocean, as estimated with the individual-species FEI (70% of sharks tracked, 2007–2017; AIS data, 2012–2016) (Methods).

Using linear regression, we tested the null hypothesis (H_0) that $\beta = 0$ after normalizing landings (in metric tonnes) by log-transformation and for median FEI per species. Regression analysis gave the equation: $\log(\text{landings}) = 1.364 + 8,732 \text{ FEI}$, with a regression coefficient (b) standard error of 3,369. We computed $r^2 = 0.45$, $F = 6.72$ and $F_{0.05(1),1.7} = 5.59$, therefore rejecting H_0 at the 5% level of significance with $P < 0.05$.



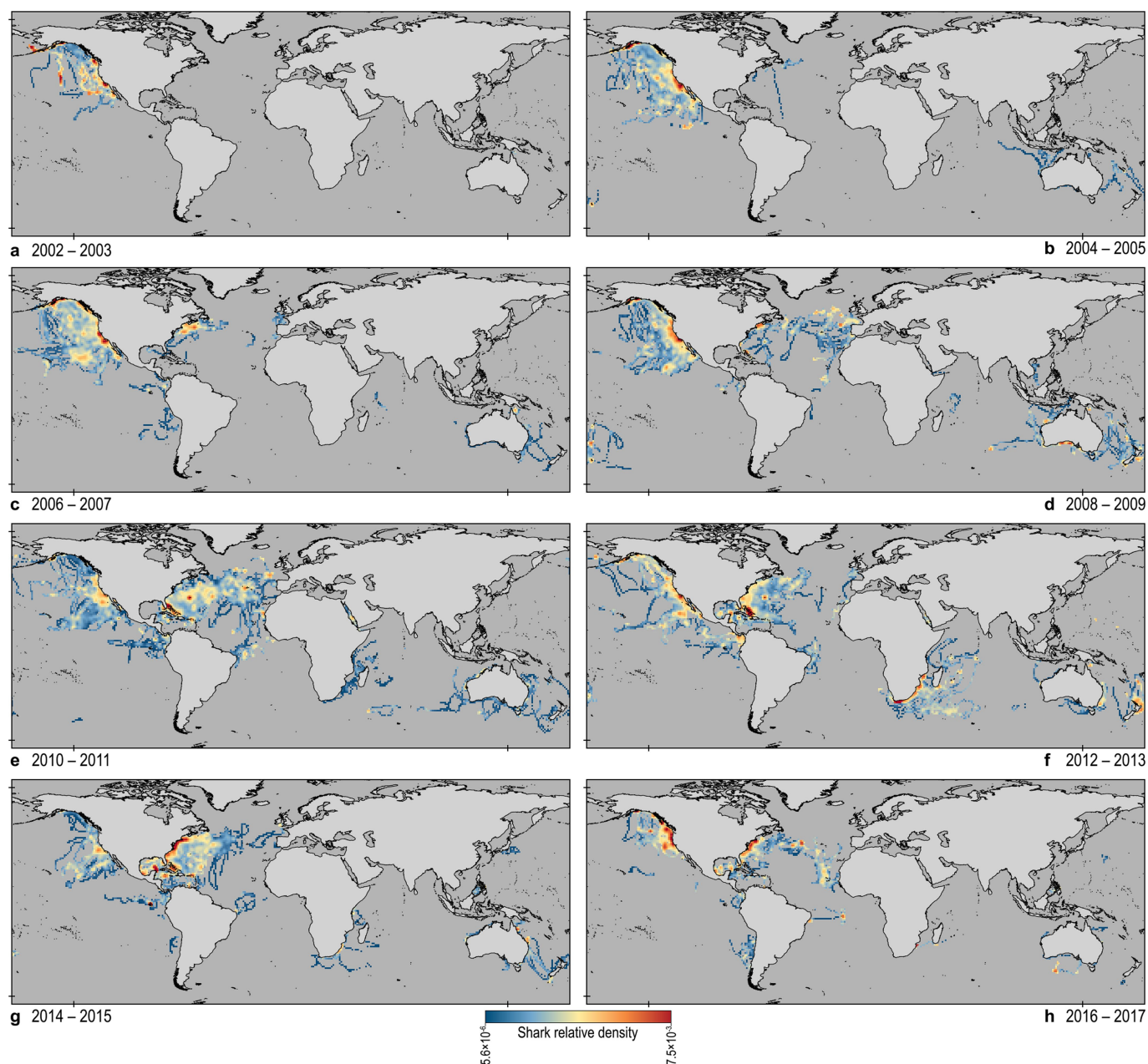
Extended Data Fig. 6 | Relative density and spatial overlap distributions for the five most data-rich individual shark species that occur in multiple oceans. a–e, Mean monthly relative density of shark species (left) tracked in 2002–2017 in comparison with species-mean FEI per grid

cell (right) for the 5 most data-rich species or taxa groups that occur in multiple oceans: blue shark (a), shortfin mako shark (b), tiger shark (c), whale shark (d) and white shark (e). Red boxes denote areas shown in Fig. 3. Shark images created by M. Dando.



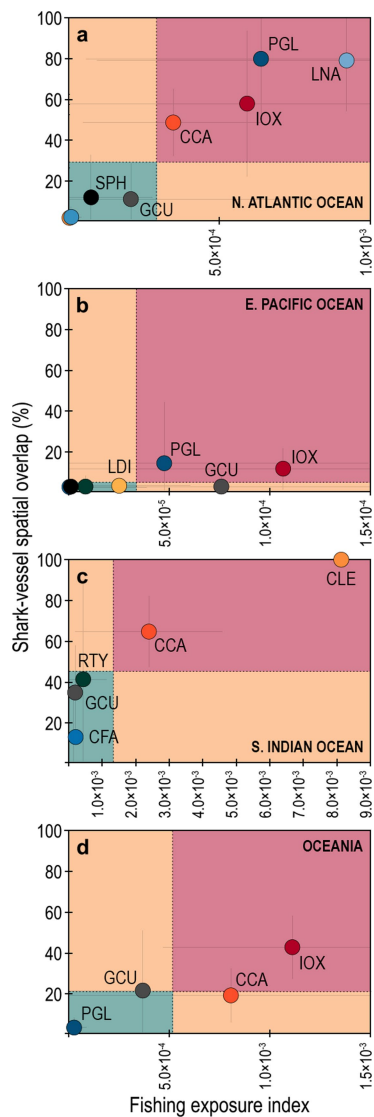
Extended Data Fig. 7 | Relative density and spatial overlap distributions for the sixth to tenth most data-rich individual shark species that occur in multiple oceans. Mean monthly relative density of shark species (left) tracked in 2002–2017 in comparison with species-mean FEI per grid cell (right) for the sixth to tenth most data-rich species or taxa groups that

occur in multiple oceans: oceanic whitetip shark (a), porbeagle shark (b), silky shark (c), bull shark (d) and hammerhead sharks (d) (comprising the scalloped hammerhead shark, the great hammerhead shark and the smooth hammerhead shark). Shark images created by M. Dando. CLO, *Carcharhinus longimanus*; SPH, *Sphyrna* spp.

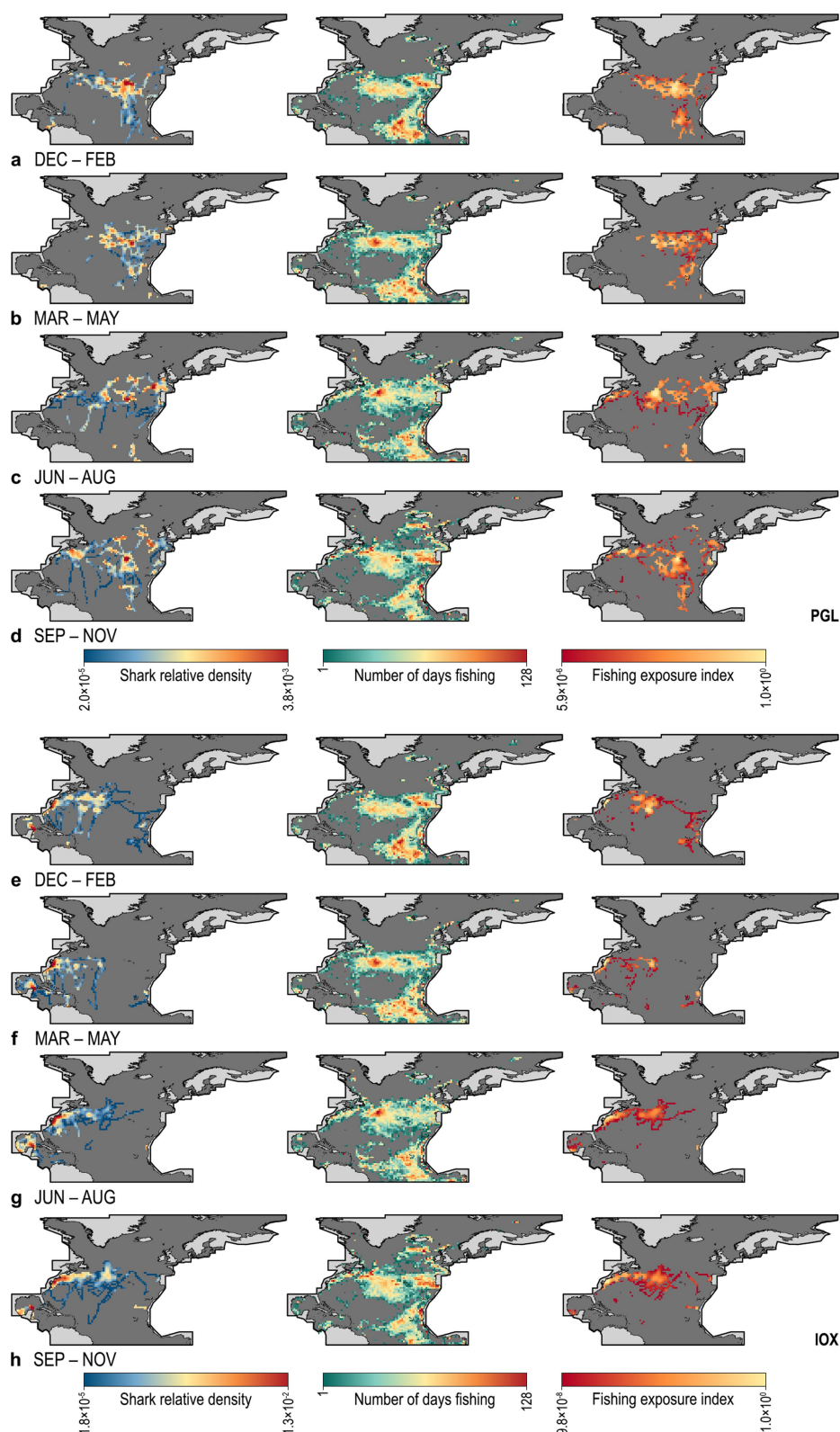


Extended Data Fig. 8 | Between-years patterns in global spatial density of pelagic sharks. Mean monthly spatial density was calculated for each two-year period across species. We used consecutive two-year groups to reduce gaps in coverage. Note that there were broad-scale shark tracks in the east Pacific Ocean in all eight two-year periods (2002–2003 to 2016–2017), in the North Atlantic Ocean between 2006–2007 and

2016–2017, in the southwest Indian Ocean in 2010–2011 to 2014–2015, and in Oceania between 2004–2005 and 2014–2015. This indicates that temporal consistency of shark tracks was present within the ocean regions studied, which suggests that the spatial hotspots that we identified are more likely to be persistent between years.



Extended Data Fig. 9 | Risk-exposure patterns of sharks from longline fisheries between 2012 and 2016. a–d, North Atlantic Ocean (a), east Pacific Ocean (b), southwest Indian Ocean (c) and Oceania (d). Note that species patterns of exposure to risk in highest-risk (red) and lowest-risk (green) zones in the years 2012–2016, which matched shark density data with longline-fishing effort data from AIS-equipped vessels directly, were very similar to patterns found for shark density (2002–2017) and longline-fishing effort data from AIS-equipped vessels (species-mean FEI) (2012–2016) (shown in Fig. 3), which indicates that there was no important effect of temporally mismatched datasets on the results. LDI, *L. ditropis*.



Extended Data Fig. 10 | Seasonal shifts in sharks, longline-fishing vessels and patterns of overlap with fishing effort. a–h, Mean quarterly relative spatial density of sharks (left), longline-fishing effort (in days) (middle) and mean FEI per grid cell (the fishing effort to which sharks were exposed in overlapped areas) (right) for blue sharks in the

North Atlantic Ocean in December–February (a), March–May (b), June–August (c) and September–November (d), and for shortfin mako sharks in the North Atlantic Ocean in December–February (e), March–May (f), June–August (g) and September–November (h).

Reporting Summary

Nature Research wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Research policies, see [Authors & Referees](#) and the [Editorial Policy Checklist](#).

Statistical parameters

When statistical analyses are reported, confirm that the following items are present in the relevant location (e.g. figure legend, table legend, main text, or Methods section).

n/a Confirmed

- ☐ ☒ The exact sample size (n) for each experimental group/condition, given as a discrete number and unit of measurement
- ☐ ☒ An indication of whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
- ☐ ☒ The statistical test(s) used AND whether they are one- or two-sided
Only common tests should be described solely by name; describe more complex techniques in the Methods section.
- ☐ ☒ A description of all covariates tested
- ☐ ☒ A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
- ☐ ☒ A full description of the statistics including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
- ☐ ☒ For null hypothesis testing, the test statistic (e.g. F , t , r) with confidence intervals, effect sizes, degrees of freedom and P value noted
Give P values as exact values whenever suitable.
- ☐ ☒ For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
- ☒ ☐ For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
- ☒ ☐ Estimates of effect sizes (e.g. Cohen's d , Pearson's r), indicating how they were calculated
- ☐ ☒ Clearly defined error bars
State explicitly what error bars represent (e.g. SD, SE, CI)

Our web collection on [statistics for biologists](#) may be useful.

Software and code

Policy information about [availability of computer code](#)

Data collection

Shark satellite tracking: Raw location data from pop-off satellite archival transmitters (PSATs) processed after ARGOS satellite acquisition using tag manufacturers custom software (Wildlife Computers; Microwave Telemetry; Desert Star Systems) to calculate latitude and longitude. Raw positions processed using a UKFSST state space model (UKFSST R package). Raw ARGOS tag locations after satellite acquisition were processed with a speed filter (in R). For both tag types (PSAT and ARGOS), daily time series of locations estimated using a continuous time correlated random walk (CTCRW) Kalman filter (crawl R package). TOPP tag data was filtered with a Bayesian based state space model using WinBugs (for priors and MCMC sampling see Methods).
Fishing vessel tracking: Processed data were acquired from the Global Fishing Watch. for code and processing details see ref. 19 in paper.

Data analysis

R, Minitab v18 and ArcGIS

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors/reviewers upon request. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Research [guidelines for submitting code & software](#) for further information.

Data

Policy information about [availability of data](#)

All manuscripts must include a [data availability statement](#). This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A list of figures that have associated raw data
- A description of any restrictions on data availability

The source code used to undertake analyses and to prepare figures, in addition to the derived data in spreadsheet form underlying Fig. 2 maps (shark relative spatial density; longline fishing effort; and shark– longline overlap and FEI) and Fig. 3 plots (spatial overlap and FEI) are freely available to download on GitHub (github.com/GlobalSharkMovement/GlobalSpatialRisk). Processed fishing vessel effort data are available to download at <http://globalfishingwatch.org/datasets-and-code/>

Field-specific reporting

Please select the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☐ Life sciences ☐ Behavioural & social sciences ☒ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/authors/policies/ReportingSummary-flat.pdf

Ecological, evolutionary & environmental sciences study design

All studies must disclose on these points even when the disclosure is negative.

Study description	The study describes the distributions of satellite tracked pelagic sharks and fishing vessels across the global oceans and calculates the extent of overlap and fishing effort different shark species are exposed to in space and time. See Methods for time periods of data collection.
Research sample	Movements of individual pelagic sharks were satellite tracked (n = 1804) from 23 threatened species in the Atlantic, Pacific and Indian oceans. Species details including number and locations of tags deployed on each species are given in the paper. Fishing vessels (n > 80,000) were tracked globally using the automatic identification system. These data were downloaded from Global Fishing Watch.
Sampling strategy	Pelagic sharks were captured alive at sea with baited hooks or with purse seines prior to tagging and subsequent release. Some sharks were tagged while free swimming. Tags were fitted externally within a few minutes. Tagging was undertaken by 30 different research groups across many countries with tagging procedures approved by institutional ethical boards and conforming to national regulations.
Data collection	Each research group collected shark track data independently by download from the ARGOS satellite service provider.
Timing and spatial scale	Pelagic sharks were tracked between 2002 and 2017. Details of tag deployments and tracking durations are detailed in the paper.
Data exclusions	Poor quality data were reported for 123 tags (72 ARGOS and 51 PSATs) due to early tag failure, premature tag pop-off and/or a high percentage of locations estimated with high spatial error, e.g. raw computed geolocations over land, all of which resulted in poor state space model fits and hence short or unreliable track reconstructions. These data were excluded.
Reproducibility	No experiments as such were conducted, rather our data are based on satellite tracked movements of individual pelagic sharks and fishing vessels.
Randomization	Randomization procedures were used and are fully described in the Methods and Supplementary Information files.
Blinding	Blinding is not relevant to this type of study because data are based on movements of wild animals and fishing vessels.
Did the study involve field work?	☒ Yes ☐ No

Field work, collection and transport

Field conditions	Tags were deployed on pelagic sharks in the Atlantic, Pacific and Indian Oceans under a range of conditions.
Location	Locations of tagging and subsequent tracks of sharks are detailed in the paper (Fig. 1; Extended Data Fig. 1).
Access and import/export	No collections or import or export of samples was undertaken.
Disturbance	Disturbance to individual shark behaviour was minimised through completion of tagging procedures within a few minutes if captured, or during free swimming. All procedures were approved by institutional and national ethical review committees.

Reporting for specific materials, systems and methods

Materials & experimental systems

n/a	Involvement in the study
☒	☐ Unique biological materials
☒	☐ Antibodies
☒	☐ Eukaryotic cell lines
☒	☐ Palaeontology
☐	☒ Animals and other organisms
☒	☐ Human research participants

Methods

n/a	Involvement in the study
☒	☐ ChIP-seq
☒	☐ Flow cytometry
☒	☐ MRI-based neuroimaging

Animals and other organisms

Policy information about [studies involving animals](#); [ARRIVE guidelines](#) recommended for reporting animal research

Laboratory animals	N/A
Wild animals	Satellite tags were fitted to individuals from 23 species of pelagic shark when captured or free swimming. Detailed information is provided in the Methods. All captured sharks were released after tag attachment. None were killed as part of the study. Tag release locations are given in Extended Data Fig. 1.
Field-collected samples	N/A

TITLE/ AUTHOR	Marine biofilms constitute a bank of hidden microbial diversity and functional potential / Zhang, W., Ding, W., Li, YX. et al.
SOURCE	Nature Communication Volume 10, Article no. 517 (Jan 2019) https://doi.org/10.1038/s41467-019-08463-z (Database: Nature)

ARTICLE

<https://doi.org/10.1038/s41467-019-08463-z>

OPEN

Marine biofilms constitute a bank of hidden microbial diversity and functional potential

Weipeng Zhang¹, Wei Ding¹, Yong-Xin Li ¹, Chunkit Tam¹, Salim Bougouffa ², Ruojun Wang¹, Bite Pei¹, Hoyin Chiang¹, Pokman Leung ¹, Yanhong Lu¹, Jin Sun¹, He Fu³, Vladimir B Bajic ², Hongbin Liu¹, Nicole S. Webster ^{4,5} & Pei-Yuan Qian¹

Recent big data analyses have illuminated marine microbial diversity from a global perspective, focusing on planktonic microorganisms. Here, we analyze 2.5 terabases of newly sequenced datasets and the *Tara* Oceans metagenomes to study the diversity of biofilm-forming marine microorganisms. We identify more than 7,300 biofilm-forming ‘species’ that are undetected in seawater analyses, increasing the known microbial diversity in the oceans by more than 20%, and provide evidence for differentiation across oceanic niches. Generation of a gene distribution profile reveals a functional core across the biofilms, comprised of genes from a variety of microbial phyla that may play roles in stress responses and microbe-microbe interactions. Analysis of 479 genomes reconstructed from the biofilm metagenomes reveals novel biosynthetic gene clusters and CRISPR-Cas systems. Our data highlight the previously underestimated ocean microbial diversity, and allow mining novel microbial lineages and gene resources.

¹Department of Ocean Science and Division of Life Science, Hong Kong University of Science and Technology, Hong Kong, China. ²Computational Bioscience Research Center, King Abdullah University of Science and Technology, Thuwal 23955, Saudi Arabia. ³Department of Marine Sciences, University of Georgia, Athens 30602 GA, USA. ⁴Australian Institute of Marine Science, St Lucia 4067, QLD, Australia. ⁵Australian Centre for Ecogenomics, University of Queensland, St Lucia 4067, QLD, Australia. Correspondence and requests for materials should be addressed to P.-Y.Q. (email: boqianpy@ust.hk)

Microorganisms contribute significantly to the health and resilience of marine ecosystems via their critical roles in biogeochemical cycling¹, their interactions with macroorganisms², and their provision of chemical cues that underpin the recruitment of marine invertebrates³. Recent advances in sequencing technology have facilitated unprecedented analyses of the ocean microbiome. The *Tara* Oceans project analyzed 7.2 terabases of metagenomic data to illuminate microbial diversity from a global perspective and generated a global ocean microbial reference gene catalog (OM-RGC)⁴. The oceanic microbiome exhibits vertical stratification with community composition primarily dependent on seawater temperature⁴. However, most global oceanic surveys have focused on free-living bacterioplankton, despite many aquatic microorganisms colonizing solid surfaces to form multi-species biofilms^{5,6}.

Surface association can have numerous ecological advantages including environmental protection, increased access to nutritional resources and enhanced opportunities for interactions with other organisms⁶. Microbial surface association can also have deleterious outcomes such as biofouling, biocorrosion, and antibiotic resistance⁶. Developing biofilms on artificial surfaces deployed in the environment is a useful way to study the behavior of marine biofilms as it allows for the observation of initial microbial attachment, community succession, and microbe-host interactions^{7–10}. According to analyses of artificial surface-supported biofilms, the composition of marine microbial biofilms is generally distinct from the surrounding seawater¹⁰ and can be influenced by both location⁷ and substratum type^{8,10}. In addition, a recent study investigated biofilms formed on marine plastic debris, and pointed out that microbes attached to these plastic fragments are different from the surrounding seawater¹¹. However, these insights have been primarily derived from

localized analyses, and the extent to which biofilm-forming microbes contribute to global microbial diversity is unknown. Similarly, while gene functions including signal transduction¹², stress response¹³, motility inhibition¹⁴, and extracellular matrix dynamics¹⁵ are frequently linked to single-species biofilms, the functional core for complex marine biofilm communities remains to be elucidated.

Ecologically, the principle that ‘everything is everywhere, but the environment selects’¹⁶ posits that all microorganisms in the ocean are globally distributed but that in any given environment most of the species are only latently present. This notion has been supported by analysis of microbial communities from environments as varied as the English Channel¹⁷ to Atlantic hydrothermal vents¹⁸.

For the present study, we hypothesized that the oceanic biofilm-forming microbiome contains previously undetected species that are only latently present in seawater, and that these communities maintain a persistent and distinct functional core. To test this, we selected eight locations across the Atlantic, Indian and Pacific Oceans, developed biofilms on different materials, and used deep metagenomic sequencing to compare the community structures with microorganisms in the adjacent seawater and in the entire *Tara* Oceans database.

Results

Microbial diversity and structure of marine biofilms. Locations and materials for biofilm development are shown in Supplementary Figs 1 and 2. In total, 101 biofilm and 24 seawater metagenomes were obtained after microscopic confirmation of the biofilms multicellular structure (Supplementary Fig. 3). In parallel, 67 metagenomes of epipelagic seawater collected from around the world were downloaded from the *Tara* Oceans study⁴.

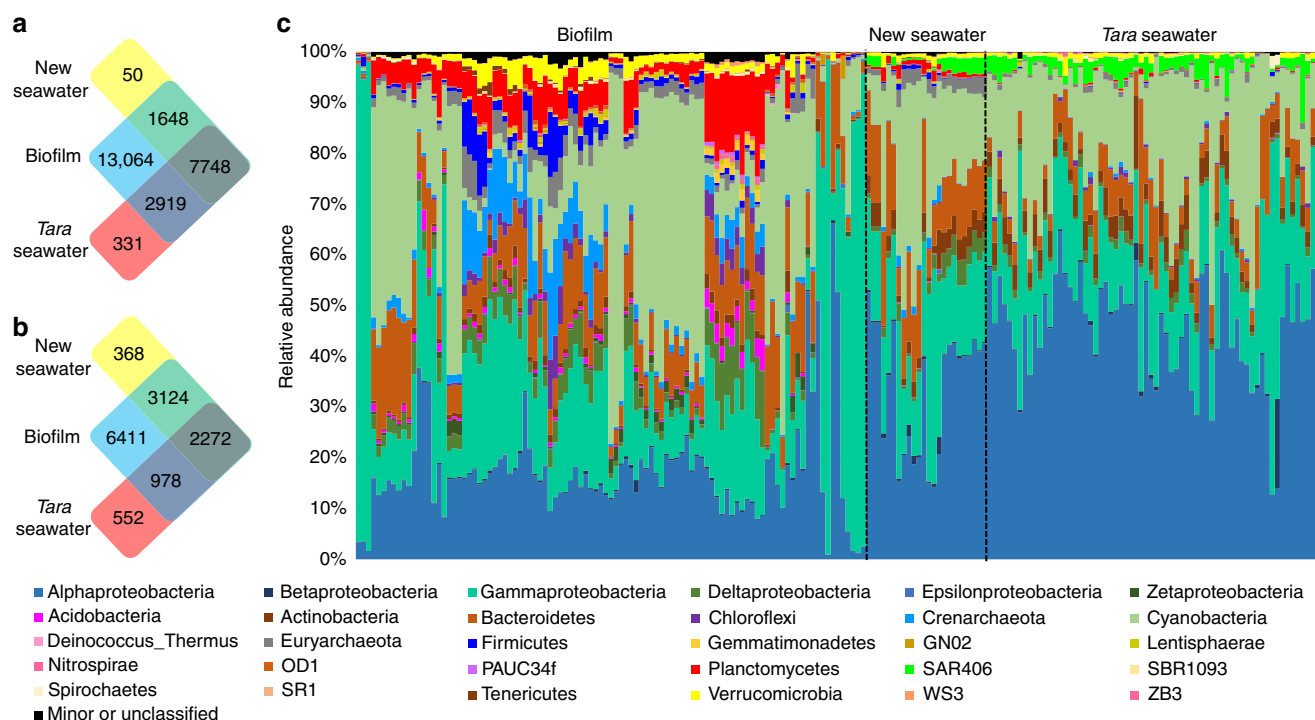


Fig. 1 Comparative taxonomic analysis of the 101 biofilm and 24 seawater metagenomes sequenced as part of the current study and the 67 previously published *Tara* Oceans metagenomes. **a** Venn diagram showing the distribution of all OTUs. **b** Venn diagram showing the distribution of abundant OTUs (OTUs with more than nine tags in at least one sample). **c** Phylum-level taxonomic structure revealing marked differences in the community profiles between the biofilm and seawater habitats. Relative abundance of the phyla was calculated based on 16S miTag numbers. Abundant phyla (the top 30 phyla in terms of maximum relative abundance) are shown with all other phyla grouped together as “Minor”. The order of samples in this figure is the same as that in Supplementary Data 1

16S rRNA gene sequences were extracted from the metagenomes for analysis (subsequently referred to as 16S miTags). Classification of 16S rRNA gene operational taxonomic units (OTU) at 97% similarity resulted in a total of 13,064 OTUs that were exclusively detected in biofilms, 50 OTUs that were unique to the metagenomes of the newly-collected seawater, and 331 OTUs that were unique to the metagenomes of the seawater used in the *Tara* Oceans study (Fig. 1a). When we restricted the analysis to only those OTUs that were abundant (i.e. OTUs with over nine

sequences in at least one sample), then 6411 of the biofilm OTUs were niche-specific (Fig. 1b). Classification of OTUs at the phylum level (class level for Proteobacteria) revealed distinct taxonomic patterns between the biofilms ($n = 101$) and the seawater samples ($n = 91$) (Fig. 1c). We also compared biofilm metagenomes with 16S miTags extracted from all reported *Tara* Oceans data (243 metagenomes yielding 31,612 OTUs) which revealed 7300 OTUs undetected in the *Tara* Oceans study (Fig. 2).

Accumulation curves (gamma-diversity analysis) showed that the rate of biofilm OTU detection dropped to 0.2% by the end of sampling (Supplementary Fig. 4), confirming that sufficient sampling had been undertaken to analyze microbial diversity in the marine biofilms. To test the effect of sequencing depth on the discovery of biofilm-specific OTUs, we additionally deeply sequenced two biofilm samples and two seawater samples collected from the same site at the same time (Hong Kong seawater; Dec, 2017), generating more than 110 gigabytes of data per sample. Based on extraction of 20, 40, 80, and 160 million reads, the specificity of the biofilms (the ratio of biofilm-specific OTUs to the total number of OTUs present in the biofilm and seawater samples) increased slightly with sequencing depth, following a linear relationship, whereas seawater samples decreased with sequencing depth (Supplementary Fig. 5),

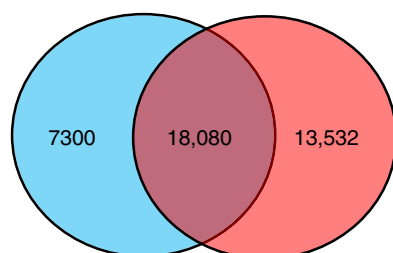


Fig. 2 Venn diagram showing the distribution of OTUs across the biofilms in the current study (blue) and the *Tara* miTags (red) comprising 16S rRNA sequences from 243 seawater samples

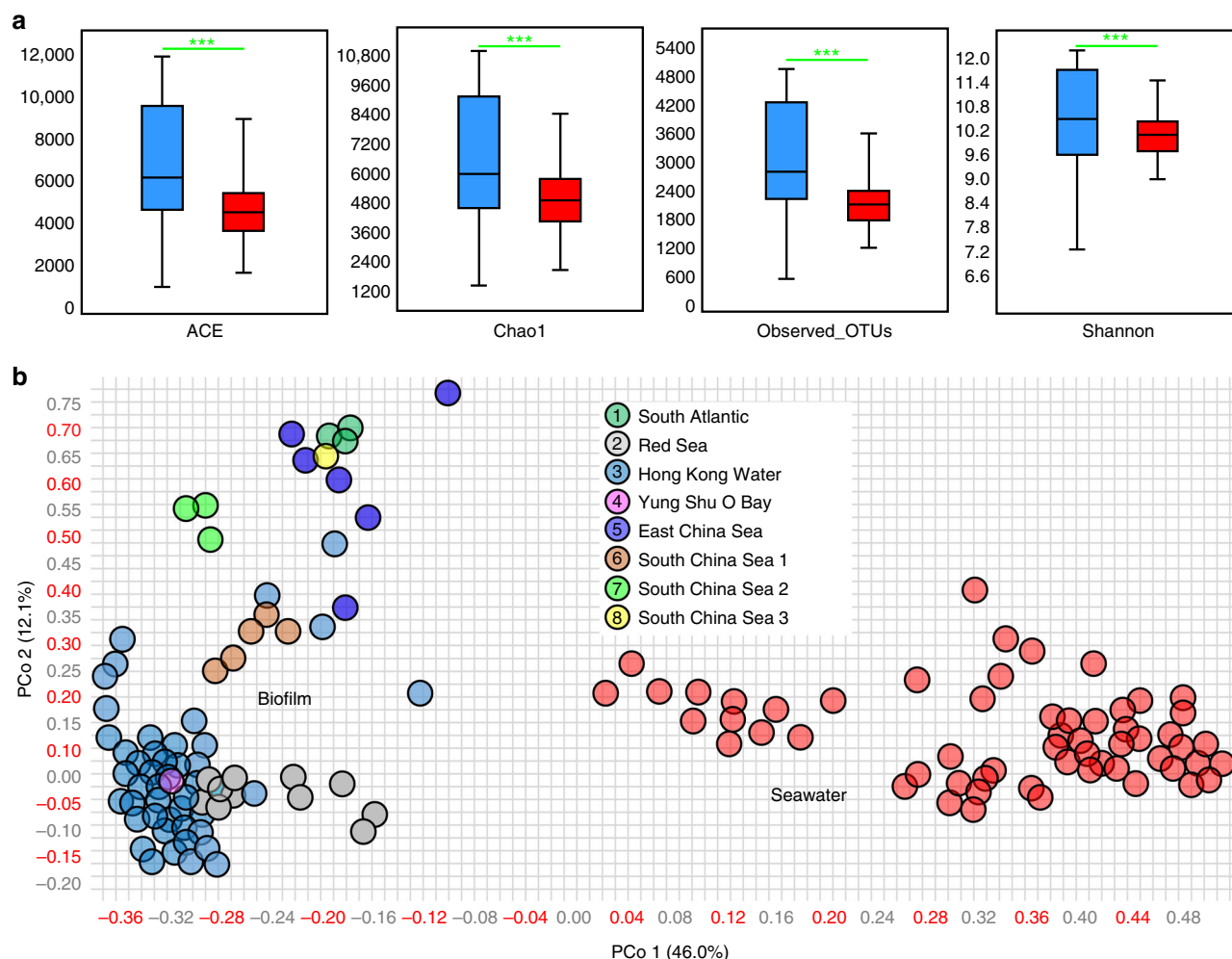


Fig. 3 Alpha- and beta-diversity. **a** ACE, Chao1 diversity, and observed OTUs are significantly different between the biofilm-associated (blue) and seawater-derived (red) microbial communities. *** p -value < 0.001 (two-tailed Student's t -tests after Shapiro-Wilk test to confirm the normal distribution of these data). In a boxplot, central line represents the median, bounds represent upper and lower quartiles, and whiskers represent maximum and minimum. **b** Jaccard similarity of the microbial communities illustrated by PCoA of the OTU matrix. These analyses were performed after normalizing the different metagenome-derived 16S rRNA gene data to the same library size, i.e. 10,000 sequences per sample

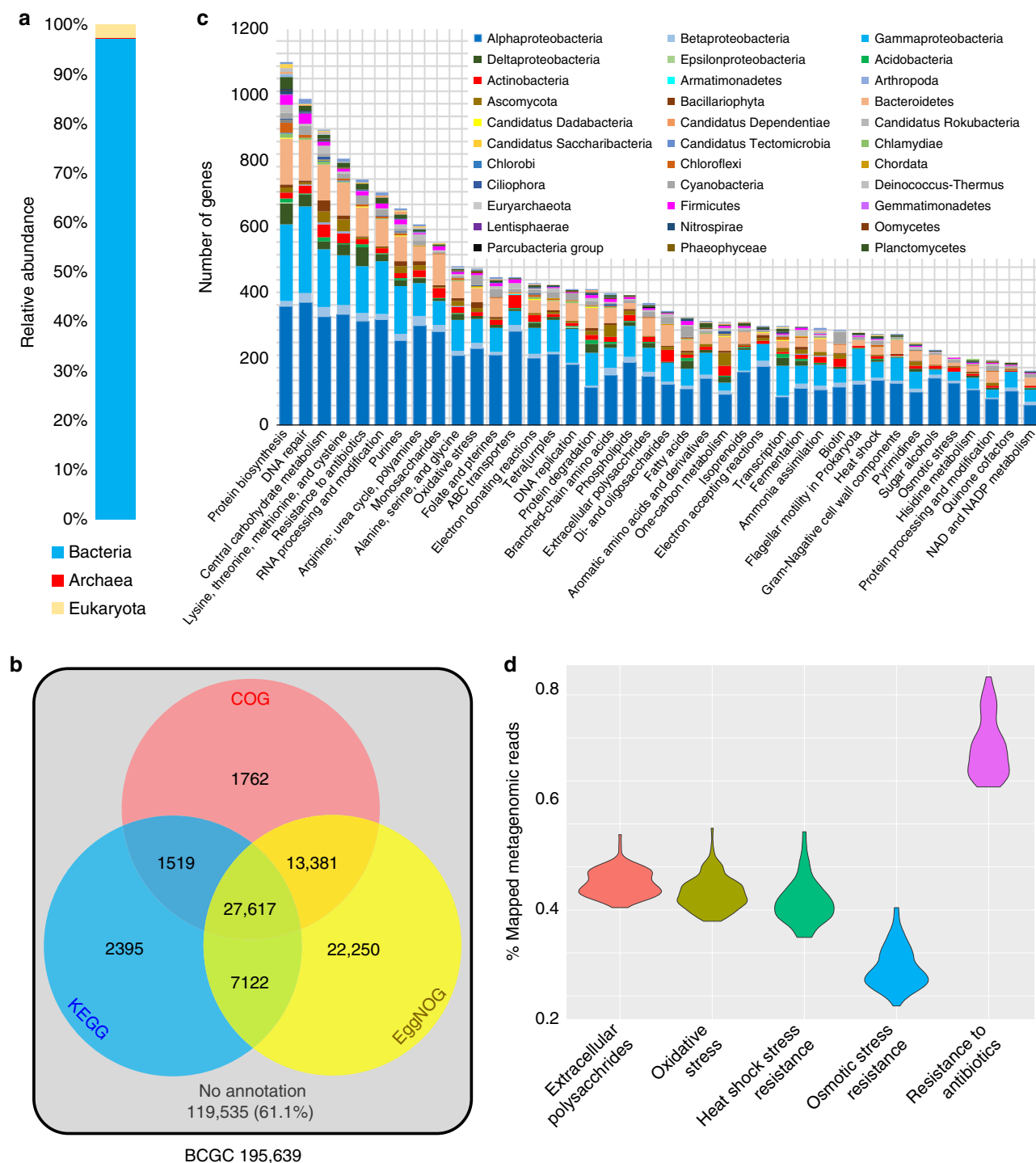


Fig. 4 Functional structuring of the biofilm core gene catalog (BCGC). **a** Taxonomic breakdown of BCGC genes, 97.3% of which belong to bacteria. **b** BCGC is a nonredundant database comprising genes that are present in more than 99 biofilms but which were not detected in any of the seawater samples. Venn diagram showing the total number of genes in the BCGC and the number of genes annotated by three databases. 61.1% of the BCGC genes could not be annotated. **c** Distribution of the annotated BCGC genes in different SEED categories and their taxonomic affiliation. The 40 most enriched SEED categories and their phylum-level affiliations are shown. **d** Reads mapping to BCGC genes in SEED categories related to extracellular polysaccharide biosynthesis, stress and antibiotic resistance, showing the abundance distribution of these functions across the 101 biofilms

suggesting that additional sequencing would reveal additional biofilm-specific species. Based on analysis of 160 million reads, 33.1% of the biofilm OTUs could not be detected in seawater.

To complement the analysis of 16S miTags, we identified protein-coding marker gene sequences for taxonomic profiling, which confirmed biofilms had different phylum-level

composition compared to the seawater microbial communities (Supplementary Fig. 6). Phylum-level composition based on protein-coding marker genes differed from those based on 16S miTags, possibly because the genomic 16S copy number varies substantially across the tree of life¹⁹. To further examine the contribution of 16S rRNA gene copy number and gene length to

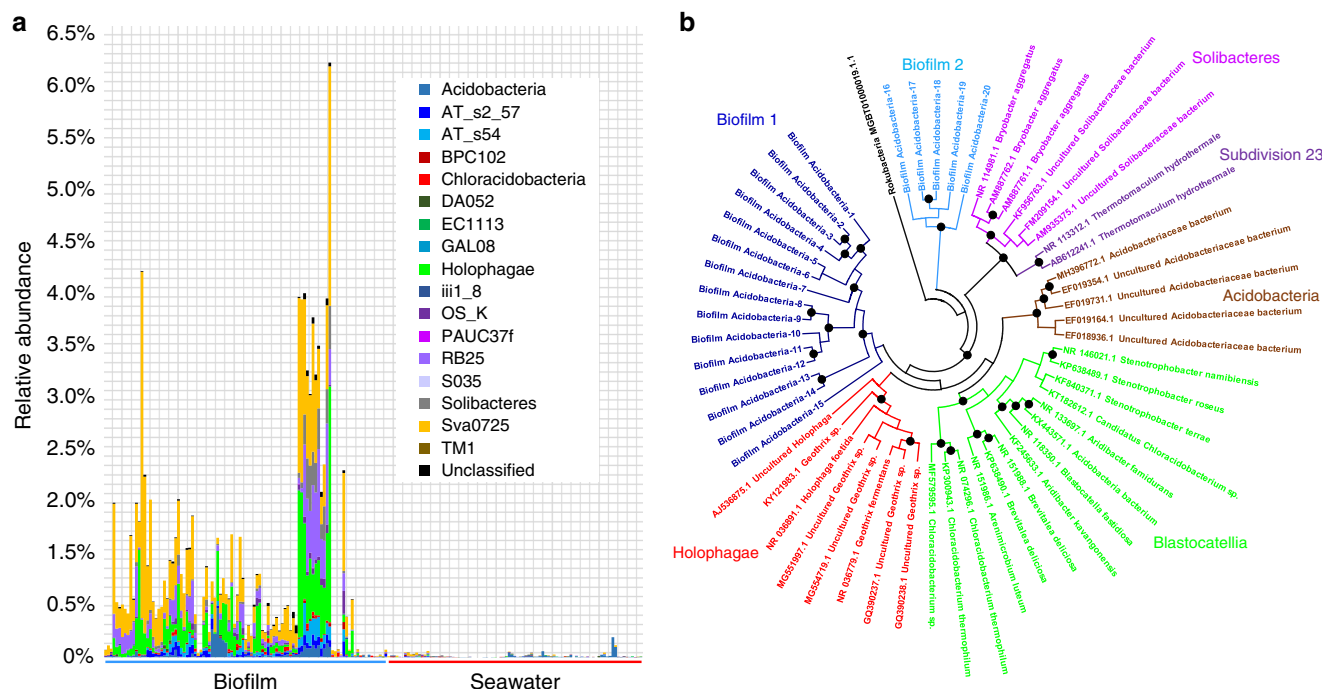


Fig. 5 Novel Acidobacteria lineages. **a** Distribution of Acidobacteria classes in the biofilms and seawater. Based on 16S miTags, several rarely studied or unclassified groups are enriched in biofilms. **b** Phylogenetic analysis of Acidobacteria 16S rRNA gene sequences (over 1000 bp) extracted from assembled biofilm metagenomes. 16S rRNA gene sequences that are available for known Acidobacteria classes were downloaded from the NCBI database and used as references and Rokubacteria was used to root the tree. Black dots indicate bootstrap values above 60. These analyses provide further support for novel microbial lineages within ocean biofilms

explaining the discrepancy between 16S miTag and protein-coding marker gene data, we normalized the data using CopyRighter, and profiled the microbial community structures again (Supplementary Fig. 7). For certain phyla, the relative abundance derived from protein-coding marker genes was more consistent with that of the normalized 16S miTag data (Supplementary Fig. 8). However, regardless of normalization, results derived from 16S miTags and protein-coding marker genes deviated substantially, consistent with results of a systematic study of 16S rRNA gene copy number correction²⁰. While we were unable to resolve the discrepancy amongst approaches within this study, variation in predicting community structure could be attributed to incompleteness of the 16S rRNA gene copy number and protein-coding marker gene databases, which lack information for many uncultured microbes.

Alpha- and beta-diversity analysis based on rarefied 16S miTags (10,000 miTags per sample) from the biofilm and seawater microbiomes revealed a median OTU richness of 2790 and 2155, respectively (Fig. 3a). The median abundance-based coverage estimators (ACE), Chao1, and Shannon diversity were higher for biofilms than for seawater and intra-habitat variation was also greatest in the biofilm samples (Fig. 3a). Principal coordinate analysis (PCoA) using rarefied (Fig. 3b) and total (Supplementary Fig. 9) 16S miTag data both revealed clear separation of the biofilm communities from the planktonic communities, confirming the taxonomic specificity of the biofilms. In addition, PCoA of samples collected from Hong Kong water in 2017 which comprised (i) biofilms developed on artificial panels, (ii) biofilms established on natural rocks, and (iii) seawater microbial communities, revealed larger dissimilarity between biofilms established on artificial panels and seawater than between biofilms occurring on natural rocks and seawater (Supplementary Fig. 10). This analysis suggests that “artificial

communities” may contribute more to the discovery of novel taxa than biofilms on natural substrates.

Functional core analysis and experimental evidence. A high level of endemism was evident in biofilm communities at the protein-coding gene level. On average, 57.7% of the open reading frames (ORFs) derived from the 101 biofilm metagenomes had no hits in the OM-RGC catalog or the newly collected seawater gene catalog from the current study (Supplementary Fig. 11). After deduplication, 11,198,462 protein-coding genes remained and formed a biofilm-specific gene catalog (BSGC). Generation of BSGC distribution profiles in each of the 101 biofilm metagenomes revealed a ubiquitous core (genes present in over 99 biofilms), defined as the biofilm core gene catalog (BCGC) comprising 195,639 nonredundant protein-coding genes.

Taxonomic breakdown of the BCGC showed that 97.3% of the genes belonged to bacteria, 2.5% to eukaryotes, and less than 0.2% to archaea (Fig. 4a). Only 38.9% of the BCGC ORFs had hits in the COG, KEGG, and/or SEED databases (Fig. 4b). Of these annotated genes, the 40 most abundant SEED categories included numerous functions involved in biofilm formation, stress response, and environmental adaptation, such as extracellular polysaccharide biosynthesis, DNA repair and resistance to oxidative stress, heat shock, osmotic stress and antibiotics; and these genes were distributed across a variety of phyla (Fig. 4c). Moreover, the abundance distribution of these genes was consistent across the 101 biofilms (Fig. 4d). Clustering of the BCGC protein domains based on hidden Markov models (HMM) classified genes into 2567 Pfam domains, with 91.0% of the BCGC genes having conserved domains including genes conserved in stress resistance and antibiotic resistance, such as the response regulator (PF00072), HSP90-like ATPase (PF02518), AcrB/AcrD/AcrF family genes (PF00873), oxidoreductase (PF01408), and

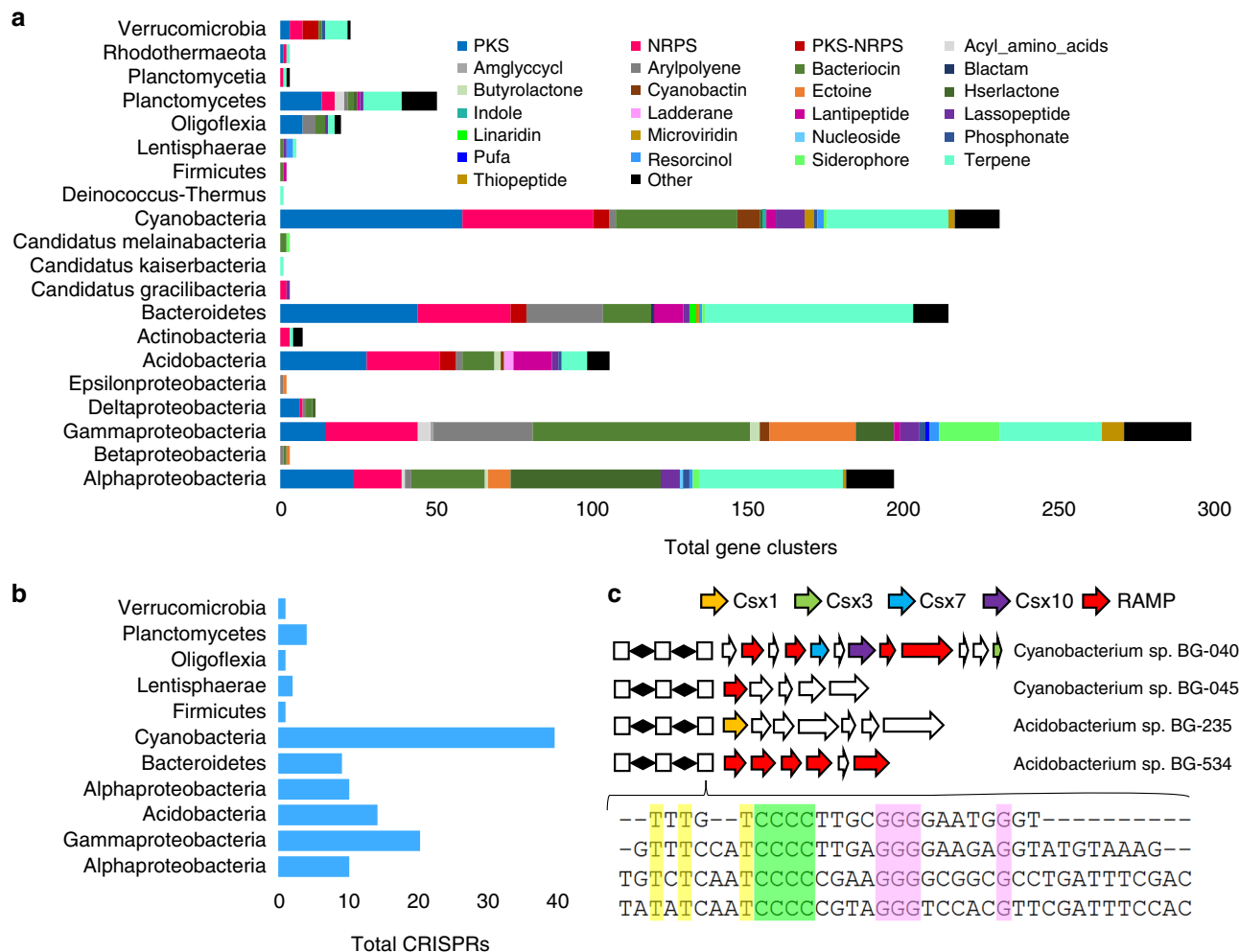


Fig. 6 Functional potential of the biofilm-forming microbes as revealed by analysis of 479 microbial genomes. **a** Biosynthetic gene clusters found on the 479 genomes recovered from 101 biofilm metagenomes. Putative product types were assigned by antiSMASH. **b** Number of CRISPR arrays identified from biofilm-derived genomes. **c** CRISPR arrays reconstructed from two cyanobacterial genomes and two acidobacterial genomes. White boxes indicate repeats and diamonds indicate spacers. Conserved T, C, and G nucleotides are displayed in yellow, green, and pink, respectively. The putative CRISPR-associated genes are shown with different colors, most of which are RNases. RAMP repeat associated mysterious protein

metallo-beta-lactamase (PF00753) (Supplementary Fig. 12). The AcrB/AcrD/AcrF family of genes encode efflux transporters such as enterobactin that mediate resistance to a number of drugs²¹. Metallo-β-lactamases were first identified in a strain of *Bacillus cereus*²² and are responsible for multidrug resistance²³. Given the identification of antibiotic resistance genes, we searched the BCGC against the MEGARes database and found 18 types of antibiotic resistance (Supplementary Fig. 13) arising from 852 genes.

The habitat specificity evident in the biofilm communities suggests that certain taxa and functions with exceptionally low abundance in seawater are enriched during transition from a mobile state to a surface-attached lifestyle. To experimentally test this, laboratory-cultured biofilm communities (Supplementary Fig. 14) and free-living communities derived from the same subtidal seawater, were sequenced. Based on two biological experiments, 918 OTUs were exclusively detected in the biofilm samples (Supplementary Fig. 15a) and of the abundant OTUs (OTUs with over nine sequences in at least one sample), 322 were also unique to biofilms (Supplementary Fig. 15b). Functional comparison revealed that 143 COGs were specific to the biofilms (Supplementary Fig. 16a); 47 of which were uncharacterized genes, and COGs related to signal transduction (e.g., histidine

kinases) and antibiotic resistance (e.g., beta-lactamases) were also identified (Supplementary Fig. 16b), consistent with the finding that genes related to signal transduction become enriched during transition from free-living to surface-associated states.

Novel microbial lineages and functional potential. At the class level, classification of Acidobacteria using 16S rRNA gene sequences from assembled biofilm metagenomes revealed several rarely studied or unclassified groups (e.g., AT_s2_57, AT_s54, and BPC102) (Fig. 5a). Phylogenetic analysis including classified Acidobacteria classes (e.g., Solibacteres, Acidobacteria, and Blastocatellia) revealed that the biofilm-derived Acidobacteria are located on two independent branches of the tree, indicating the existence of previously unexplored lineages (Fig. 5b).

The taxonomic and functional specificity of ocean biofilms led us to further explore novel functional potential through genomic analysis. From the 101 biofilm metagenomes, we recovered 479 partial microbial genomes (only contigs over 10 kb were used). These microbes belonged to 25 different microbial phyla (Supplementary Fig. 17), including Alpha-, Beta-, Gamma-, Delta-, and Epsilonproteobacteria, Acidobacteria, six ‘Candidatus’ phyla, Euryarchaeota, amongst others. Within these contigs, we

identified 1148 biosynthetic gene clusters, distributed across 20 microbial phyla (Fig. 6a). Within the identified gene clusters, 153, 194, and 20 were inferred to synthesize nonribosomal peptides (NRPS), polyketides (PKS), and NRPS-PKS hybrids, respectively. When searching against the MIBiC²⁴ database, 85.6% of the NRPS and PKS biosynthetic proteins showed <60% similarity. The largest number of gene clusters came from Gammaproteobacteria, followed by Cyanobacteria and Bacteroidetes. While only six genomes belonged to Acidobacteria, 103 biosynthetic gene clusters were predicted from these genomes. One acidobacterial genome encoded long NRPS or PKS gene clusters (Supplementary Fig. 18). On the 8-Mb contig from this bacterium, we identified 14 NRPS or PKS biosynthetic loci that were 846.9 kb in total length, and the longest loci was 127.3 kb. We then sought to identify CRISPR-Cas systems in the microbial genomes, finding 116 CRISPR arrays distributed across 11 bacterial phyla (Fig. 6b). The largest number of CRISPR arrays belonged to Cyanobacteria, followed by Gammaproteobacteria and Acidobacteria. Searching against the CRISPRdb database²⁵ using CRISPR repeat sequences resulted in only 12 significant hits (e-value < 1e−5). Through mining the CRISPR-Cas loci, we discovered several loci with similar repeats. For example, a CCCC/GGG motif was discovered within CRISPR repeats from two cyanobacterial and two acidobacterial genomes (Fig. 6c). Examining sequences adjacent to these four CRISPR arrays revealed several putative RNases, including Csx1, Csx3, Csx7, Csx10, and repeat associated mysterious proteins (RAMPs) (Fig. 6c), for which BLASTp searches revealed weak similarity (<40%) to known genes in the NCBI-Nr database. For example, the Csx1 protein in *Acidobacterium* sp. BG-235 possessed two relatively conserved regions with known Csx1 proteins (Supplementary Fig. 19a) while phylogenetic analysis located it in an independent branch of the tree (Supplementary Fig. 19b).

Discussion

The OM-RGC catalog generated by *Tara* Oceans revolutionized our understanding of oceanic microbial diversity and functional potential and provided a platform to resolve the ecological roles of the microbial community. 16S miTags derived from the *Tara* Oceans metagenomic datasets mapped to a total of 35,650 OTUs, with the rate of new gene detection only 0.01% by the end of seawater sampling⁴. However, by sampling diverse biofilm surfaces assembled from seawater across broad spatial and temporal scales and performing comparisons with the OM-RGC, we revealed a large reservoir of novel microbial OTUs and functional genes not previously considered part of the global ocean microbiome. Overall, metagenomic analyses of these biofilms increased the known microbial diversity of the global oceans by more than 20%. Of the biofilm ORFs, 57.7% had no hits in the OM-RGC catalog or the newly collected seawater gene catalog, providing further support for underestimated diversity in the ocean microbiome.

Analysis of the taxonomic and functional core of the biofilm-forming microbiome revealed the existence of persistent microbial groups and associated functions. The findings of beta-diversity analyses are noteworthy given that biofilms were collected across such broad spatial and temporal scales and included diverse substrate surfaces. The BCGC contained endemic genes with conserved domains, suggesting that gene sequences may have evolved along divergent paths without affecting the functional domains. Stress response and antibiotic resistance genes appear to be important components of the biofilm functional core. Stress response contributes to bacterial survival in natural environments and can trigger biofilm formation²⁶. The function of antibiotic resistance genes in natural microbial communities is often uncertain, as their

abundance is not solely determined by the presence of these antibiotics²⁷. However, antibiotics have been shown to act as signaling molecules within natural microbial communities²⁸, and thus antibiotic resistance genes may play roles outside of the ‘war’ metaphor, consistent with recent findings in soil microbial communities²⁹. Our laboratory experiment also highlighted the role of specific functions in determining niche differentiation between surface-associated and planktonic communities. We surmise that the biofilm core gene set underpins environmental adaptation and microbe–microbe interaction.

The biofilm microbiome also serves as a resource for the discovery of novel microbial lineages and functional genes. Previous analyses based on publicly available acidobacterial genomes identified few biosynthetic clusters^{30–32}, whereas a recent study showed that Acidobacteria from soil express unusually large PKS and NRPS biosynthetic genes³³. Here we show that Acidobacteria in biofilms formed in the marine environment also contain numerous biosynthetic gene clusters. In addition, we identified several PKS biosynthetic genes in *Oligoflexia* which had not previously been linked to secondary metabolite production. In recent years, a wealth of sequence information has accumulated for CRISPR-Cas, facilitating discovery of new systems and novel functions³⁴. Here we identified a number of CRISPR arrays in the biofilm microbial genomes, consistent with an expectation for enhanced interactions between viruses and bacteria within biofilms^{35,36}. The finding of CRISPR loci with conserved repeats in cyanobacterial and acidobacterial genomes and the presence of adjacent putative RNases suggests the existence of conserved but functionally unknown CRISPR-Cas systems in marine biofilms.

While surface-associated microorganisms must be seeded from the seawater microbiome, many microbial species in seawater are simply too scarce to be captured in global sequencing efforts and can only be detected when their relative abundance increases during biofilm formation. Here we demonstrate the important niche differentiation that occurs between planktonic and surface-associated communities and show that marine biofilms represent a novel species bank of hidden microbial diversity and functional potential in the ocean.

Methods

Sampling. Biofilms were developed at a depth of 1–2 m at eight locations spanning Sapelo Island in the South Atlantic, Red Sea, Hong Kong Seawater, Yung Shu O Bay, East China Sea, South China Sea 1, South China Sea 2, and South China Sea 3 (Supplementary Fig. 1). Biofilms developed in Hong Kong seawater were collected annually between 2013 and 2017. Artificial panels and polystyrene Petri dishes were used to establish the biofilms: 9 × 1.2 cm sterile polystyrene Petri dishes (Supplementary Fig. 2a), 11 × 11 cm zinc panels used for preventing marine biofouling (Supplementary Fig. 2b), and 5 × 5 cm panels of aluminum, poly(ether-ether-ketone), polytetrafluoroethylene, poly(vinyl chloride), stainless steel, and titanium (Supplementary Fig. 2c) held in place by a customized device (Supplementary Fig. 2d). Most biofilms were fully established on Petri dishes after 12 d, and 30 d were required for visible biofilm formation on the metal surfaces. Biofilm development was limited to 3 d in the South China Sea for logistical reasons. For biofilms collected in Jun, Jul, Aug, and Sep, the panels were covered with mesh to prevent settlement of macroorganisms. Biofilms were also removed from rocks present in the subtidal zone (Supplementary Fig. 2e). Collected biofilms were immediately transferred to the laboratory, and the biomass was removed using sterile cotton tips and stored at −80 °C in 5 ml of DNA storage buffer (500 mM NaCl, 50 mM Tris-HCl, 40 mM EDTA and 50 mM glucose). Samples were observed under a Zeiss LSM 510 confocal laser scanning microscope (Carl Zeiss, Jena, Germany) as previously described³⁷. The observation revealed thick films with multiple cell layers, thus confirming microbial attachment (Supplementary Fig. 3). Prior to the observation, microorganisms were labeled with the protein dye fluorescein isothiocyanate (FITC) (Sigma, Poole, UK). Images were generated using a 63 × 1.4 objective, and signals were recorded in the green channel (excitation 488 nm, emission 522/32 nm). Adjacent seawater samples were collected at the same time as biofilm retrieval and successively filtered through 0.22-μm and 0.1-μm polycarbonate membrane filters (Millipore, Massachusetts, USA). The filters were stored in 5 ml of DNA storage buffer at −80 °C. In total, 101 biofilm and 24 seawater samples were collected. Two additional seawater samples were collected for a laboratory experiment on biofilm formation (see below for details).

DNA extraction. Before DNA extraction, 10 mg/ml lysozyme was added to the samples and incubated at 37 °C for 30 min. To test the efficiency of DNA extraction, a subset of biofilms was also extracted with the TIANamp Genomic DNA Kit (Tiangen Biotech, Beijing, China), the AllPrep DNA/RNA Mini Kit (Qiagen, Hilden, Germany), and the DNeasy Blood & Tissue Kit (Qiagen, Hilden, Germany), following the Manufacturer's protocols; in total, six biofilms were collected from two Hong Kong seawater sites, with three biofilms collected per site for each of the three extraction methods. The TIANamp Genomic DNA Kit was used to extract DNA from all other biofilm and seawater samples.

Metagenomic analyses. Most samples were sequenced at the Novogene Bioinformatics Institute (Novogene, Beijing, China) and a subset was sequenced at the Beijing Genomics Institute (BGI, Beijing, China). After construction of 350 bp insert libraries, metagenomic samples (including both biofilms and seawater) were pair-end sequenced on the HiSeq X Ten System at Novogene and the HiSeq 2500 System at BGI. The read length of the data generated on both platforms was 150 bp. Metagenomes were generated for 101 biofilm and 24 adjacent seawater metagenomes, with 115 ± 39 million reads per biofilm metagenome and 130 ± 33 million reads per seawater metagenome. Quality control was performed on a local server using the software NGS QC Toolkit (version 2.0)³⁸. Reads containing adaptors, low quality reads (assigned by a quality score <20 for >30% of the read length) or unpaired high-quality reads were removed. Following quality control, short reads from the biofilm and seawater metagenomes were assembled into contigs using the software MEGAHIT (version 1.0.2)³⁹ with kmer values from 21 to 121. The OM-RGC and 67 metagenomes of surface seawater in the *Tara* Oceans study were downloaded for comparison, with 302 ± 114 million reads per metagenome. Detailed information about samples and corresponding sequencing data is shown in Supplementary Data 1. Information on assembled metagenomic contigs is shown in Supplementary Data 2.

Taxonomic profiling. 16S miTags were extracted from the unassembled metagenomic data using Parallel-META3⁴⁰, which relies on HMMER (version 3.1)⁴¹ to predict the 16S miTags from both the forward and reverse sequences. Bowtie2⁴² was used to map the 16S miTags to a database that integrates GreenGenes⁴³ with RDP⁴⁴ and SILVA⁴⁵ (assigned by BLASTn with e-value < $1e-30$ and similarity >97%). OTUs were picked at 97% similarity or above. To reduce the false discovery rate due to sequencing errors, only OTUs identified in both forward and reverse paired reads in a sample were retained (thus all singletons were removed). Considering the primary objective of this analysis was to estimate the total number of biofilm-specific species and considerably more data exists for seawater metagenomes than for the biofilm metagenomes, OTU accumulative analysis of the 101 biofilms was performed using the 'vegan' package in R with 100 permutations. The full OTU list generated by the 16S miTags and used for Venn diagram comparison is given in Supplementary Data 3. The full list of phyla in biofilms and seawater based on 16S miTag analysis is given in Supplementary Data 4. To assess the effect of 16S rRNA gene copy number and gene length on taxonomic classification, results derived from the 16S miTags were normalized using the software Copy-Righter⁴⁶, which performs lineage-specific gene length and gene copy number correction. The databases *genlength_img40_gg201210* and *ssu_img40_gg201210.txt* were used. The full list of phyla in biofilms and seawater based on normalized 16S miTags is given in Supplementary Data 5. Metagenomic reads were also mapped to the mOTU.v1.padded database consisting of 40 marker genes for taxonomic profiling as previously described⁴⁷. Phylum-level profiles were generated for the 101 biofilm and 91 seawater samples. The list of phyla in all biofilm and seawater microbial communities based on these marker genes is given in Supplementary Data 6.

Sample accumulation curve and sequencing depth analysis. The accumulation curve of detected OTUs in the 101 biofilms was calculated using the 'vegan' package in R with 100 permutations. The OTU table based on the 16S miTags was used as the query. To determine how specificity was influenced by sequencing depth, two biofilms were developed on a Petri dish for 12 days in Hong Kong seawater in Dec 2017 and two seawater samples were collected adjacent to the sites of biofilm development. Metagenomic sequencing was performed on the HiSeq X Ten System in Novogene (Beijing, China) to generate 110 Gb data. The seqtk program (<https://github.com/lh3/seqtk>) was used to randomly extract 20 M, 40 M, 80 M, and 160 M reads from the metagenomes. The extracted reads were used for 16S miTags identification and OTU classification. The biofilm and seawater specificity was calculated based on the formulas:

$$\text{Biofilm specificity} = \frac{\text{Number of biofilm-specific OTUs}}{\text{Number of OTUs in biofilm and/or water}}$$

$$\text{Water specificity} = \frac{\text{Number of water-specific OTUs}}{\text{Number of OTUs in biofilm and/or water}}$$

Alpha- and beta-diversity analyses. Alpha- and beta-diversity analyses were performed on the 101 biofilm and 91 seawater metagenomes. To normalize data

size, 10,000 16S miTags were extracted from each sample (only forward sequencing files were used). OTUs were picked at 97% similarity or above. ACE, Chao1, observed OTUs, and Shannon diversity were calculated using scripts (alpha_diversity.py and default parameters) implemented in QIIME⁴⁸. Jaccard distances, which are based on the presence/absence of taxa, were generated using the OTU table for pairwise comparison of the biofilm and seawater communities. Jaccard distances were revealed by PCoA using the software PAST (version 2.0)⁴⁹. In addition, PCoA was performed using the total 16S miTags. The OTU tables constructed using rarefied (10,000 per sample) 16S miTags and total 16S miTags are presented in Source Supplementary Data 7 and 8, respectively.

Identification and annotation of the biofilm functional core gene set. ORFs were predicted from the contigs using MetaGeneMark (version 2.8) (predict genes on both strands; gene overlaps are allowed; probability of initiation and termination in non-coding state 0.5). Protein sequences (minimum length 100) derived from the 24 seawater metagenomes were combined and subjected to CD-HIT⁵⁰ (>95% sequence identity for >60% of the length of the shorter sequences) to generate a NSGC comprising 2,541,441 genes. Protein sequences (minimum length 100) derived from the biofilm metagenomes were BLASTp (e-value < $1e-7$; >60% sequence identity for >60% of the length of the shorter sequences) searched against OM-RGC⁴ and the new seawater gene reference catalog generated in this study. Proteins without BLASTp hits were then searched against the OM-RGC and the new seawater gene reference catalog using USEARCH (global alignments)⁵¹ to confirm BLASTp results. Protein sequences without BLASTp or USEARCH hits were subjected to CD-HIT (>95% sequence identity for >60% of the length of the shorter sequences) to generate a nonredundant catalog called BSGC. The BSGC genes that were present in more than 99 of the 101 biofilms were defined as the BCGC. For functional annotation, BCGC proteins were DIAMOND⁵² BLASTp (e-value < $1e-7$; >60% sequence identity for >60% of the length of the shorter sequences) searched against the COG⁵³, KEGG (2016 version)⁵⁴, and eggNOG databases⁵⁵. eggNOG individual gene annotations and functional category profiles of the BCGC were generated by linking the eggNOG annotations with the SEED subsystems database⁵⁶ using mapping files documented in MOCAT⁵⁷. Moreover, the BCGC protein sequences were searched against the Pfam database⁵⁸ using HMMER hmmscan (e-value < $1e-7$) to identify conserved functional domains. Finally, the same protein sequences were BLASTp searched against the MEGARes database⁵⁹ (e-value < $1e-7$; >60% sequence identity for >60% of the length of the shorter sequences) to identify antibiotic resistance genes. Data from all BCGC gene annotations (COGs, KEGG, SEED, and Pfam) are presented in Supplementary Data 9. The full list of antibiotic resistance genes is provided in Supplementary Data 10.

Laboratory experiment on biofilm formation. Hong Kong seawater samples were collected in duplicate, 10 m apart in Jan 2018. Two-hundred milliliters of each sample were filtered through 0.1-μm polycarbonate membrane filters. Microbes were washed from the membranes using 20 ml of 0.1-μm filtered and autoclaved seawater and each sample was divided into two parts for parallel experiments. In treatment 1 (biofilm-forming culture), 10 ml of each aliquot was placed in the modified M9 minimum medium (prepared using 0.1-μm filtered and autoclaved seawater) and incubated at 22 °C for 24 h in the wells of a six-well plate (BD Biosciences, California, USA). Films that formed on the bottom surface of the plate were removed for DNA extraction and metagenomic sequencing. In treatment 2 (free-living culture), the remaining 10 ml of each aliquot was suspended for 24 h in 50 ml falcon tubes in an incubator shaker (22 °C, 100 rpm). Microorganisms in the falcon tubes were collected by centrifugation at 4000 × g for 10 min for DNA extraction and metagenomic sequencing. Taxonomic analysis was performed based on 16S miTags, as described above. Functional analysis was performed using searches against the COG database. The OTUs and protein-coding genes are given in Supplementary Data 11 and 12, respectively.

Mining novel lineages. 16S rRNA genes (over 1000 bp) were extracted from contigs based on HMM using the software Meta_RNA⁶⁰. The 16S rRNA gene sequences of representative members in all classes of Acidobacteria were downloaded from GenBank. The 16S rRNA gene sequence retrieved from a genome of Rokubacteria, which are close relatives of Acidobacteria in the tree of life⁶¹, was used to root the tree. The maximum-likelihood phylogenetic tree was constructed in MEGA (version 7.0)⁶² using the Tamura-Nei substitution model and bootstrapping was performed with 1000 repetitions.

Genome binning and analyses. Genome binning was performed with contigs longer than 10 kb. A total of 105,905 contigs were clustered into genome bins using a hybrid binning approach. Initially, reads from a single metagenome were mapped to the corresponding assembly using Bowtie2 (fastq as query input files) to generate a coverage profile, which was used to inform the binning software MaxBin2⁶³ (minimum probability for EM algorithm 0.8; 107 marker genes) to separate the contigs. Subsequently, these contig groups were individually subjected to a second binning process using MetaBAT⁶⁴. The resulting output genome bins were examined for completeness and potential contamination using CheckM⁶⁵ (lineage_wf). FetchMG⁴⁷ (version 1.0) was used to identify single-copy genes from the

genome bins to check for potential contamination and analyze the taxonomic affiliation using default bit scores. Closely related genome bins were compared based on average nucleotide identity (ANI)⁶⁶. Genomes with an ANI value higher than 0.99 were considered duplicates and the most complete genome bin was retained. Genome bins were processed using antiSMASH (version 4.0)⁶⁷ (default parameters) for biosynthetic gene cluster identification. Proteins involved in NRPS or PKS biosynthesis were searched against the MiBIG repository²⁴ using BLASTP (e-value < 1e-7; >60% sequence identity for >60% of the length of the shorter sequences) for similarity assessment. CRISPR arrays were identified using the CRISPRs mining tool MinCED⁶⁸ (minimum number of repeats a CRISPR 3; minimum length of the CRISPR repeats 23; maximum length of the CRISPR repeats 47; minimum length of the CRISPR spacers 26; maximum length of the CRISPR spacers 300) and selectively checked using the online tool CRISPRFinder⁶⁹ (<http://crispr.i2bc.paris-saclay.fr/Server/>) using default parameters. Proteins adjacent to the CRISPR arrays were searched against the online NCBI-Nr database for annotation. The Csx1 protein sequence and its top hits in the NCBI-Nr database were included for alignment and phylogenetic analysis using MEGA (version 7.0)⁶². A maximum-likelihood tree was constructed using the Jones-Taylor-Thornton substitution model, bootstrapping with 500 repetitions. Genome information, including taxonomic affiliation, genome completeness, and potential contamination is shown in Supplementary Data 13. Total biosynthetic gene clusters and CRISPR arrays are given in Supplementary Data 14 and 15, respectively.

Reporting summary. Further information on experimental design is available in the Nature Research Reporting Summary linked to this article.

Data availability

All metagenomic datasets have been deposited in the NCBI database under BioProject accession no. PRJNA438384 (BioSample no. SAMN08714533 for the 101 biofilm metagenomes; BioSample no. SAMN0871453 for the 24 adjacent seawater metagenomes; BioSample no. SAMN08714627 for the two biofilm and two seawater metagenomes that have been sequenced for more than 110 Gb data; and BioSample no. SAMN08714628 for the metagenomes of the laboratory cultured biofilm and free-living microbes). Gene catalogs, including the biofilm-specific genes and core genes, as well as the seawater gene sequences are uploaded to figshare (BSGC, <https://figshare.com/s/7e193dceb329c02cb166>, <https://doi.org/10.6084/m9.figshare.6743327>; BCGC, <https://figshare.com/s/53f27d05d443c6c947e5>, <https://doi.org/10.6084/m9.figshare.6743324>; NSGC, <https://figshare.com/s/7815ea074e5c2d7ef1c5>, <https://doi.org/10.6084/m9.figshare.6743459>). The 479 microbial genomes are uploaded to figshare (<https://figshare.com/s/2994fdaf79112b99907>, <https://doi.org/10.6084/m9.figshare.7082684>). Supplementary Figs 1–19 and Supplementary Data 1–15 are present in the supplementary materials.

Received: 16 September 2018 Accepted: 21 December 2018

Published online: 31 January 2019

References

- Falkowski, P. G., Barber, R. T. & Smetacek, V. V. Biogeochemical controls and feedbacks on ocean primary production. *Science* **281**, 200–207 (1998).
- Bourne, D. G., Morrow, K. M. & Webster, N. S. Coral holobionts—Insights into the coral microbiome: Underpinning the health and resilience of reef ecosystems. *Ann. Rev. Microbiol.* **70**, 317–340 (2016).
- Hadfield, M. G. Biofilms and marine invertebrate larvae: what bacteria produce that larvae use to choose settlement sites. *Annu. Rev. Mar. Sci.* **3**, 453–470 (2011).
- Sunagawa, S. et al. Structure and function of the global ocean microbiome. *Science* **348**, 1261359 (2015).
- Salta, M., Wharton, J. A., Blache, Y., Stokes, K. & Briand, J. F. Marine biofilms on artificial surfaces: structure and dynamics. *Environ. Microbiol.* **15**, 2879–2893 (2013).
- Dang, H. & Lovell, C. R. Microbial surface colonization and biofilm development in marine environments. *Microbiol. Mol. Biol. Rev.* **80**, 91–138 (2016).
- Lee, O. O. et al. In situ environment rather than substrate type dictates microbial community structure of biofilms in a cold seep system. *Sci. Rep.* **4**, 3587 (2014).
- Pedersen, K. Biofilm development on stainless steel and PVC surfaces in drinking water. *Water Res.* **24**, 239–243 (1990).
- Chung, H. C. et al. Bacterial community succession and chemical profiles of subtidal biofilms in relation to larval settlement of the polychaete *Hydroides elegans*. *ISME J.* **4**, 817 (2010).
- Zhang, W. et al. Species sorting during biofilm assembly by artificial substrates deployed in a cold seep system. *Sci. Rep.* **4**, 6647 (2014).
- Bryant, J. A. et al. Diversity and activity of communities inhabiting plastic debris in the North Pacific Gyre. *mSystems* **1**, e00024–16 (2016).
- Carniol, K. & Gilmore, M. S. Signal transduction, quorum-sensing, and extracellular protease activity in *Enterococcus faecalis* biofilm formation. *J. Bact.* **186**, 8161–8163 (2004).
- Landini, P. Cross-talk mechanisms in biofilm formation and responses to environmental and physiological stress in *Escherichia coli*. *Res. Microbiol.* **160**, 259–266 (2009).
- Guttenplan, S. B. & Kearns, D. B. Regulation of flagellar motility during biofilm formation. *FEMS Microbiol. Rev.* **37**, 849–871 (2013).
- Zhang, W. et al. Extracellular matrix-associated proteins form an integral and dynamic system during *Pseudomonas aeruginosa* biofilm development. *Front. Cell. Infect. Microbiol.* **5**, 40 (2015).
- De Wit, R. & Bouvier, T. ‘Everything is everywhere, but, the environment selects’; what did Baas Becking and Beijerinck really say? *Environ. Microbiol.* **8**, 755–758 (2006).
- Gibbons, S. M. et al. Evidence for a persistent microbial seed bank throughout the global ocean. *Proc. Natl. Acad. Sci. USA* **110**, 4651–4655 (2013).
- Gonnella, G. et al. Endemic hydrothermal vent species identified in the open ocean seed bank. *Nat. Microbiol.* **1**, 16086 (2016).
- Acinas, S. G., Marcelino, L. A., Klepac-Ceraj, V. & Polz, M. F. Divergence and redundancy of 16S rRNA sequences in genomes with multiple *rrn* operons. *J. Bacteriol.* **186**, 2629–2635 (2004).
- Louca, S., Doebeli, M. & Parfrey, L. W. Correcting for 16S rRNA gene copy numbers in microbiome surveys remains an unsolved problem. *Microbiome* **6**, 41 (2018).
- Horiyama, T. & Nishino, K. AcrB, AcrD, and MdtABC multidrug efflux systems are involved in enterobactin export in *Escherichia coli*. *PLoS One* **9**, e108642 (2014).
- Sabat, L. D. & Abraham, E. P. Zinc as a cofactor for a cephalosporinase from *Bacillus cereus* 569. *Biochem. J.* **98**, 11–13 (1966).
- Esposito, E. P. et al. A novel IncA/C1 group conjugative plasmid, encoding VIM-1 metallo-beta-lactamase, mediates the acquisition of carbapenem resistance in ST104 *Klebsiella pneumoniae* isolates from neonates in the intensive care unit of V. Monaldi hospital in Naples. *Front. Microbiol.* **8**, 2135 (2017).
- Medema, M. H. et al. Minimum information about a biosynthetic gene cluster. *Nat. Chem. Biol.* **11**, 625–631 (2015).
- Grissa, I., Vergnaud, G. & Pourcel, C. The CRISPRdb database and tools to display CRISPRs and to generate dictionaries of spacers and repeats. *BMC Bioinforma.* **8**, 172 (2007).
- Geier, H., Mostowy, S., Cangelosi, G. A., Behr, M. A. & Ford, T. E. Autoinducer-2 triggers the oxidative stress response in *Mycobacterium avium*, leading to biofilm formation. *Appl. Environ. Microbiol.* **74**, 1798–1804 (2008).
- Singer, R. S., Ward, M. P. & Maldonado, G. Can landscape ecology untangle the complexity of antibiotic resistance? *Nat. Rev. Microbiol.* **4**, 943–952 (2006).
- Romero, D., Traxler, M. F., López, D. & Kolter, R. Antibiotics as signal molecules. *Chem. Rev.* **111**, 5492–5505 (2011).
- Allen, H. K., Moe, L. A., Rodbummer, J., Gaarder, A. & Handelsman, J. Functional metagenomics reveals diverse β -lactamases in a remote Alaskan soil. *ISME J.* **3**, 243 (2009).
- Hadjithomas, M. et al. IMG-ABC: a knowledge base to fuel discovery of biosynthetic gene clusters and novel secondary metabolites. *mBio* **6**, e00932–e15 (2015).
- Cimermanic, P. et al. Insights into secondary metabolism from a global analysis of prokaryotic biosynthetic gene clusters. *Cell* **158**, 412–421 (2014).
- Wang, H., Fewer, D. P., Holm, L., Rouhiainen, L. & Sironen, K. Atlas of nonribosomal peptide and polyketide biosynthetic pathways reveals common occurrence of nonmodular enzymes. *Proc. Natl. Acad. Sci. USA* **111**, 9259–9264 (2014).
- Crits-Christoph, A., Diamond, S., Butterfield, C. N., Thomas, B. C. & Banfield, J. F. Novel soil bacteria possess diverse genes for secondary metabolite biosynthesis. *Nature* **558**, 440–444 (2018).
- Koonin, E. V., Makarova, K. S. & Zhang, F. Diversity, classification and evolution of CRISPR-Cas systems. *Curr. Opin. Microbiol.* **37**, 67–78 (2017).
- Sutherland, I. W., Hughes, K. A., Skillman, L. C. & Tait, K. The interaction of phage and biofilms. *FEMS Microbiol. Lett.* **232**, 1–6 (2004).
- Carrolo, M., Frias, M. J., Pinto, F. R., Melo-Cristino, J. & Ramirez, M. Prophage spontaneous activation promotes DNA release enhancing biofilm formation in *Streptococcus pneumoniae*. *PLoS One* **5**, e15678 (2010).
- Ding, W. et al. Anti-biofilm effect of a butenolide/polymer coating and metatranscriptomic analyses. *Biofouling* **34**, 111–122 (2018).
- Patel Ravi, K. & Mukesh, J. NGS QC Toolkit: a toolkit for quality control of next generation sequencing data. *PLoS One* **7**, e30619 (2012).
- Li, D., Liu, C. M., Luo, R., Sadakane, K. & Lam, T. W. MEGAHIT: an ultra-fast single-node solution for large and complex metagenomics assembly via succinct de Bruijn graph. *Bioinformatics* **31**, 1674–1676 (2015).

40. Jing, G. et al. Parallel-META 3: Comprehensive taxonomical and functional analysis platform for efficient comparison of microbial communities. *Sci. Rep.* **7**, 40371 (2017).
41. Eddy, S. R. Accelerated profile HMM searches. *PLoS. Comput. Biol.* **7**, e1002195 (2011).
42. Langmead, B. & Salzberg, S. L. Fast gapped-read alignment with Bowtie 2. *Nat. Methods* **9**, 357–359 (2012).
43. DeSantis, T. Z. et al. Greengenes, a chimera-checked 16S rRNA gene database and workbench compatible with ARB. *Appl. Environ. Microbiol.* **72**, 5069–5072 (2006).
44. Cole, J. R. et al. Ribosomal Database Project: data and tools for high throughput rRNA analysis. *Nucleic Acids Res.* **42**, 633–642 (2013).
45. Quast, C. et al. The SILVA ribosomal RNA gene database project: improved data processing and web-based tools. *Nucleic Acids Res.* **41**, 590–596 (2012).
46. Angly, F. E. et al. CopyRighter: a rapid tool for improving the accuracy of microbial community profiles through lineage-specific gene copy number correction. *Microbiome* **2**, 11 (2014).
47. Sunagawa, S. et al. Metagenomic species profiling using universal phylogenetic marker genes. *Nat. Methods* **10**, 1196 (2013).
48. Caporaso, J. G. et al. QIIME allows analysis of high-throughput community sequencing data. *Nat. Methods* **7**, 335 (2010).
49. Hammer, Ø., Harper, D. A. T. & Ryan, P. D. PAST: paleontological statistics software package for education and data analysis. *Palaeontol. Electron.* **4**, 1–9 (2001).
50. Fu, L., Niu, B., Zhu, Z., Wu, S. & Li, W. CD-HIT: accelerated for clustering the next-generation sequencing data. *Bioinformatics* **28**, 3150–3152 (2012).
51. Edgar, R. C. Search and clustering orders of magnitude faster than BLAST. *Bioinformatics* **26**, 2460–2461 (2010).
52. Buchfink, B., Xie, C. & Huson, D. H. Fast and sensitive protein alignment using DIAMOND. *Nat. Methods* **12**, 59 (2015).
53. Galperin, M. Y., Makarova, K. S., Wolf, Y. I. & Koonin, E. V. Expanded microbial genome coverage and improved protein family annotation in the COG database. *Nucleic Acids Res.* **43**, 261–269 (2014).
54. Kanehisa, M., Furumichi, M., Tanabe, M., Sato, Y. & Morishima, K. KEGG: new perspectives on genomes, pathways, diseases and drugs. *Nucleic Acids Res.* **45**, 353–361 (2016).
55. Jensen, L. J. et al. eggNOG: automated construction and annotation of orthologous groups of genes. *Nucleic Acids Res.* **36**, 250–254 (2007).
56. Overbeek, R. et al. The subsystems approach to genome annotation and its use in the project to annotate 1,000 genomes. *Nucleic Acids Res.* **33**, 5691–5702 (2005).
57. Kultima, J. R. et al. MOCAT2: a metagenomic assembly, annotation and profiling framework. *Bioinformatics* **32**, 2520–2523 (2016).
58. Finn, R. D. et al. The Pfam protein families database: towards a more sustainable future. *Nucleic Acids Res.* **44**, 279–285 (2015).
59. Lakin, S. M. et al. MEGARes: an antimicrobial database for high throughput sequencing. *Nucleic Acids Res.* **45**, 574–580 (2016).
60. Huang, Y., Gilna, P. & Li, W. Identification of ribosomal RNA genes in metagenomic fragments. *Bioinformatics* **25**, 1338–1340 (2009).
61. Hug, L. A. et al. Banfield, A new view of the tree of life. *Nat. Microb.* **1**, 16048 (2016).
62. Kumar, S., Stecher, G. & Tamura, K. MEGA7: molecular evolutionary genetics analysis version 7.0 for bigger datasets. *Mol. Biol. Evol.* **33**, 1870–1874 (2016).
63. Wu, Y. W., Simmons, B. A. & Singer, S. W. MaxBin 2.0: an automated binning algorithm to recover genomes from multiple metagenomic datasets. *Bioinformatics* **32**, 605–607 (2016).
64. Kang, D. D., Froula, J., Egan, R. & Wang, Z. MetaBAT, an efficient tool for accurately reconstructing single genomes from complex microbial communities. *Peer J.* **3**, e1165 (2015).
65. Parks, D. H., Imelfort, M., Skennerton, C. T., Hugenholtz, P. & Tyson, G. W. CheckM: assessing the quality of microbial genomes recovered from isolates, single cells, and metagenomes. *Genome Res.* **25**, 1043–1055 (2015).
66. Yoon, S. H., Ha, S. M., Lim, J. M., Kwon, S. J. & Chun, J. A large-scale evaluation of algorithms to calculate average nucleotide identity. *Antonie. Van. Leeuwenhoek* **110**, 1281–1286 (2017).
67. Blin, K. et al. antiSMASH 4.0-improvements in chemistry prediction and gene cluster boundary identification. *Nucleic Acids Res.* **45**, 36–41 (2017).
68. Minced: Mining CRISPRs in environmental datasets <https://github.com/ctSkennerton/minced/tree/master>.
69. Grissa, I., Vergnaud, G. & Pourcel, C. CRISPRFinder: a web tool to identify clustered regularly interspaced short palindromic repeats. *Nucleic Acids Res.* **35**, 52–57 (2007).

Acknowledgements

We thank Mr. Huihui Zhu, Ms. Yujia Nie, and Ms. Xin Gong from Novogene for DNA sequencing and technical support. We thank Mr. Bo Yang for technical support in construction of Linux platform. We thank Ms. Qian Ding, Dr. Zhiwu Sun, Dr. Tim Wong, and Dr. Zhaoming Gao for sample collection. The authors are also grateful for English editing by Ms. Alice Cheung. We also thank the fundings provided by the China Ocean Mineral Resources Research and Development Association (COMRRDA17SC01) and the Strategic Priority Research Program of Chinese Academy of Sciences (XDB06010102) to P.Y.Q.

Author contributions

W.Z. and P.Y.Q. designed the project; W.Z. and W.D. performed major metagenome and 16S data analyses; S.B. and V.B.B. provided Linux platform and are involved in data analyses; Y.L., H.F., and H.L. proposed cruises and collected samples; W.D., C.T., R.W., B.P., H.C., and P.L. collected samples and extracted DNA; W.Z., Y.X.L., N.S.W., and P.Y.Q. wrote the paper; P.Y.Q., J.S., and Y.X.L. provided funding support for DNA sequencing and human resources; all the authors discussed and revised the paper.

Additional information

Supplementary Information accompanies this paper at <https://doi.org/10.1038/s41467-019-08463-z>.

Competing interests: The authors declare no competing interests.

Reprints and permission information is available online at <http://npg.nature.com/reprintsandpermissions/>

Journal peer review information: *Nature Communications* thanks Matthias Labrenz, John Eppley and Luis Rodriguez-R for their contribution to the peer review of this work. Peer reviewer reports are available.

Publisher's note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2019

TITLE/ AUTHOR	Big Data Driven Marine Environment Information Forecasting: A Time Series Prediction Network / J. Wen, J. Yang, B. Jiang, H. Song and H. Wang
SOURCE	IEEE Transactions on Fuzzy Systems Volume 29, No. 1, Pages 4-18 (Jan. 2021) https://doi.org/10.1109/TFUZZ.2020.3012393 (Database: IEEE)

Big Data Driven Marine Environment Information Forecasting: A Time Series Prediction Network

Jiabao Wen^{ID}, Jiachen Yang^{ID}, *Member, IEEE*, Bin Jiang^{ID}, Houbing Song^{ID}, *Senior Member, IEEE*, and Huihui Wang^{ID}, *Senior Member, IEEE*

Abstract—The continuous development of industry big data technology requires better computing methods to discover the data value. Information forecast, as an important part of data mining technology, has achieved excellent applications in some industries. However, the existing deviation and redundancy in the data collected by the sensors make it difficult for some methods to accurately predict future information. This article proposes a semisupervised prediction model, which exploits the improved unsupervised clustering algorithm to establish the fuzzy partition function, and then utilize the neural network model to build the information prediction function. The main purpose of this article is to effectively solve the time analysis of massive industry data. In the experimental part, we built a data platform on Spark, and used some marine environmental factor datasets and UCI public datasets as analysis objects. Meanwhile, we analyzed the results of the proposed method compared with other traditional methods, and the running performance on the Spark platform. The results show that the proposed method achieved satisfactory prediction effect.

Index Terms—Big data, forecasting model, fuzzy time series, long short-term memory (LSTM), semisupervised learning.

I. INTRODUCTION

NOWADAYS, people's demand for information has promoted the development of big data in many fields [1], [2]. In the big data framework, there are some specialized tools and methods for data processing. For example, Flume and Logstash [3], [4] have solved the problem of receiving multiple channels of data, and Spark SQL and Spark Streaming have solved the problem of batch processing and stream processing. In terms of methods, data value extraction based on machine learning mining technology helps people better understand the

observation goals. Regarding the big data ecosystem, time series data (TSD) has the largest proportion, and it reflects the rules of observed things [5]. The TSD analysis can not only be applied to intelligent data systems but also provide services for the Internet of Things system [6]. For example, periodical household electricity consumption statistics, peak traffic management, network data time series statistics [7], etc. Therefore, effective analysis of TSD can predict future information changes and make relevant decisions in time to promote the development of the industries.

The research of time series model (TSM) is an important field in machine learning, and analyzes information changes based on intelligent algorithms and observation data [8], [9]. In general, the future trend of observation targets is affected by historical data, and the information does not mutate in a short period of time. Based on this rule, autoregressive integrated moving average (ARIMA) model and Holt-Winters regression model have been applied to short-term TSD mining, including decompose, prediction, classification, clustering, etc. [10]–[12]. These methods based on mathematical modeling are significant for short and near-term forecasting of a single sequence. However, as data attributes and calculation cycles grow, these methods can cause predictions to deviate from reality and make decision errors. Obviously, artificial neural network (ANN) can analyze complex high-dimensional feature datasets by a large number of weight parameters. Regarding the issue of long-period time series, the long short-term memory (LSTM) network in ANN shows an excellent aspect, and it connects continuous time states to obtain long-term and complex time series learning. For example, [13] describes a deep learning model based on LSTM, which obtains powerful spatiotemporal features from wind speed and direction data of neighboring wind farms to predict the wind speed time series of the environment.

Conventional TSM has been widely used in various fields, but there are some problems that have not been solved in practical applications, as follows.

- 1) The accuracy of the forecasting model mainly depends on long-term large amounts of historical data. When the training dataset appears as a small sample, it cannot provide enough experience for TSM. This requires deep mining of limited datasets to find their underlying rules.
- 2) Due to external environmental interference during the information collection process of the sensor, there are damaged data and missing data in the dataset, which will reduce the accuracy of TSM. In addition to the necessary filtering of datasets, the sequence differences brought by

Manuscript received February 27, 2020; revised May 8, 2020, June 19, 2020, and July 10, 2020; accepted July 22, 2020. Date of publication July 28, 2020; date of current version December 30, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 61871283, in part by the Major Civil-Military Integration Project of Tianjin City under Grant 18ZXJMTG00170, and in part by the Natural Science Foundation of Tianjin City under Grant 18JCJQC46400. (*Corresponding author: Jiachen Yang.*)

Jiabao Wen, Jiachen Yang, and Bin Jiang are with the School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China (e-mail: Wen_Jiabao@tju.edu.cn; yangjiachen@tju.edu.cn; jiangbin@tju.edu.cn).

Houbing Song is with the Department of Electrical, Computer, Software, and Systems Engineering, Embry-Riddle Aeronautical University, Daytona Beach, FL 32114 USA (e-mail: h.song@ieee.org).

Huihui Wang is with the Department of Engineering, Jacksonville University, Jacksonville, FL 32211 USA (e-mail: hwang1@ju.edu).

Color versions of one or more of the figures in this article are available online at <https://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TFUZZ.2020.3012393

the unexpected data should be weakened during the TSM process.

- 3) Observing the time series changes of the target state may be accidental. The random external environment will affect short-term changes of the observations, but historical data cannot support the model's analysis of sudden conditions. It is necessary to quickly adapt to the environment and find rules from long-term historical knowledge.

Therefore, we introduce the concept of fuzzy theory into TSD. In 1993, fuzzy time series was first proposed by Song and Chissom [14] who established a fuzzy time series model (FTSM). After that, researchers proposed some improved FTSMs, which have shown advantages in processing TSD based on incomplete and small datasets [15], [16]. The characteristic of FTSM is to blur the time samples to reduce the numerical bias in the observation datasets. FTSM regards the observation data as a fuzzy body, and does not completely trust any one piece of data, and its purpose is to maximize the accuracy of the training model.

In this article, we propose a TSD processing model for Spark-based fuzzy partitioning LSTM networks (SFPLN), which is divided into two parts: supervised learning and unsupervised learning. Considering the nonuniformity of the observed data, we partitioned the data attributes based on the fuzzy c-means (FCM) method, and segment the time series based on the regular triangular fuzzy set [17]. In the supervised learning part, we build a new TSM. Considering the long-term memory ability of the model, we build a training model based on LSTM, and use the output of the unsupervised learning part as the training set of the model. Finally, the output is linearly analyzed through a regression layer. The entire method is run on Spark, and its purpose is to take advantage of Spark resilient distributed dataset's (RDD's) efficient parallel processing [18]. The main contributions of this article are as follows.

- 1) This article proposes a semisupervised fuzzy TSM, which predicts future trends based on the prior knowledge of the target. When dealing with incomplete TSD, it weakens the impact of unexpected data based on fuzzy partitioning. This method is different from the traditional method's excessive reliance on training data, and shows advantages in defective TSD.
- 2) This article partitioning the datasets attributes and time based on fuzzy methods, and establish fuzzy sets, respectively. Then, the fuzzy sets are used for LSTM model training to achieve the learning of long-term historical data. The added fuzzy partitioning solves the problem that traditional methods have difficulty dealing with irregular TSD.
- 3) The SFPLN is built on Spark and applied to some special fields, such as marine environment big data, which shows that the data distribution is sparse and the collection time interval is random. The experimental part of this article has not only been verified on some public datasets, but also has been analyzed in detail on long-term marine hydrological datasets.

The rest of this article is organized as follows. Section II reviews the state-of-the-art related work. Section III provides basic

knowledge. Section IV illustrates the process of the proposed method. Section V gives the experimental studies. Section VI concludes this article.

II. RELATED WORK

Forecasting time series with high accuracy has been a challenging problem. In this section, we first introduce some state-of-the-art TSMs, and then explain the application of fuzzy theory to TSD. Finally, we introduced the situation of big data systems involving fuzzy theory in recent years.

A. Time Series Forecasting Models

In recent years, the development of big data technology in various industries has promoted many studies on information prediction. Lv *et al.* [19] proposed a new deep learning-based traffic flow prediction method that considers the spatial-temporal correlation of traffic flows. It uses a stacked autoencoder model to learn general traffic flow features and trains in a greedy layered manner to address the shortcomings of existing traffic flow prediction methods. Gonzalez-Vidal *et al.* [20] proposed a method to convert a time-dependent database into a structure that standard machine learning algorithms can handle, and then apply different types of feature selection methods to regression tasks. Then, the study is applied to multivariate time-varying sequence data points for energy prediction of intelligent buildings. Haupt and Kosović [21] used physical and dynamical theories and computational intelligence to predict the power conditions of wind and solar resources. Their article describes an example system for utility applications and how to meet utility needs while solving big data problems. Merigaud and Ringwood [22] proposed a framework that uses Gaussian waves as hypotheses. It uses the combination of wave heights at various points in time and space as Gaussian vectors to solve the time series prediction problem of marine environmental information. And the applicability of the spectrum-based predictor in real-time wave elevation prediction is discussed.

B. Time Series Based on Fuzzy Theory

Yang and Liu [23] proposed a time series prediction model based on the hybrid combination of high-order fuzzy cognitive maps (FCMs) with the redundant wavelet transform for processing large-scale nonstationary time series. The redundant nonstationary time series are decomposed into multivariate time series by redundant Haar wavelet transform, and the multivariate time series are accumulated to obtain the predicted time series for each time step. Cai *et al.* [24] proposed an FTSM based on ant colony optimization, which uses ant colony algorithm to appropriately divide the text space to improve prediction performance. Using the auto-regression method instead of the traditional high-order method to make better use of historical information has strong practicability. Chen and Chen [25] proposed a hybrid FTSM based on granular computing for stock price forecasting. This model realizes the prediction of fuzzy time series based on binning-based partition and entropy-based discretization method. Salgado *et al.* [26] proposed a mixed

fuzzy clustering algorithm based on dynamic time regularization distance to solve the problem that time deviation of physiological variables may mask the discovery of sharing patterns and trends among different patients. Efendia *et al.* [27] used triangular fuzzy numbers to take nonstationary stock market data as prediction objects, and established an improved fuzzy random autoregressive model, whose inputs are high-low and single-point data of stock market prices. Cheng and Yang [28] proposed a stock index fuzzy time series prediction model based on rough set rule induction, which uses a rule-based algorithm instead of fuzzy logic relationships to extract prediction rules from time series observation data. Song *et al.* [29] proposed a fuzzy neural network to improve the learning ability of FCM. Their method combines the inference mechanism of traditional models and the determination of membership functions, as well as the quantification of causality. It can automatically construct the FCM model of the system under study based on the data, thereby reducing manual intervention to predict changes in information on fuzzy time series.

C. Fuzzy Technology in Data System

In recent years, fuzzy theory has achieved many successful cases in data systems with its ability to express and quantify uncertainty. Fuzzy theory improves the processing efficiency of big data and brings more possibilities for data mining. G. Selvachandran *et al.* [30] proposed a new Markov complex fuzzy inference system to improve the performance of traditional fuzzy systems and complex fuzzy systems. Among the existing fuzzy inference systems, this article successfully proves that the proposed Mandani cyclic fuzzy inference system has low computational overhead, and proposes a more effective method to deal with TSD and time period phenomena. De Souza *et al.* [31] proposed a fuzzy optimal path forest (OPF), which is an improved version of the standard OPF classifier. It learns the membership of the samples in an unsupervised way and further merges them in supervised training. This information is used to identify the most relevant training samples, thereby improving the classification step. Long Giang *et al.* [32] proposed some new incremental methods using hybrid filter wrappers and fuzzy partitions to blur rough set distances to extend the model's attribute settings with rough sets. Experimental results show that the algorithm significantly reduces the reduction and achieves a higher precision cardinality than other filter increment methods such as IV-FS-FRS-2, IARM, ASS-IAR, IFSA IFSD. Wang *et al.* [33] discussed the challenges of current big data and clarified the application trend of fuzzy sets in big data processing. Meanwhile, the authors analyzed the reasons for the role of fuzzy sets in data systems, and introduced some more complex fuzzy set applications to expand information, intelligence, and related decision-making in various fields such as health care, business, management, and government. Segatori *et al.* [34] proposed a distributed fuzzy decision trees (FDT) learning scheme based on MapReduce programming model to meet the time and space requirements of big data. This scheme generates a strong fuzzy partition for each continuous attribute through a distributed fuzzy discretizer. This work effectively integrates fuzzy theory

into big data systems. Elkano *et al.* [35] proposed a new distributed learning algorithm CFM-BD from scratch, used to build an accurate and compact rule-based big data fuzzy classification system. It implements a fuzzy rule classifier under big data to maintain a good balance between accuracy and interpretability. Pulgar-Rubio *et al.* [36] proposed a new subgroup discovery task algorithm MEFSDBD. The algorithm is developed based on the MapReduce paradigm in Apache Spark and can efficiently process high-dimensional datasets. In fact, the algorithm is the first subgroup discovery algorithm in evolutionary fuzzy systems that approximates big data. It can analyze the quality of each mapped subgroup obtained with respect to the original dataset, thereby improving the quality of these subgroups.

III. PRELIMINARIES AND BACKGROUNDS

In this section, we first introduce the principle of fuzzy time series. Then, we explain the neural network based time series forecasting process in supervised learning in order to prepare for our proposed method. Finally, we describe the Apache Spark cluster computing framework. This framework is exploited in our experiments.

A. Fuzzy Time Series

In 1993, Song *et al.* [29] proposed a definition of fuzzy time series based on fuzzy theory. Let $U = \{u_1, u_2, \dots, u_n\}$ as the domain and define A as the fuzzy set of U , expressed as

$$A = f_A(u_1)/u_1 + f_A(u_2)/u_2 + \dots + f_A(u_n)/u_n \quad (1)$$

where f_A represents a membership function, i.e., $f_A: U \in [0, 1]$. $f_A(u_i)$ represents the fuzzy membership of u_i on the fuzzy set A , i.e., $f_A(u_i) \in [0, 1] (1 \leq i \leq n)$.

Suppose there exists a time domain $t = \{-l, -l+1, \dots, 0, 1, \dots, l-1, l\}$, $S(t) \in R$. $S(t)$ defines a set of fuzzy sets $f_i(t) (i = 1, 2, n)$ and $F(t) = \{f_1(t), f_2(t), \dots, f_n(t)\}$, then we define $F(t)$ is a fuzzy time series on $S(t)$. Because $F(t)$ is a time-varying function of t , its fuzzy domain may also change with t , so $S(t)$ is used as the fuzzy domain. Therefore, $\exists F(t-1) \rightarrow F(t)$.

To implement sample forecasting in fuzzy time series, the sample domain needs to be divided by intervals. For example, the domain $U = [D_{\min} - \sigma', D_{\max} + \sigma'']$ can be divided into m fuzzy subintervals with interval length $\bar{s}$, expressed as

$$m = [(D_{\max} - D_{\min})/\bar{s}] \quad (2)$$

$$\bar{s} = \frac{1}{n-1} \sum_{t=2}^n |S(t) - S(t-1)| \quad (3)$$

where $D_{\max}$ represents the maximum value of the sample data, $D_{\min}$ represents the minimum value of the sample data, n is the number of samples, $S(t)$ is the sample observation at time t , and σ', σ'' are the standard deviation of the training samples.

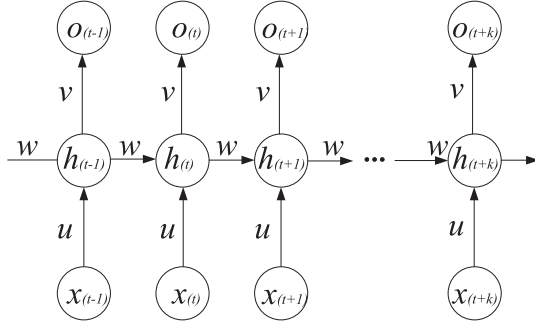


Fig. 1. RNN structural framework.

Then, define the fuzzy set on the domain and the corresponding fuzzy semantic variable A_i . Fuzzy training data to get

$$\begin{cases} A_1 = f_{11}/u_1 + f_{12}/u_2 + \dots + f_{1n}/u_n \\ A_2 = f_{21}/u_1 + f_{22}/u_2 + \dots + f_{2n}/u_n \\ \vdots \\ A_n = f_{n1}/u_1 + f_{n2}/u_2 + \dots + f_{nn}/u_n \end{cases} \quad (4)$$

Finally, establish fuzzy relationships (e.g., $A_i^{t-1} \rightarrow A_i^t \rightarrow A_i^{t+1}$), and predict future changes in sample information.

B. Time Series Modeling Methods

At present, there are mainly two types of time series modeling methods. The first is a based-statistical ARIMA linear model, which predicts future trends according to the prior information change rule. The second is a based-neural network supervised learning model, which trains network weight parameters by historical data to achieve the convergence of forecasting errors. ANN consists of an input layer, a hidden layer, and an output layer, which can effectively solve the linear regression problem. The input layer shows the feature number of the samples, and the output layer is obtained by the linear and nonlinear transformation of the hidden layer, whose logistic regression function shows $h_\phi(x) = (1 + e^{-\phi^T x})^{-1}$.

The recurrent neural network (RNN) in ANN is suitable for modeling time series [37], [38]. Different from ANN in nonstationary series, RNN predicts the change of information by the temporal correlation, as shown in Fig. 1, which represents the update status of network hidden layer weights at different moments. The state of the network at the next moment depends on the state of the last moment, forming a state loop network over time. $x_{(t)}$ represents the sequence input at time t . w represents the weight update with the sequence change. $O_{(t)}$ represents the output at time t . The difference between $O_{(t)}$ and the determined value given by training represents the loss function. The loss of the network is accumulating as the sequence goes on. In short, the network function is expressed as

$$\begin{aligned} x_t &= f(x_{t-1}, \phi) \\ &= f(f(x_{t-2}, \phi), \phi) \\ &= f(f(f(x_{t-3}, \phi), \phi), \phi) \\ &= \dots \\ &= f(f(f(\dots f(x_1, \phi), \dots), \phi), \phi) \end{aligned} \quad (5)$$

where ϕ represents time-independent network parameters, $f(\cdot)$ represents a time-independent mapping function, and x_t represents the sample state at time t .

LSTM, as a variant of RNN, shows more effective forecasting accuracy on complex long time series [39]. Compared with ARIMA, LSTM has the ability to solve the noise and nonlinear problem, and has better forecasting effect on long-term models. However, LSTM is not as good as ARIMA when dealing with univariate data with linear relationship and time dependence.

C. Apache Spark

In 2004, Google proposed the MapReduce framework, which is a cluster-based high-performance parallel big data computing platform that is used to handle parallel operations for large-scale datasets (> 1 TB) [40], [41]. MapReduce is mainly divided into two phases, Map and Reduce, expressed as $\langle key, value \rangle$ [42]. Based on this framework, in 2009, UC Berkeley AMP Lab developed Spark, an open-source cluster computing platform, which is a fast and general-purpose computing engine designed for large-scale data processing [43]. Spark extends the MapReduce calculation model to support interactive query and stream processing modes.

Compared with the way that MapReduce frequent reads and writes hard disks, Spark uses RDD to save a lot of I/O resources [44], [45]. RDDs are expressed as special datasets with fault tolerance mechanisms that are distributed on the nodes of the cluster. Spark stores data distributed across all nodes via Hadoop distributed file system (HDFS) to improve system throughput and distributed storage [46]. The combination of RDD and HDFS improves the computing performance of the Spark framework, and can autonomously avoid single node failures.

Spark uses Zookeeper to provide coordinated management services in the cluster, and uses HBase to store and manage data. While Spark converts the train data to RDD, the train data is divided into small blocks based on random partitioning strategy. Then, these blocks are independently calculated in each node. Spark's built-in machine learning library implements a wide range of machine learning and data mining algorithms for clustering (e.g., k -means), classification (e.g., support vector machine and Bayes), collaborative filtering (e.g., alternating least squares), regression problems (e.g., linear regression and Lasso), etc. [47].

Nowadays, the Spark framework has been extended to the application of big data in some industries.

IV. PROPOSED SEMISUPERVISED TIME SERIES FORECASTING MODEL

A. Unsupervised Fuzzy Partitioning

Partitioning of continuous attributes is a crucial aspect in fuzzy time series and, therefore, should be introduced in detail. FCM is an effective unsupervised clustering algorithm [48]. It uses the mean square approximation of clustering centers to summarize the features of a large number of samples, and selects the center value that satisfies the characteristics of the observation dataset. Among them, fuzzy partitioning based on

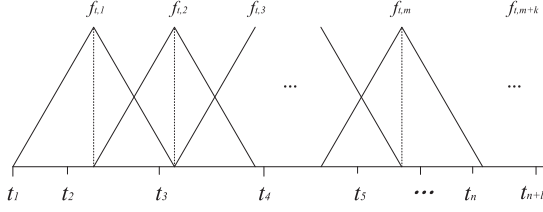


Fig. 2. Association rules for triangular fuzzy numbers.

data features has proven to be very effective. In this section, we propose an unsupervised fuzzy partitioning method (UFPM) for triangular fuzzy partitioning of time series big data.

The proposed UFPM is unsupervised method, which can automatically generate feature classes and perform fuzzy division on time series. This algorithm is based on the characteristic distribution of observation data, and uses the clustering process to divide the observation data into different fuzzy sets. Similar to FCM, this process converges the data to some area. In this article, however, big data exhibits randomness, and the time and attributes of each dataset are nonuniform. In this section, we will discuss UFPM for datasets with unknown timing in big data. In the following, we first recall some cluster definitions and then describe the proposed method.

Let $X = [x_1, x_2, \dots, x_n]$ be the K attribute of the test dataset, and $T = [t_1, t_2, \dots, t_n]$ be n moments of the test dataset in time series. We assume that each element in X corresponds to T . Besides, X exists different features, and T exists different time intervals. And we first perform fuzzy segmentation on X and T , respectively. T is segmented based on the uniform triangle partitioning. Let $F_t = [f_{t,1}, f_{t,2}, \dots, f_{t,m}]$ be the fuzzy set, where m represents the number of fuzzy subsets. Among them, we have

$$f_{a,i} = \{ < t_i, \mu_{f_{t,i}}(t_i), \nu_{f_{t,i}}(t_i) > | t_i \in T \} \quad (6)$$

where $\mu_{f_{t,i}}(t_i)$ represents the degree of t_i belonging to $f_{t,i}$, $\nu_{f_{t,i}}(t_i)$ represents the degree of t_i not belonging to $f_{t,i}$, that is, $\mu_{f_{t,i}}(t_i) : T \rightarrow [0, 1]$, $\nu_{f_{t,i}}(t_i) : T \rightarrow [0, 1]$, and satisfies $0 \leq \mu_{f_{t,i}}(t_i) + \nu_{f_{t,i}}(t_i) \leq 1, \forall t_i \in T$. In Fig. 2, the time series T of the observation data is divided into a plurality of fuzzy subsets. Since T is continuously increasing, the size of its fuzzy set should be infinite. This article uses one day and one month as the interval between two adjacent fuzzy subsets.

The fuzzy partitioning of X is divided into two parts. On the one hand, X is clustered into K classes according to attributes, which uses the FCM method. On the other hand, a new fuzzy rule is established based on the completed fuzzy set partitioning, and its purpose is to effectively distinguish the different attribute features in X . In the following, we will discuss the process of fuzzy partitioning of X in detail.

Step 1: Suppose X is divided into K classes, and satisfies

$$\begin{cases} k_p = \{x_i | \|x_i - v_p\| \leq \|x_i - v_q\|, x_p \in X\}, p \neq q \\ v_p = \sum x_i / |k_p|, x_i \in k_p \end{cases} \quad (7)$$

where $\|\cdot\|$ represents radial distance between samples, and v_p represents the p th clustering center. Generally, the clustering principle depends on the sum of squared error (SSE), which

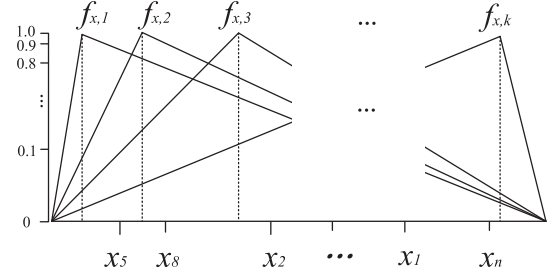


Fig. 3. Membership distribution based on clustering center.

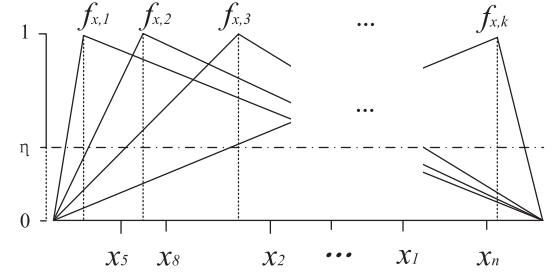


Fig. 4. Threshold partition method (TPM) process diagram.

is expressed as $SSE = \sum_{p=1}^k \sum_{x_i \in k_p} \|x_i - v_p\|^2$. Similar to FCM, the proposed method continuously updates the membership function on the basis of initialization until the cluster center remains unchanged. The clustering loss function based on the membership function is expressed as

$$J_f = \sum_{p=1}^k \sum_{i=1}^n \mu_p(x_i)^c \|x_i - v_p\|^2 \quad (8)$$

where c represents smoothing factor, and $c = 2$ in general. $\mu_p(x_i)$ represents the degree of the sample x_i belonging to the cluster center k_p . To get the optimal clustering result, it satisfies $\partial J_f / \partial v_p = 0$, and $\partial J_f / \partial \mu_p(x_i) = 0$. Then, repeat (9) and (10) to obtain the optimal solution

$$v_p = \frac{\sum_{i=1}^n \mu_p(x_i)^c x_i}{\sum_{i=1}^n \mu_p(x_i)^c} \quad (9)$$

$$\mu_p(x_i) = \frac{\|x_i - v_p\|^{-2/(c-1)}}{\sum_{q=1}^k \|x_i - v_q\|^{-2/(c-1)}}. \quad (10)$$

The proposed UFPM will perform secondary segmentation on the basis of attribute clustering. Let $F_x = [f_{x,1}, f_{x,2}, \dots, f_{x,k}]$ as the fuzzy set. Fig. 3 shows the attributes distribution of X based on the cluster center. It can be assumed to be an unconventional triangular fuzzy rule, which satisfies $\mu_{f_{x,j}}(x_i) \neq 0$ and

$$\begin{cases} 0 < \mu_{f_{x,j}}(x_i) < 1 \quad \forall i \in n, j \in k \\ \sum_{j=1}^k \mu_{f_{x,j}}(x_i) = 1 \quad \forall i \in n \end{cases} \quad (11)$$

To conveniently use the regular fuzzy rules, we set a threshold η to split the membership function horizontally as shown in Fig. 4. The purpose is to effectively distinguish the correlation of X , which is convenient for later time series calculation. Algorithm 1 shows the basic process of the TPM.

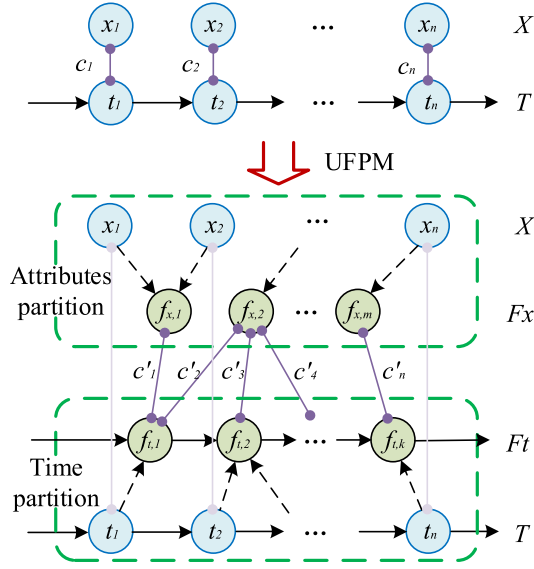


Fig. 5. Process of obtaining new sequence based on UFPM.

Algorithm 1: The Process of TPM.**Input:**

Observation dataset X ; Fuzzy set F_x ; Threshold value η

Output:

New fuzzy membership function $\mu_{f_{x,j}}(x_i)$

- 1: Initialize the original fuzzy membership function $\mu_{f_{x,j}}(x_i)$
- 2: **Thresholdpartition:**
- 3: **for** each $i \in n, j \in k$ **do**
- 4: **if** $\mu_{f_{x,j}}(x_i) > \eta$ **then**
- 5: $\mu_{f_{x,j}}(x_i) = \mu_{f_{x,j}}(x_i) - \eta$
- 6: **else**
- 7: $\mu_{f_{x,j}}(x_i) = 0$
- 8: **end if**
- 9: **end for**
- 10: **Normalization:**
- 11: **for** each $i \in n, j \in k$ **do**
- 12: $\mu_{f_{x,j}}(x_i) = \frac{\mu_{f_{x,j}}(x_i)}{\sum_{j=1}^k \mu_{f_{x,j}}(x_i)}$
- 13: **end for**

From the above, we get the UFPM method flow as shown in Algorithm 2, which consists of two parts, time and attribute partition, each of which has been described in detail above. Fig. 5 describes the process of obtaining a new fuzzy sequence based on UFPM. The blue unit represents the initial sequence distribution, and the green unit represents the new fuzzy partition sequence. Its significance is to avoid model fitting caused by large erroneous collected values when training the prediction model to improve training accuracy.

B. LSTM Fuzzy Learning

In this section, we illustrate how to construct a fuzzy LSTM, which is mainly composed of fuzzy input (FI) and fuzzy transmission (FT) of hidden layers. Among them, FI is obtained by

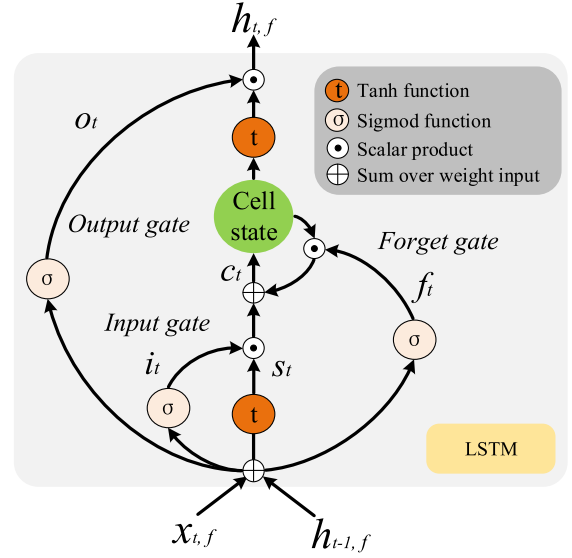


Fig. 6. LSTM network structure.

Algorithm 2: UFPM.

Input: Dataset includes $t = t_1, t_2, \dots, t_n$ and all attributes

Output: New fuzzy-dataset

- 1: **Time series partition :**
- 2: Create regular triangular fuzzy sets
- 3: Define membership function intervals (e.g. day, etc.)
- 4: Get time fuzzy set F_t
- 5: **Attributes partition :**
- 6: **repeat**
- 7: Get i th attributes k cluster centers based on FCM
- 8: Let k cluster centers as fuzzy members
- 9: Get F_x based on Algorithm 1
- 10: **until** All attributes are partitioned
- 11: Align time series to generate new fuzzy-dataset
- 12: **return** fuzzy-dataset

fuzzy partitioning of time series information, and FT is obtained by FI and the recurrent network learning together.

LSTM as a special type of RNN, it utilizes cyclic training between network neurons to ensure the model's memory of information [49]. Unlike other neural network method, RNN is good at dealing with time series problems. It can use the previous state and the current state to infer the information trend at the next time. However, RNN can only save the short-term change memory, it cannot solve the problem of long-term dependence of information. LSTM is designed to solve this problem as shown in Fig. 6, which shows the input gate, output gate, and forget gate. Among them, the forget gate uses the sigmoid function to determine what information needs to be forgotten. The input gate is used to save the input information to the status unit. The output gate is used to output the result of weighted information.

To improve the forecasting accuracy of LSTM, we build a fuzzy time series reasoning model, which blurs the input of the

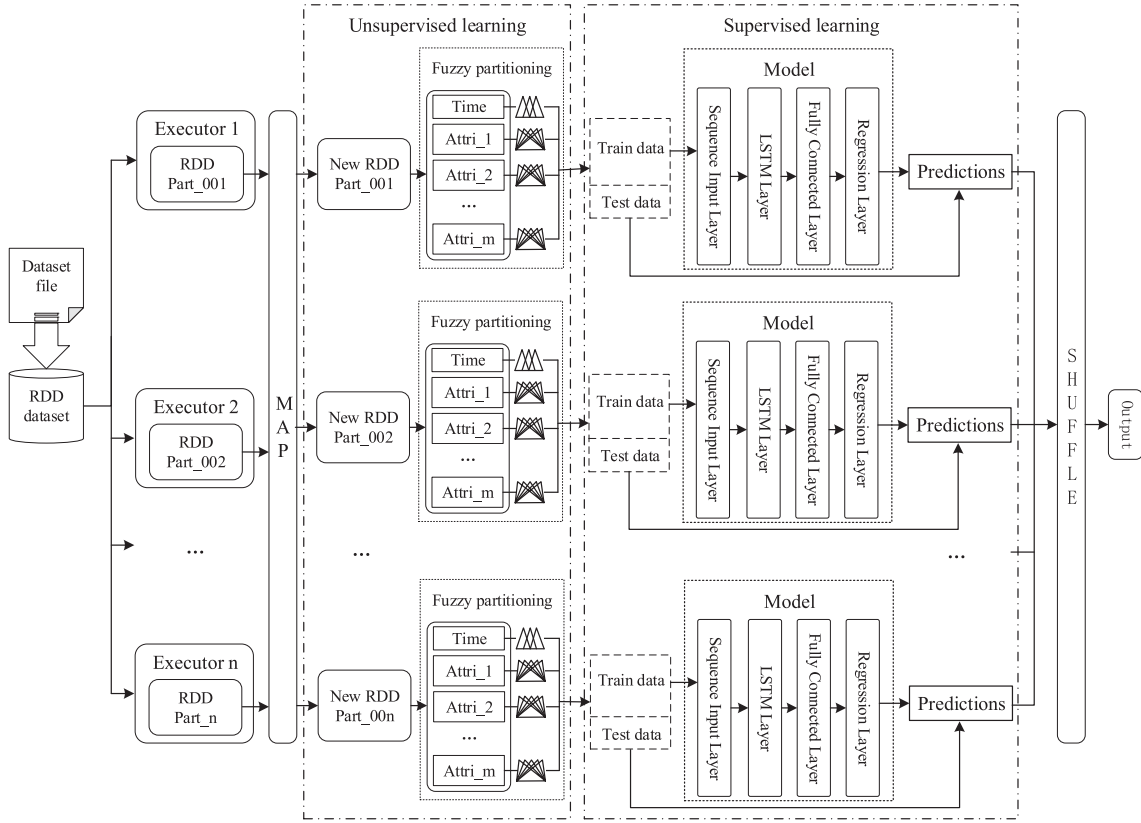


Fig. 7. Framework of SFPLN, which is mainly composed of two parts: unsupervised learning and supervised learning.

network. The FI obtained by Algorithm 1 is

$$\tilde{x}_{t,f} = f(x_t, f_{x,t}, \eta). \quad (12)$$

Hence, the FT inference process based on LSTM network is as follows:

$$i_t = \sigma(w_{ix}\tilde{x}_{t,f} + w_{ih}h_{t-1,f} + b_i) \quad (13)$$

$$f_t = \sigma(w_{fx}\tilde{x}_{t,f} + w_{fh}h_{t-1,f} + b_f) \quad (14)$$

$$o_t = \sigma(w_{ox}\tilde{x}_{t,f} + w_{oh}h_{t-1,f} + b_o) \quad (15)$$

$$s_t = \tanh(w_{sx}\tilde{x}_{t,f} + w_{sh}h_{t-1,f} + b_s) \quad (16)$$

$$c_t = s_t \odot i_t + c_{t-1} \odot f_t \quad (17)$$

$$h_{t,f} = \tanh(c_t) \odot o_t \quad (18)$$

where w_{ix} , w_{fx} , w_{ox} , and w_{sx} are weight matrices to be connected to the input with respect to the input gate, forget gate, output gate, and cell unit; $\tilde{x}_{t,f}$ represents the FI at time t ; $h_{t-1,f}$ represents the FI between hidden layers at time $t-1$; b_i , b_f , b_o , and b_s are the deviation, and expressed as diagonal matrices; c_t represents cell output state; i_t , f_t , and o_t are the output of three gates; and $\odot$ is the scalar product of two vectors. In this article, we mainly deal with stationary information changes based on time series. And using stacked LSTM network as a model, which can deeply learn the hidden laws of data over time.

C. Semisupervised Forecasting Approach

In this section, we mainly explain how to use predictive models to achieve TSD analysis of industry big data, and propose a semisupervised SFPLN. It is mainly divided into three parts. First, RDD parallel processing of industry datasets is implemented on Spark. Second, fuzzification of prediction model input data. Third, the establishment of TSM based on LSTM network framework.

The RDD partition function in Spark implements a large amount of data processing. RDD is considered as a read-only collection of partition records. Every RDD is divided into multiple partitions, each partition representing a dataset fragment. We construct the RDD based on the file system, and then read the data files in HDFS. In Fig. 7, the dataset file is divided into n parts for execution, and each RDD part generates a new RDD based on the MAP function. The original TSD has problems such as sequence deletion and attribute value deviation. We preprocess each RDD by Algorithm 3, which depends on adjacent TSD to fill the missing information. It can be implemented in Dataframe and window functions derived from Spark RDD classes. In Algorithm 3, S_i represents the original dataset. S'_i represents the preprocessed training dataset. df represents the established RDD subset. dv represents the difference between the upper and lower quantiles of the data, and it is used to eliminate outliers in the dataset. Then, $\text{map}()$ is used to detect the null data, and $\text{fit}()$ is used to fill in the missing points that detected or removed.

Algorithm 3: RDD Preprocess Method.**Input:** Original dataset S_i **Output:** RDD trained dataset S'_i

```

1: package com.dkl.leanring.spark.ml
2: import
   org.apache.spark.ml.regression.LinearRegression
3: import org.apache.spark.ml.linalg.Vectors
4: import org.apache.spark.sql.SparkSession
5: for  $i = 0$  to  $n$  do
6:    $df = spark.createDataFrame(S_i)$ 
7:   Detect duplicate data  $\leftarrow df.distinct()$ 
8:   Remove duplicate data  $\leftarrow df.dropDuplicates()$ 
9:   Detect null data  $\leftarrow df.map(\lambda row :$ 
   (row['lable'], sum([x == None for x in
   row]))).collect()
10:   $dv = \text{upper quartile} - \text{lower quartile}$ 
11:   $Outliers \in [(-\infty, normal - 1.5dv) \cup (normal +$ 
    $1.5dv, \infty)]$ 
12:  Remove Outliers from  $df$ 
13:   $val lr = \text{new LinearRegression}()$ 
14:  .setMaxIter(10000)
15:  .setRegParam(0.3)
16:  .setElasticNetParam(0.8)
17:   $S'_i = lr.fit(df)$ 
18: end for
19: return  $S'_i$ 

```

We use Algorithm 3 to partitioning newly defined RDDs, including time series partition and attribute partition. In this process, we use the method in Section IV-A to obtain the attribute fuzzy set and temporal fuzzy set, respectively. They are used as training and test datasets for follow-up models. The purpose is to regulate the distribution of irregular time series and weaken the influence of error values in attribute data in order to train SFPLN more accurately. The unsupervised learning part of the SFPLN framework consists of UFPM, which takes little time to improve the reliability of the datasets. It is important for the analysis of large and long-term samples of industries big data.

Fig. 7 shows the frame structure of SFPLN data processing, which is mainly composed of two parts: unsupervised learning and supervised learning. Before processing, the RDD dataset is divided into multiple executors in parallel, and then preprocessed with MAP module to generate new RDD files. In the unsupervised learning part, the new RDD files are fuzzy partitioned based on UFPM method to generate regular serialized information (RSI), which is transmitted to the next process. The supervised learning part shows a neural network model, which includes a sequence input layer, an LSTM layer, a fully connected layer, and a regression layer. RSI is divided to train dataset and test dataset, which are obtained by Algorithm 2. In Fig. 7, considering that RSI shows fuzzy serialization structure, a regression layer and an antifuzzy function are added to achieve long-term prediction of regular information.

TABLE I
DESCRIPTION OF DATASET

#DataSets	Characteristics			
	Type	Size	Date	Instances
Wod_Tem01	Text.csv	1.18G	1973-2000	1048576
Wod_Tem02	Text.csv	80M	1995-2013	244369
Wod_Sal01	Text.csv	96M	1994-2012	145957
Wod_Sal02	Text.csv	82M	1994-2013	134955
Wod_Oxy	Text.csv	23.8M	1991-2012	132056
Wod_Phos	Text.csv	9.4M	1995-2014	122954
Wod_Sil	Text.csv	17.8M	1994-2012	123606
Wod_NO3	Text.csv	17.3M	1999-2013	122384
Wod_Ch1	Text.csv	6.7M	1994-2012	70287
Wod_Al1	Text.csv	5.2M	1994-2012	68852

V. EXPERIMENTAL STUDY

In this section, we conducted some experiments to prove the effectiveness of the proposed SFPLN method, focusing on the following three crucial aspects.

- 1) Analyze the prediction performance of the proposed model.
- 2) The applicability of SFPLN to industry big data is verified based on marine environmental data.
- 3) The extensibility of SFPLN is discussed.

All experiments were run on the server side, which consisted of four distributed nodes. Each node is equipped with CPU (Intel Core i7 CPU 2.2 GHz, 8 threads), 16 GB of RAM and 1 TB hard drive. All nodes run on CentOS 7 system, and 1 TB of marine environment data is backed up on HDFS.

A. Experimental Datasets

Considering the characteristics of time series information and the industrial applicability of the proposed method, we apply the marine environment datasets as experimental data. Effective marine data processing is an important issue in marine monitoring applications. Some published researches have proposed some methods for efficient processing of marine environmental data [50], [51]. A variety of seawater information changes periodically, which is beneficial to test the efficiency and accuracy of prediction methods [52]. In Table I, ten datasets on the marine environment were downloaded from the NOAA,¹ including seawater temperature, salinity, oxygen content, phosphorus content, silicon content, nitrogen content, chlorophyll, and alkalinity. These datasets are all in the same text format, which contains the acquisition time (e.g., year, month, and day), position (e.g., depth, longitude and latitude), Cruise_ID, ISO_country, WOD_unique. To reflect the prediction effect of the proposed model on industry big data, we use a large amount of samples as the model training sets.

Considering the irregular distribution of time information, we use UFPM to preprocess the data to obtain fuzzy and uniform information. Regarding the impact of climate in different sea

¹<https://www.nodc.noaa.gov/>

TABLE II
FUZZY PARTITIONS RESULTS BASED ON UFPM

#DataSets	#Range (Unit)	#K	#Partitions
Wod_Tem	[15.52, 29.84] (°C)	15	#01:15.52-17.17 #02:17.17-18.23 #03:18.23-19.40 #04:19.40-20.44 #05:20.44-21.28 #06:21.28-22.01 #07:22.01-22.97 #08:22.97-23.93 #09:23.93-24.75 #10:24.75-26.04 #11:26.04-27.13 #12:27.13-28.02 #13:28.02-28.91 #14:28.91-29.84
Wod_Sal	[29.11, 34.87] (PSU)	15	#01:29.11-30.54 #02:30.54-31.23 #03:31.23-31.87 #04:31.87-32.36 #05:32.36-32.71 #06:32.71-33.12 #07:33.12-33.42 #08:33.42-33.73 #09:33.73-33.97 #10:33.97-34.22 #11:34.22-34.45 #12:34.45-34.60 #13:34.60-34.71 #14:34.71-34.87
Wod_Oxy	[3.84, 8.55] (mg/L)	10	#01:3.841-4.515 #02:4.515-4.832 #03:4.832-5.124 #04:5.124-5.429 #05:5.429-5.677 #06:5.677-6.083 #07:6.083-6.492 #08:6.492-7.130 #09:7.130-8.551
Wod_Phos	[0.03, 1.91] (mg/L)	8	#01:0.032-0.123 #02:0.123-0.245 #03:0.245-0.377 #04:0.377-0.523 #05:0.523-0.741 #06:0.741-1.050 #07:1.050-1.912
Wod_Sil	[0.64, 21.64] (mg/L)	15	#01:0.642-1.744 #02:1.744-2.866 #03:2.866-3.778 #04:3.778-4.661 #05:4.661-5.482 #06:5.482-6.402 #07:6.402-7.373 #08:7.373-8.521 #09:8.521-9.797 #10:9.797-11.25 #11:11.25-12.91 #12:12.91-14.88 #13:14.88-17.95 #14:17.95-21.64
Wod_NO3	[0.1, 31.49] (mg/L)	15	#01:0.101-0.473 #02:0.473-0.962 #03:0.962-1.528 #04:1.528-2.016 #05:2.016-3.016 #06:3.016-4.001 #07:4.001-5.018 #08:5.018-6.162 #09:6.162-8.004 #10:8.004-9.966 #11:9.966-12.55 #12:12.55-15.75 #13:15.75-21.58 #14:21.58-31.49
Wod_Ch1	[0.07, 18.16] (mg/L)	15	#01:0.074-0.193 #02:0.193-0.297 #03:0.297-0.395 #04:0.395-0.496 #05:0.496-0.608 #06:0.608-0.712 #07:0.712-0.884 #08:0.884-1.016 #09:1.016-1.413 #10:1.413-2.077 #11:2.077-4.826 #12:4.826-8.163 #13:8.163-13.32 #14:13.32-18.16
Wod_Alk	[1.97, 2.33] (mg/L)	8	#01:1.969-2.096 #02:2.096-2.178 #03:2.178-2.217 #04:2.217-2.263 #05:2.263-2.294 #06:2.294-2.323 #07:2.323-2.332

areas, the area we selected from the dataset is N30;55;° to N33;20;°, E128;92;° to E132;63;°, and 0 to −20 m depth.

B. Performance Analysis

1) *Evaluation Index*: To illustrate the effectiveness of the proposed model, we adopted the following indicators as evaluation criteria. Assume that the actual value is x and the predicted value is $p(x)$. Since the research is a regression problem, the root-mean-squared error (RMSE) and mean absolute error (MAE) are expressed as

$$\text{RMSE} = \sqrt{\frac{1}{m} \sum_{i=1}^m (x_i - p(x_i))^2} \quad (19)$$

$$\text{MAE} = \frac{1}{m} \sum_{i=1}^m |x_i - p(x_i)| \quad (20)$$

where m represents the number of samples. RMSE can represent the loss function of regression problem. Compared to MAE, RMSE is more sensitive to outliers. In addition, R -squared is used to evaluate the accuracy of prediction, as follows:

$$R^2 = 1 - \frac{\sum_i (p(x_i) - x_i)^2}{\sum_i (\bar{x}_i - x_i)^2}. \quad (21)$$

2) *Discussion of Settings*: Based on the above description, we analyze the performance of SFPLN through some experiments. Using UFPM to perform fuzzy segmentation on the dataset, classify irregular random sequences into K classes, and specify fuzzy membership functions according to K cluster centers. In Table II, we have selected some data between N30;55;° and N33;20;°, because the attributes changes in the four seasons are more obvious compared to the equatorial, north and south poles. They were divided into $K - 1$ fuzzy intervals based on the calculated cluster centers. Then, UFPM is used to fuzzy partition the data to correspond to the membership of each group of data.

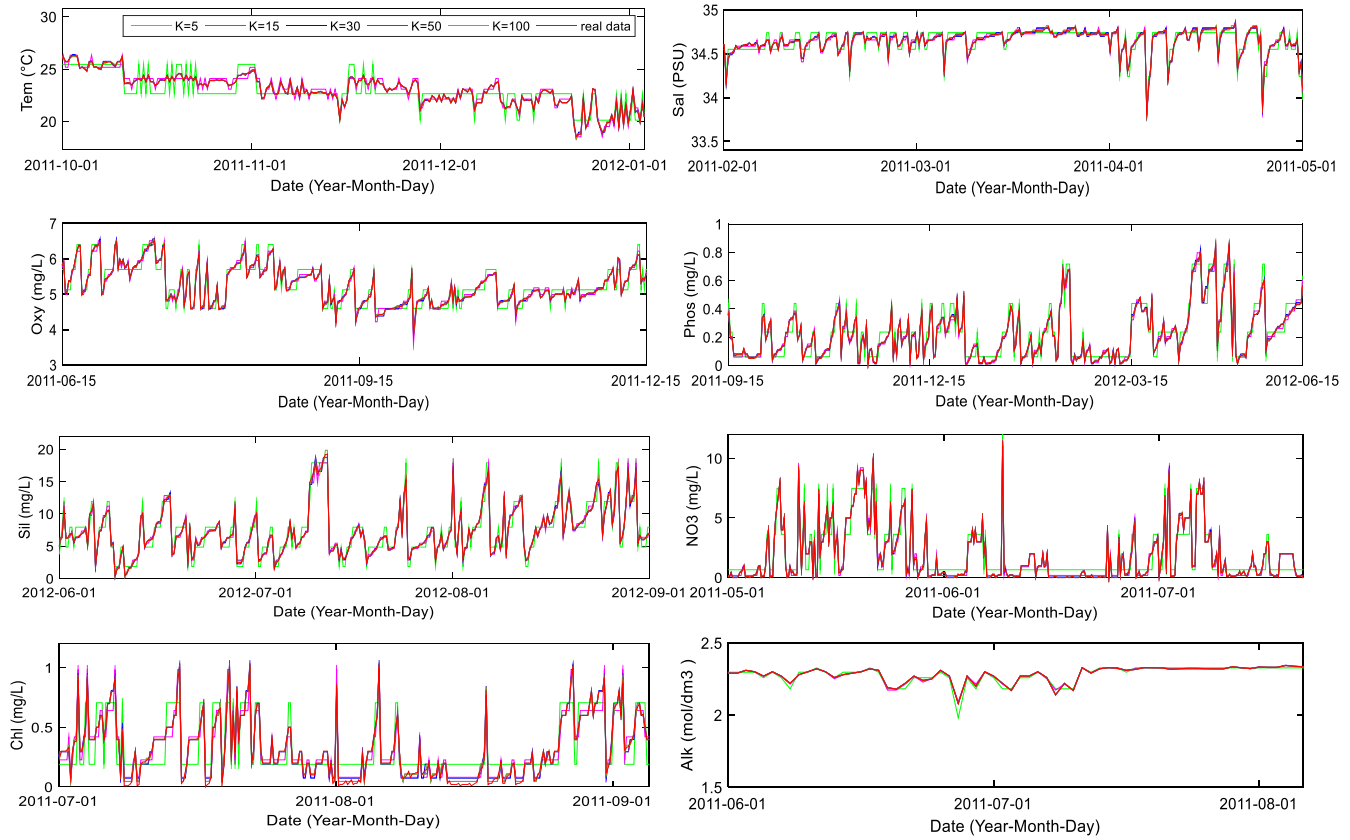
Table II lists that the distribution of seawater temperature and salinity is uniform, which means that the difference between the partition intervals is close. This kind of uniform information can be easily divided into different levels of changes, and the gradient method can be used to crudely predict their changing trends. However, unlike temperature and salinity, the distribution of silicon and nitrogen is nonuniform. For example, the interval difference of Wod_NO3's partition # 01 is $0.47 - 0.10 = 0.37$,

TABLE III
COMPARISON OF RESULTS BASED ON DIFFERENT K

#DataSets	#K	RMSE	MAE	#DataSets	#K	RMSE	MAE
Wod_Tem	15	0.936	0.796	Wod_Sal	10	0.923	0.748
	30	1.318	1.140		15	1.007	0.848
	40	1.203	1.036		20	1.117	0.943
Wod_Oxy	10	1.139	1.026	Wod_Phos	8	0.863	0.643
	15	1.383	1.131		15	0.682	0.500
	20	1.224	1.011		20	0.982	0.825
Wod_Sil	10	2.167	1.830	Wod_NO3	10	1.942	1.503
	15	1.823	1.203		15	3.264	2.989
	20	3.221	2.986		20	4.356	3.236
Wod_Ch1	10	1.323	0.922	Wod_Alk	8	0.242	0.158
	15	1.287	1.068		15	0.385	0.262
	20	1.212	0.992		20	0.741	0.741

and the interval difference of # 14 is $31.49 - 21.58 = 9.91$. This means that the value of the attribute nitrogen in seawater normally remains below 21.58, and only a few cases will be greater than 21.58. Gradient method will appear large errors and abrupt changes when solving such problems. SFPLN reduces the gradient difference between attribute values of different sizes in order to reduce the prediction error caused by information mutation.

From the above, UFPM is an unsupervised learning model, which improves the prediction accuracy by optimizing the number of K . Fig. 8 shows a comparison of the processed data at different K values. The red line indicates the true data value. The remaining color lines represent the processed data, which is obtained from the raw data divided into K classes. According to UFPM, the original data is divided into $K - 1$ intervals. It can be seen that as the value of K is larger, the processed data is closer to the real data. However, as the training data of the prediction model, if the value of K is too large, the collection errors existing in the real data will be reproduced. In Table III, we exemplify some comparative experiments to illustrate the prediction effect under different datasets. It is worth noting that the value of K depends on the predicted object. If the range of the prediction object is large, the value of K is large, otherwise it is small.

Fig. 8. Comparison of the processed data at different K values.TABLE IV
METHOD PARAMETER SETTING

Method	Parameters
EXS	Holt function
ARIMA	$p=2$ $d=1$ $q=2$
LSTM	Epoch=400 Units=200 Learn rate=0.005
SFPLN	$K=5\sim 50$ Epoch=400 Units=200 Learn rate=0.005

The results show that the determination of the number of fuzzy partitions K can optimize the prediction performance of the prediction model. The proposed model, as a semisupervised machine learning method, can analyze the data distribution and optimize the calculation process according to the set parameters.

C. Compare Different Methods

To illustrate the effectiveness of the proposed method, we compared some of the other time series prediction methods. Table IV lists the main parameter settings of each method, including exponential smoothing (EXS), ARIMA model, and LSTM networks. In addition, this experiment uses the same dataset and the same simulation environment.

Table V lists the SFPLN training process settings for each dataset. We select about 50000 samples in each dataset as the training set. The time and space parameters of each variable are set as input, and each environmental variable is set as output.

Since SFPLN is a semisupervised learning method, we set the K value between 5 and 50 during the fuzzy segmentation process. Then, the fuzzy segmented dataset is passed to the LSTM network layer through the sequence input layer. After training the parameters, the information is passed to the regression layer through the fully connected layer to produce prediction results. Table VI lists the comparison of the experimental results of the four methods based on the datasets in this article. The change of some marine factors is not satisfied with the regular time series distribution, which leads to poor prediction of some datasets. For example, seawater alkalinity is more dependent on regional changes than time. Silicon, nitrogen, and phosphorus content are not only affected by human pollution but also by random mesoscale vortices. The results show that the proposed method based on most of the datasets have better prediction effect than the other methods. Among them, EXS and ARIMA are both suitable for stationary time series, and they have better results in predicting regular information. Compared with the proposed method, the LSTM network lacks the fuzzy partitions process, which can explain the effect comparison between the improved method and the original method. Similar to LSTM, the proposed method also requires a large number of training sets. It shows that the prediction effect is significantly better than other algorithms on the Wod_Tem and Wod_Sal datasets. From Table VI, except for the Wod_Ch1 and Wod_Sil datasets, the prediction accuracy of SFPLN in the other datasets is the highest. Regarding the Wod_Ch1 and Wod_Sil datasets, we discuss the contingency of

TABLE V
NETWORK MODEL TRAINING SETUP

Dataset	Train/Valid	Test	Input	Output	Parameters
Wod_tem	50234/16372	18035	Year, month, day, hour, longitude, latitude, depth	Temperature value	Fuzzy partitioning (K=5 to 50), Sequence Input Layer (InputSize=7), Fully Connected Layer (Weights = 1×100 single, WeightLearnRateFactor=1, BiasLearnRateFactor=10), LSTM Layer (batchsize=128, epochs=400, learn rate=0.005, numFeatures=7, numResponses = 1, numHiddenUnits = 100×2 , LearnRateDropPeriod=125, LearnRateDropFactor=0.2), Regression Layer (LossFunction = 'mean-squared-error')
Wod_Sal	48964/16882	17230		Salinity value	
Wod_Oxy	47320/15340	16334		Oxygen content	
Wod_Phos	47953/15674	15924		Phosphorus content	
Wod_Sil	45031/14354	14650		Silicon content	
Wod_NO3	44357/14032	14328		Nitrogen content	
Wod_Ch1	44935/14011	14208		Chlorophyll	
Wod_Alkal	42018/13802	13427		Alkalinity	

TABLE VI
COMPARISON OF RESULTS BASED ON DIFFERENT METHODS IN MARINE ENVIRONMENTAL DATASETS

#Data Sets	EXS			ARIMA			LSTM			SFPLN		
	RMSE	MAE	R^2	RMSE	MAE	R^2	RMSE	MAE	R^2	RMSE	MAE	R^2
Wod_Tem	1.073	0.743	0.927	1.057	0.796	0.929	2.171	1.529	0.842	0.9366	0.736	0.933
Wod_Sal	0.707	0.343	0.536	0.681	0.372	0.569	1.064	0.845	0.502	0.571	0.354	0.623
Wod_Oxy	0.406	0.215	0.721	0.394	0.221	0.736	0.997	0.809	0.649	0.383	0.131	0.761
Wod_Phos	0.196	0.111	0.458	0.188	0.114	0.504	0.203	0.142	0.469	0.182	0.101	0.536
Wod_Sil	3.057	1.899	0.467	2.910	1.887	0.517	1.497	1.242	0.531	1.823	1.203	0.522
Wod_NO3	2.956	1.736	0.527	2.836	1.706	0.565	2.0727	1.593	0.651	1.942	1.503	0.698
Wod_Ch1	0.868	0.228	0.575	0.830	0.236	0.612	1.181	0.982	0.597	1.287	1.068	0.569
Wod_Alkal	0.069	0.045	0.235	0.385	0.262	0.341	0.099	0.742	0.276	0.068	0.043	0.472

training set selection from the research results. The long-term sequence information of big data is uncertain, and the growth of environmental factors is irregular, which leads to low accuracy of the training model. In Fig. 9, we exemplify the calculation results of temperature, salinity, oxygen, and phosphorus based on SFPLN to illustrate the time series prediction of different factors. The blue line represents real data on environmental factors over a two-year period, and the red line represents short-term forecast data. In general, the method in this article shows advantages in information prediction, and it can make up for some of the shortcomings of traditional methods.

In Fig. 10, we compare five datasets with better prediction effects when the SFPLN sets different K values. When K is appropriate, the best prediction effect can be achieved. The reason is that when K is small, the datasets will be roughly divided into few types, resulting in low learning efficiency of the network model. And when K is large, the interference and errors of the datasets can hardly be ignored, resulting in low accuracy of the network model.

D. UCI Extension Experiments

In order to illustrate the effectiveness of the method in this article, we cited some UCI public datasets for extended experiments. These datasets include PM2.5 data of five Chinese cities datasets, appliances energy prediction (AEP) and sales transactions dataset weekly (SYDW). Table VII lists the detailed attributes of these datasets. Considering the applicability of the proposed method, we have introduced more weather datasets such as PM2.5, because the weather changes show natural regularity relative to household electricity and commodity sales.

TABLE VII
DESCRIPTION OF UCI DATASET

#DataSets	Characteristics			
	Type	Size	Date	Instances
BJ PM2.5	Text.csv	1.91M	2010-2015	52585
CD PM2.5	Text.csv	2.86M	2010-2015	52585
GZ PM2.5	Text.csv	3.19M	2010-2015	52585
SH PM2.5	Text.csv	2.9M	2010-2015	52585
SY PM2.5	Text.csv	2.89M	2010-2015	52585
AEP	Text.csv	11.4M	-	19737
SYDW	Text.csv	0.305M	-	812

Similar to the above marine environment experiment, we also compared the simulation effects of the four methods. The parameter settings of these algorithms are shown in Table IV. When processing the PM2.5 dataset, EXS represents a one-dimensional information prediction method, and its dependent variable is set to the regional PM2.5 value. Regarding the ARIMA method, we set the time (year, month and day), dew point, humidity, temperature, and air pressure as independent variables, and set the PM2.5 value as the dependent variable. Meanwhile, set the time, dew point, humidity, temperature, and air pressure as the training input of LSTM and SFPLN network models, and PM2.5 as the training output. The purpose is to make full use of periodically changing atmospheric environmental factors to predict PM2.5 values. When processing AEP dataset, the power consumption is used as the dependent variable of EXP. And the independent variables of ARIMA are set to temperature, humidity, light, and time, and the dependent variable is power

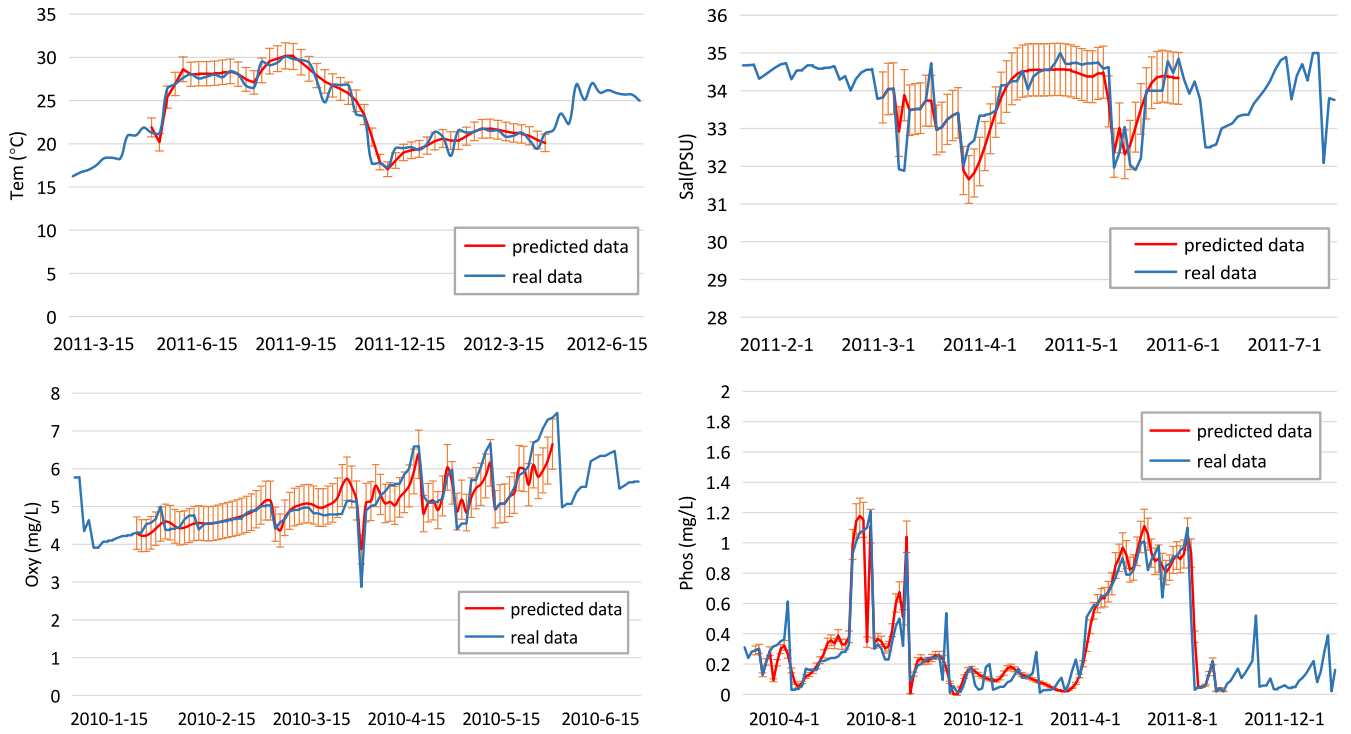


Fig. 9. Prediction results based on SFPLN. The yellow line represents the error range (the temperature error range is set to 5%, the salinity error range is set to 2%, the oxygen error range is set to 10%, and the phosphorus error range is set to 8%).

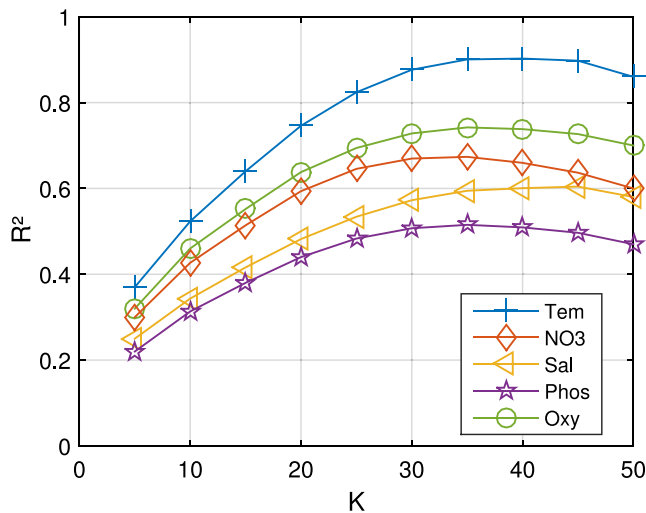


Fig. 10. Comparison of R squared.

consumption. At the same time, the temperature, humidity, light, and time are set to the training input of the LSTM and SFPLN network models, respectively, and the power consumption is set to the training output. When processing the STDW dataset, the specified product consumption is used as the dependent variable of EXP. The independent variable of ARIMA is set to time and consumption of unspecified products, and the dependent variable is the sales volume of specified products. Simultaneous, the time and consumption of unspecified products are the training inputs of the LSTM and SFPLN network models, respectively,

and the specified product sales volume is set as the training output.

In the prediction process, we randomly extracted information from different datasets as experimental data. Table VIII lists the prediction results of various methods for the experiment. It can be seen that the proposed method has achieved a good effect in the prediction of PM2.5. The reason is that the SFPLN method divides the information of different value ranges in a limited space, thereby weakening the influence of interference and discrete data on the results. When simulating the GZ PM2.5 dataset, the prediction effect of the LSTM network is higher than that of SFPLN, which may be due to the randomness of the data. When dealing with AEP and STD, information is greatly affected by human factors, so it will cause unstable prediction results. In general, the proposed method effectively improves the prediction accuracy compared to traditional methods, and reduces the error through fuzzy partition.

E. System Performance Evaluation

Considering the time consumption of the built Spark system, we compared the running time of the dataset in Table I under different threads of Spark, as shown in Fig. 11. This comparison excludes the training time of the supervised learning network model. As Spark workers increase the number of processing threads, the runtime required to generate models decreases. However, when there are more than 3 threads, there is little difference in the calculation time reduction of all datasets.

In this article, we performed three operations to compare with MapReduce in terms of data size and number of nodes.

TABLE VIII
COMPARISON OF RESULTS BASED ON DIFFERENT METHODS IN UCI DATASETS

#Data Sets	EXS			ARIMA			LSTM			SFPLN		
	RMSE	MAE	R^2	RMSE	MAE	R^2	RMSE	MAE	R^2	RMSE	MAE	R^2
BJ PM2.5	13.397	8.611	0.484	4.658	3.420	0.731	0.712	0.589	0.884	0.302	0.183	0.931
CD PM2.5	24.502	12.035	0.619	3.985	2.105	0.650	0.802	0.721	0.889	0.532	0.467	0.901
GZ PM2.5	14.287	7.662	0.509	4.634	3.575	0.592	0.651	0.557	0.875	0.421	0.261	0.912
SH PM2.5	12.579	7.407	0.518	4.419	3.241	0.603	1.238	1.064	0.822	0.835	0.732	0.885
SY PM2.5	16.959	9.233	0.595	4.113	3.033	0.634	1.512	1.324	0.782	0.714	0.622	0.876
AEP	14.567	12.611	0.831	14.571	12.617	0.495	13.423	10.775	0.842	2.322	1.642	0.914
STDW	8.522	5.036	0.605	8.479	5.031	0.611	4.021	2.094	0.826	4.324	2.341	0.743

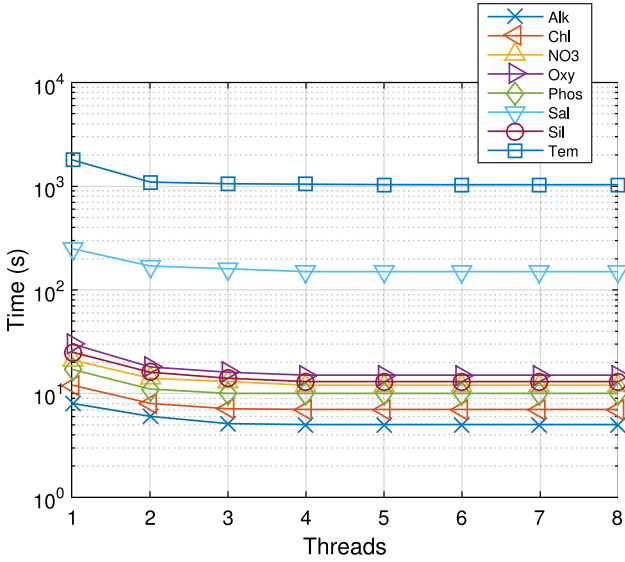


Fig. 11. Time consumption of datasets under different threads.

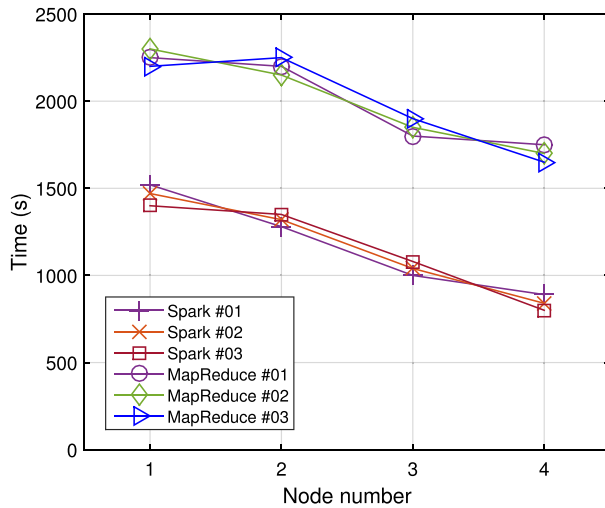


Fig. 12. Comparative analysis of the number of computing nodes.

The result of this comparison is shown in Fig. 12. Hadoop MapReduce uses a multiprocess model, whereas Spark uses a multithreaded model. We analyze the time comparison of the two calculation modes under different settlement nodes. The

MapReduce multiprocess model facilitates detailed control of the resources occupied by each task, but consumes more startup time and is not suitable for running low-latency types of jobs. All tasks on the Spark node can run in an Executor, and the resources occupied by the Executor can be continuously used by multiple batches of tasks. For applications with a large number of tasks, the running time can be greatly reduced. Overall, the proposed method effectively saves the running time of traditional methods.

VI. CONCLUSION

In this article, a semisupervised learning prediction model was established on the big data processing framework. First, the model was based on clustering method to segment fuzzy time series information. Then, the LSTM network was trained based on the fuzzy segmentation dataset. Finally, a prediction model capable of estimating complex information in time series is formed. This method is currently deployed on Apache Spark to complete the time series prediction of industry big data. In the experimental part, we use real marine environment datasets as experimental data to test the performance of the proposed model. We analyze the impact of different number of partitions on prediction in detail, and illustrate the performance comparison between the remaining prediction methods and the proposed method. It can be concluded from the final results that the proposed model can be applied to the time series prediction of some industry information.

In future work, we will try to apply this method to financial, medical, and other industries. And to improve some current problems of the model, such as how to use small sample data to predict long-term time series information. At present, big data technology is an important information channel for the development of each industry. However, most of the time series information is redundant and intermittent. Fuzzy time series theory will be an important channel to solve these problems. The research around fuzzy time series theory will be the focus of long-term work on data mining in the future.

REFERENCES

- [1] S. Zhou, Q. Xu, and Q. Qi, "Big data in mobile social networks: A QoE-oriented framework," *IEEE Netw.*, vol. 30, no. 1, pp. 52–57, Jan./Feb. 2016.
- [2] N. Zhong *et al.*, "Brain informatics-based big data and the wisdom web of things," *IEEE Intell. Syst.*, vol. 30, no. 5, pp. 2–7, Sep./Oct. 2015.
- [3] M. Birjali, A. Beni-Hssane, and M. Erritali, "Analyzing social media through big data using InfoSphere BigInsights and Apache Flume," *Proc. Comput. Sci.*, vol. 113, pp. 280–285, 2017.

- [4] M. Zaharia, R. S. Xin, P. Wendell, T. Das, and I. Stoica, "Apache Spark: A unified engine for big data processing," *Commun. ACM*, vol. 59, no. 11, pp. 56–65, 2016.
- [5] F. Xu *et al.*, "Big data driven mobile traffic understanding and forecasting: A time series approach," *IEEE Trans. Serv. Comput.*, vol. 9, no. 5, pp. 796–805, Sep./Oct. 2017.
- [6] B. Jiang, J. Yang, Z. Lv, and H. Song, "Wearable vision assistance system based on binocular sensors for visually impaired users," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1375–1383, Apr. 2019.
- [7] B. Jiang, J. Yang, H. Xu, H. Song, and G. Zheng, "Multimedia data throughput maximization in internet-of-things system based on optimization of cache-enabled UAV," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3525–3532, Apr. 2019.
- [8] W. Pedrycz, A. Jastrzebska, and W. Homenda, "Design of fuzzy cognitive maps for modeling time series," *IEEE Trans. Fuzzy Syst.*, vol. 24, no. 1, pp. 120–130, Feb. 2016.
- [9] Q. Han, D. Jie, E. Airoldi, and V. Tarokh, "SLANTS: Sequential adaptive nonlinear modeling of vector time series," *IEEE Trans. Signal Process.*, vol. 65, no. 19, pp. 4994–5005, 2017.
- [10] G. Sun, L. Song, H. Yu, V. Chang, X. Du, and M. Guizani, "V2V routing in a VANET based on the autoregressive integrated moving average model," *IEEE Trans. Veh. Technol.*, vol. 68, no. 1, pp. 908–922, Jan. 2019.
- [11] Y. S. Lee and L.-I. Tong, "Forecasting time series using a methodology based on autoregressive integrated moving average and genetic programming," *Knowl.-Based Syst.*, vol. 24, no. 1, pp. 66–72, 2011.
- [12] S. Arora and J. W. Taylor, "Short-term forecasting of anomalous load using rule-based triple seasonal methods," *IEEE Trans. Power Syst.*, vol. 28, no. 3, pp. 3235–3242, Aug. 2013.
- [13] C. O. Sakar, S. O. Polat, M. Katircioglu, and Y. Castro, "Real-time prediction of online shoppers' purchasing intention using multilayer perceptron and LSTM recurrent neural networks," *Neural Comput. Appl.*, vol. 31, no. 12, pp. 6893–6908, 2019.
- [14] Q. Song and B. S. Chissom, "Fuzzy time series and its models," *Fuzzy Sets Syst.*, vol. 54, pp. 269–277, 1993.
- [15] B. Q. Sun, H. Guo, H. R. Karimi, Y. Ge, and X. Shan, "Prediction of stock index futures prices based on fuzzy sets and multivariate fuzzy time series," *Neurocomputing*, vol. 151, no. 3, pp. 1528–1536, 2015.
- [16] H. J. Sadaei, R. Enayatifar, F. G. Guimarães, M. Mahmud, and Z. A. Alzamil, "Combining ARFIMA models and fuzzy time series for the forecast of long memory time series," *Neurocomputing*, vol. 175, no. part A, pp. 782–796, 2016.
- [17] J. C. Bezdek, R. Ehrlich, and W. Full, "FCM: The fuzzy C-means clustering algorithm," *Comput. Geosci.*, vol. 10, no. 2/3, pp. 191–203, 1984.
- [18] W. Zhou *et al.*, "MetaSpark: A spark-based distributed processing tool to recruit metagenomic reads to reference genomes," *Bioinformatics*, vol. 33, no. 7, pp. 1090–1092, 2017.
- [19] Y. Lv, Y. Duan, W. Kang, Z. Li, and F. Y. Wang, "Traffic flow prediction with big data: A deep learning approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 865–873, Apr. 2015.
- [20] A. Gonzalez-Vidal, F. Jimenez, and A. F. Gomez-Skarmeta, "A methodology for energy multivariate time series forecasting in smart buildings based on feature selection," *Energy Buildings*, vol. 196, pp. 71–82, 2019.
- [21] S. E. Haupt and B. Kosović, "Variable generation power forecasting as a big data problem," *IEEE Trans. Sustain. Energy*, vol. 8, no. 2, pp. 725–732, Apr. 2017.
- [22] A. Merigaud and J. V. Ringwood, "Incorporating ocean wave spectrum information in short-term free-surface elevation forecasting," *IEEE J. Ocean. Eng.*, vol. 44, no. 2, pp. 401–414, Apr. 2019.
- [23] S. Yang and J. Liu, "Time-series forecasting based on high-order fuzzy cognitive maps and wavelet transform," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 6, pp. 3391–3402, Dec. 2018.
- [24] Q. Cai, D. Zhang, Z. Wei, and S. C. H. Leung, "A new fuzzy time series forecasting model combined with ant colony optimization and auto-regression," *Knowl.-Based Syst.*, vol. 74, no. 1, pp. 61–68, 2015.
- [25] M. Y. Chen and B. T. Chen, "A hybrid fuzzy time series model based on granular computing for stock price forecasting," *Inf. Sci.*, vol. 294, no. 2, pp. 227–241, 2015.
- [26] C. M. Salgado, M. C. Ferreira, and S. M. Vieira, "Mixed fuzzy clustering for misaligned time series," *IEEE Trans. Fuzzy Syst.*, vol. 25, no. 6, pp. 1777–1794, Dec. 2017.
- [27] R. Efendia, N. Arbaay, and M. M. Deris, "A new procedure in stock market forecasting based on fuzzy random auto-regression time series model," *Inf. Sci.*, vol. 441, pp. 113–132, 2018.
- [28] C. H. Cheng and J. H. Yang, "Fuzzy time-series model based on rough set rule induction for forecasting stock price," *Neurocomputing*, vol. 302, pp. 33–45, 2018.
- [29] H. Song, C. Miao, R. Wuyts, Z. Shen, and F. Catthoor, "Implementation of fuzzy cognitive maps using fuzzy neural network and application in prediction of time series," *IEEE Trans. Fuzzy Syst.*, vol. 18, no. 2, pp. 233–250, Apr. 2010.
- [30] G. Selvachandran *et al.*, "A new design of Mamdani complex fuzzy inference system for multi-attribute decision making problems," *IEEE Trans. Fuzzy Syst.*, p. 1, 2019, doi: [10.1109/TFUZZ.2019.2961350](https://doi.org/10.1109/TFUZZ.2019.2961350).
- [31] R. W. R. De Souza, J. V. C. De Oliveira, L. A. Passos, W. Ding, J. P. Papa, and V. Albuquerque, "A novel approach for optimum-path forest classification using fuzzy logic," *IEEE Trans. Fuzzy Syst.*, doi: [10.1109/TFUZZ.2019.2949771](https://doi.org/10.1109/TFUZZ.2019.2949771).
- [32] N. Long Giang *et al.*, "Novel incremental algorithms for attribute reduction from dynamic decision tables using hybrid filter-wrapper with fuzzy partition distance," *IEEE Trans. Fuzzy Syst.*, vol. 28, no. 5, pp. 858–873, May 2020.
- [33] H. Wang, Z. Xu, and W. Pedrycz, "An overview on the roles of fuzzy set techniques in big data processing: Trends, challenges and opportunities," *Knowl.-Based Syst.*, vol. 118, pp. 15–30, Feb. 15, 2017.
- [34] A. Segatori, F. Marcelloni, and W. Pedrycz, "On distributed fuzzy decision trees for big data," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 1, pp. 174–192, Feb. 2018.
- [35] M. Elkano, J. A. Sanz, E. Barrenechea, H. Bustince, and M. Galar, "CFM-BD: A distributed rule induction algorithm for building compact fuzzy models in big data classification problems," *IEEE Trans. Fuzzy Syst.*, vol. 28, no. 1, pp. 163–177, Jan. 2020.
- [36] F. Pulgar-Rubio, A. J. Rivera-Rivas, M. D. Pérez-Godoy, P. González, C. J. Carmona, and M. J. del Jesus, "MEFASD-BD: Multi-objective evolutionary fuzzy algorithm for subgroup discovery in big data environments—A MapReduce solution," *Knowl.-Based Syst.*, vol. 117, pp. 70–78, Feb. 2017.
- [37] T. G. Barbounis, J. B. Theocharis, M. C. Alexiadis, and P. S. Dokopoulos, "Long-term wind speed and power forecasting using local recurrent neural network models," *IEEE Trans. Energy Convers.*, vol. 21, no. 1, pp. 273–284, Mar. 2006.
- [38] C. H. Lu and C. C. Tsai, "Adaptive predictive control with recurrent neural network for industrial processes: An application to temperature control of a variable-frequency oil-cooling machine," *IEEE Trans. Ind. Electron.*, vol. 55, no. 3, pp. 1366–1375, Mar. 2008.
- [39] S. Wen, Y. Wang, Y. Tang, Y. Xu, and T. Zhao, "Real-time identification of power fluctuations based on LSTM recurrent neural network: A case study on Singapore Power System," *IEEE Trans. Ind. Inform.*, vol. 15, no. 9, pp. 5266–5275, Sep. 2019.
- [40] J. Dean and S. Ghemawat, "MapReduce: Simplified data processing on large clusters," in *Proc. 6th Symp. Operating Syst. Des. Implementation*, 2004, pp. 137–150.
- [41] R. Lämmel, "Google's MapReduce programming model—Revisited," *Sci. Comput. Program.*, vol. 70, no. 1, pp. 1–30, 2008.
- [42] H. Ogawa, H. Nakada, R. Takano, and T. Kudoh, "SSS: An implementation of key-value store based MapReduce framework," in *Proc. IEEE 2nd Int. Conf. Cloud Comput. Technol. Sci.*, 2010, pp. 754–761.
- [43] M. A. Alsheikh, D. Niyato, S. Lin, H.-P. Tan, and Z. Han, "Mobile big data analytics using deep learning and Apache Spark," *IEEE Netw.*, vol. 30, no. 3, pp. 22–29, May/Jun. 2016.
- [44] R. B. Ray, M. Kumar, and S. K. Rath, *Fast Computing of Microarray Data Using Resilient Distributed Dataset of Apache Spark*. Berlin, Germany: Springer, 2016.
- [45] J. Chen, L. Kenli, R. Huigui, B. Kashif, L. Keqin, and P. S. Yu, "A periodicity-based parallel time series prediction algorithm in cloud computing environments," *Inf. Sci.*, vol. 496, pp. 506–537, 2019.
- [46] D. Chen *et al.*, "Real-time or near real-time persisting daily healthcare data into HDFS and ElasticSearch index inside a big data platform," *IEEE Trans. Ind. Inform.*, vol. 13, no. 2, pp. 595–606, Apr. 2017.
- [47] X. Meng *et al.*, "MLlib: Machine learning in Apache Spark," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 1235–1241, 2015.
- [48] X. Bai, Z. Chen, Z. Yu, Z. Liu, and L. Yi, "Spatial information based FCM for infrared ship target segmentation," *IEEE Trans. Cybern.*, vol. 46, no. 12, pp. 5127–5131, Jan. 2015.
- [49] K. Greff, R. K. Srivastava, J. Koutnk, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space odyssey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2222–2232, Oct. 2017.

- [50] B. Cao, J. Zhao, P. Yang, Z. Lv, X. Liu, and G. Min, "3-D multiobjective deployment of an industrial wireless sensor network for maritime applications utilizing a distributed parallel algorithm," *IEEE Trans. Ind. Inform.*, vol. 14, no. 12, pp. 5487–5495, Dec. 2018.
- [51] J. Yang, J. Wen, Y. Wang, B. Jiang, H. Wang, and H. Song, "Fog-based marine environmental information monitoring toward ocean of things," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 4238–4247, May 2020.
- [52] A. Arguez, T. R. Karl, M. F. Squires, and R. S. Vose, "Uncertainty in annual rankings from NOAA's global temperature time series," *Geophys. Res. Lett.*, vol. 40, no. 22, pp. 5965–5969, 2013.



Jiabao Wen received the M.S. degree in control engineering from Inner Mongolia University, Hohhot, China, in 2017. He is currently working toward the Ph.D. degree with the School of Electrical and Information Engineering, Tianjin University, Tianjin, China.

His current research interests include data processing, cloud computing, and Internet of Things.



Jiachen Yang (Member, IEEE) received the M.S. and Ph.D. degrees in communication and information engineering from Tianjin University, Tianjin, China, in 2005 and 2009, respectively.

He is currently a Professor with Tianjin University. He was a Visiting Scholar with the Department of Computer Science, School of Science, Loughborough University, Loughborough, U.K. His current research interests include ocean information processing, Internet of Things, cloud computing, big data analytics, stereo vision research, pattern recognition, and image

quality evaluation.



Bin Jiang received the B.S. and M.S. degrees in communication and information engineering, in 2013 and 2016, respectively, from Tianjin University, Tianjin, China, where he is currently working toward the Ph.D. degree.

He is also a Visiting Scholar with the Department of Electrical, Computer, Software, and Systems Engineering, Embry-Riddle Aeronautical University, Daytona Beach, FL, USA, where he is a member of Security and Optimization for Networked Globe Laboratory. His current research interests include

multimedia quality and information security.



Houbing Song (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the University of Virginia, Charlottesville, VA, USA, in August 2012.

In August 2017, he joined the Department of Electrical, Computer, Software, and Systems Engineering, Embry-Riddle Aeronautical University, Daytona Beach, FL, USA, where he is currently an Assistant Professor and the Director of the Security and Optimization for Networked Globe Laboratory (www.SONGLab.us). He was a faculty member of West

Virginia University from August 2012 to August 2017. He is the Editor of four books and the author of more than 100 articles. His current research interests include cyber-physical systems, Internet of Things, cloud computing, big data analytics, connected vehicle, wireless communications and networking, and optical communications and networking.

Dr. Song has been an Associate Technical Editor for *IEEE Communications Magazine* since 2017. He is a senior member of the ACM.



Huihui Wang (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the University of Virginia, Charlottesville, VA, USA, in 2013.

In 2011, she was an Engineering Intern with Qualcomm, Inc. In 2013, she joined the Department of Engineering, Jacksonville University, Jacksonville, FL, USA, where she is currently an Assistant Professor and the Founding Chair of the Department of Engineering. She has authored more than 30 articles and holds one U.S. patent. Her research interests include

cyber-physical systems, Internet of Things, healthcare and medical engineering based on smart materials, robotics, haptics based on smart materials/structures, ionic polymer metallic composites, and MEMS.

TITLE/ AUTHOR	Internet of Things in Marine Environment Monitoring: A Review. Sensors / Xu, G., Shi, Y., Sun, X., & Shen, W.
SOURCE	Sensors Volume 19, No. 7, Pages 1711 (April 2019) http://dx.doi.org/10.3390/s19071711 (Database: MDPI AG)

Review

Internet of Things in Marine Environment Monitoring: A Review

Guobao Xu ¹, Yanjun Shi ², Xueyan Sun ³ and Weiming Shen ^{3,*}

¹ School of Electronics and Information Engineering, Guangdong Ocean University, Zhanjiang 524088, China; xuguobao@126.com

² School of Mechanical Engineering, Dalian University of Technology, Dalian 116024, China; syj@dlut.edu.cn

³ State Key Lab of Digital Manufacturing Equipment and Technology, Huazhong University of Science & Technology, Wuhan 430074, China; sunxueyanxy@163.com

* Correspondence: wshen@ieee.org; Tel.: +86-27-8754-3129

Received: 5 March 2019; Accepted: 6 April 2019; Published: 10 April 2019



Abstract: Marine environment monitoring has attracted more and more attention due to the growing concern about climate change. During the past couple of decades, advanced information and communication technologies have been applied to the development of various marine environment monitoring systems. Among others, the Internet of Things (IoT) has been playing an important role in this area. This paper presents a review of the application of the Internet of Things in the field of marine environment monitoring. New technologies including advanced Big Data analytics and their applications in this area are briefly reviewed. It also discusses key research challenges and opportunities in this area, including the potential application of IoT and Big Data in marine environment protection.

Keywords: Internet of Things; Big Data; wireless sensor networks; marine environment monitoring

1. Introduction

Although the Internet of Things (IoT) has been defined in various perspectives, there is a common version that is widely accepted by scholars as follows: IoT is a dynamic global network infrastructure with self-configuring capabilities based on standard and interoperable communication protocols, where physical and virtual “things” have identities, physical attributes, and virtual personalities and use intelligent interfaces, and are seamlessly integrated into the information network [1]. Recently, IoT has been widely accepted as a promising paradigm that can transform our society and industry. It can achieve the seamless integration of various devices equipped with sensing, identification, processing, communication, actuation, and networking capabilities [2]. A wireless sensor network (WSN) plays a key role in IoT. It consists of a large number of distributed sensors interconnected through wireless links for physical and environmental monitoring purposes. On the other hand, Big Data is considered as an emerging technology and has become a very active research area, primarily involving topics related to data mining, machine learning, database, and distributed computing.

During the past couple of decades, wireless sensor networks (WSNs), as a subset of IoT, have been widely utilized in a variety of smart applications and services, including smart home [3], smart building [4,5], smart transportation [6,7], smart industrial automation [8,9], smart healthcare [10], smart grids [11], and smart cities [12]. Similar IoT-based technologies can certainly be applied to the monitoring and protection of marine environments.

With the development of our society and economy, the marine environment has drawn increasing attention from scientists and scholars. Conventional marine environment monitoring systems such as oceanographic and hydrographic research vessels are very expensive. Their data collection and

analysis processes are time-consuming and the collected data have a low resolution. The Internet of Things (IoT) has been evolved from wireless sensor networks (WSNs). Compared with WSNs, IoT has much stronger data processing capabilities, enabling intelligent control of objects.

In a typical IoT-based marine environment monitoring system, different sensors are deployed to measure and monitor various physical and chemical parameters like water temperature and pressure, wind direction and speed, salinity, turbidity, pH, oxygen density, and chlorophyll levels. An advanced IoT-based marine environment monitoring and protection system would also be able to control some objects, devices, or equipment within the monitored marine environment, in order to adjust some physical and chemical parameters so as to improve the marine environment.

While the design, development, and deployment of an IoT-based marine environment monitoring and protection system is needed to address some critical issues including autonomy, adaptability, scalability, simplicity, and self-healing [13,14], following requirements specific to the harsh marine environments should be considered [15]:

- (1) High water resistance: Sensor and actuator nodes need to have very high levels of water resistance;
- (2) Strong robustness in hardware: Hardware or equipment needs stronger robustness due to the aggressive and complex marine environment with currents, waves, tides, typhoons, etc.;
- (3) Low energy consumption and energy harvesting: Energy conservation and harvesting measures need to be considered due to long communication distances and an environment in constant motion;
- (4) Stability of Radio signal: Special techniques may be required to ensure the stability of radio signals since the oscillation of the radio antenna can cause an unstable line-of-sight between transmitters and receivers [16] and bad weather conditions can also affect the stability of radio signals;
- (5) Other issues: Devices and sensor nodes should be highly reliable because of the difficult deployment and maintenance; the need for buoy and mooring devices; sensor coverage needs to be carefully calculated because of large areas [17]; equipment should be designed against possible acts of vandalism.

We conducted a comprehensive review of WSN applications in marine environment monitoring in 2014 [18]. When we planned and started this updated review, we wanted to expand the scope from WSN to IoT and to cover the protection aspect in addition to the monitoring because of the characteristics of IoT not only on sensing but also on actuation. However, we did not find enough references related to protection. In this paper, we do not consider the protection aspect as a focus but keep the related discussions. We envision the protection from two perspectives: (1) Results from the advanced data analytics based on the data collected by the monitoring system can be fed back to the marine environment management agencies/control centers for quick decision making and real time manual interventions in order to protect the marine environment from some disasters (e.g., oil spills and bad weather damages); (2) some autonomous vessels or other devices (actuators) can also automatically and quickly react to disasters and other events in order to protect the marine environment. Similarly, we wanted to cover the Big Data analytics and its applications to marine environment monitoring and protection, but we did not find sufficient references. Therefore, we did not make the Big Data a focus of this paper. However, we strongly believe that Big Data analytics will play a more and more important role in this area in the future.

The remainder of the paper is organized as follows: Section 2 provides an overview of the fundamentals of IoT-based marine environment monitoring systems; Section 3 presents a summary of some related projects and systems under five application areas; Section 4 reviews a few new technologies applied in the marine environment monitoring systems; Section 5 addresses some research challenges and opportunities in this area; Section 6 provides some concluding remarks.

2. Overview of IoT in Marine Environment Monitoring

This section provides an overview of IoT in marine environment monitoring, including various applications, common system architectures, typical sensing nodes and sensing parameters, and related wireless communication technologies.

2.1. IoT-Based Marine Environment Monitoring Applications

IoT-based marine environment monitoring application areas include: (1) Ocean sensing and monitoring; (2) water quality monitoring; (3) coral reef monitoring; (4) marine (either offshore or deep-sea) fish farm monitoring; (5) wave and current monitoring. Different applications use different IoT system architectures, sensing and control technologies, and communication technologies.

An ocean sensing and monitoring system is a general marine environment monitoring system, which existed for a long time, previously using oceanographic and hydrographic research vessels. A water quality monitoring system usually monitors water conditions and qualities, including water temperature, pH, turbidity, conductivity, and dissolved oxygen (DO) for ocean bays, lakes, rivers, and other water bodies. A coral reef monitoring system typically monitors coral reef habitats and the surrounding environments. A marine fish farm monitoring system monitors water conditions and qualities including temperature and pH, measures the amount of fecal waste and uneaten feed for a fish farm, as well as fish conditions and activities including the number of dead fishes. A wave and current monitoring system measures waves and currents for safe and secure waterway navigations.

2.2. Common IoT-Based System Architectures for Marine Environment Monitoring and Protection

The Internet of Things is usually to achieve “knowing, thinkable, and controllable” to the surrounding world [19], which means that the IoT is able to perceive, think, and control the world by collecting, processing, and analyzing the data of the world. It can make intelligent judgments that impact on the outside world. IoT researchers have proposed different IoT system architectures in the research literature. Among them, a five layered system architecture was proposed by Antao et al. [20]. Similarly, we also believe that a typical IoT-based marine environment monitoring and protection system has five layers: Perception and execution layer, transmission layer, data pre-processing layer, application layer, and business layer, as shown in Figure 1.

(1) Perception and Execution Layer

The perception and execution layer is the bottom layer of the architecture. It includes sensor and actuator devices, with the objective of sensor data collection and command actuation. In IoT-based marine environment monitoring and protection systems, this layer can also include GPS sensors, energy harvesting devices, in addition to regular water condition and quality monitoring sensors. Note that most existing marine environment monitoring systems do not have any execution functions and therefore do not include actuators.

(2) Data Transmission Layer

The main function of the data transmission layer is to transmit various collected data to the data processing layer via communication networks, mostly mobile or wireless communication networks. At the same time, control measures made by users or intelligent applications (reference engines) are transferred from the application layer to the perception and execution layer, thus that corresponding devices or actuators can take required actions (such as device repositioning, increasing or decreasing temperature settings, releasing food in fish farms).

(3) Data Pre-Processing Layer

The data pre-processing layer is in the middle of the IoT system architecture, where the raw data received can be stored and pre-processed, using advanced data mining technologies. It also completes

information aggregation or disaggregation, data cleaning and fitting or screening, sharing as needed, and sometimes it triggers alerts or warnings based on pre-defined rules.

(4) Application Layer

The application layer provides services according to different applications requested by users. For example, it provides water condition and quality data as well as the amount of fecal waste and uneaten feed for a fish farm. The main purpose of this layer is to provide smart application services to meet users' needs. In IoT-based marine environments, this layer covers water quality monitoring, coral reef monitoring, marine (either offshore or deep-sea) fish farm monitoring, wave and current monitoring [21–23].

(5) Business Layer

The business layer is the top layer and manages the overall IoT system activities and services, including creating business models, business logic flowcharts, graphic representations, according to the data received from the application layer. It also monitors and verifies outputs of the other four layers according to the business models in order to enhance services and maintain users' privacy [21,23].

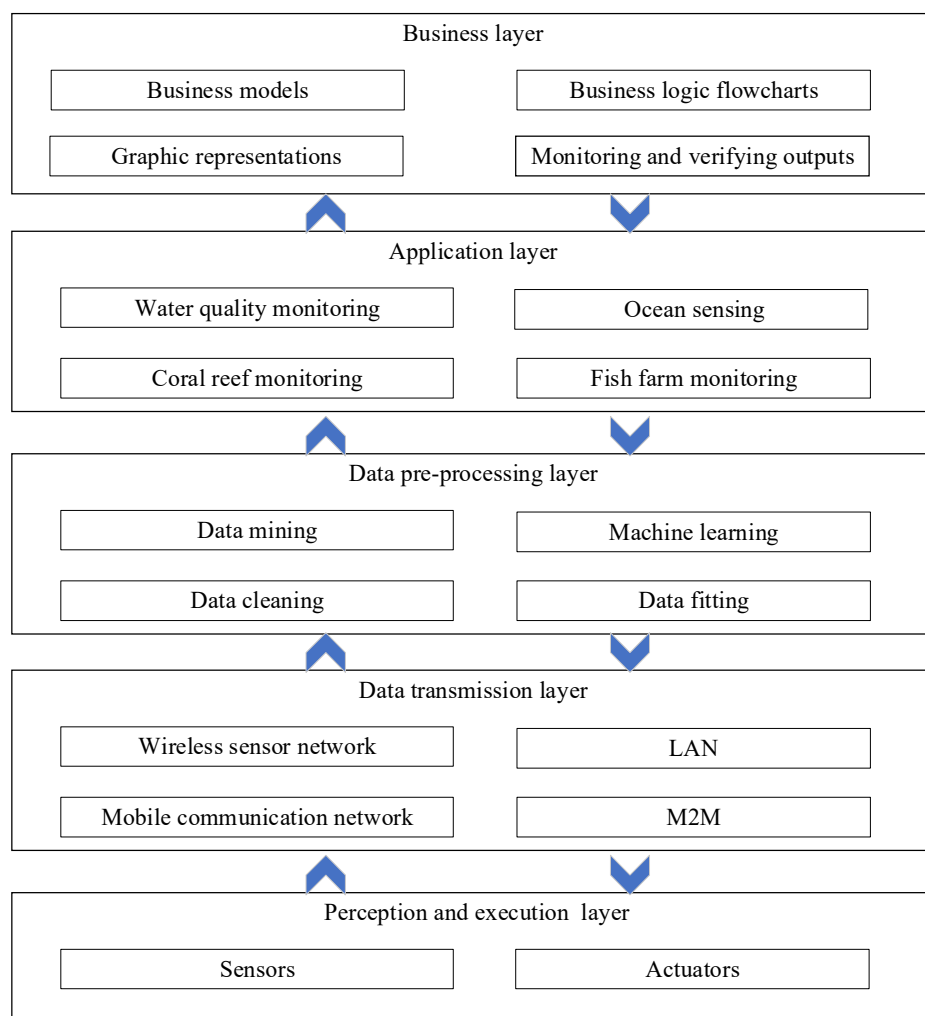


Figure 1. Common layered architecture for Internet of Things (IoT)-based marine environment monitoring and protection applications.

Such a layered system architecture provides a good picture of the data/information flow in IoT-based marine environment monitoring and protection systems. Figure 2 shows a common physical

architecture of IoT-based marine environment monitoring and protection systems. It includes sensor nodes and actuator nodes, sink nodes, a base station, a system server, and user terminals. Sensor nodes are used to sense and monitor environmental parameters such as water temperature and pH, salinity, turbidity, oxygen density, and chlorophyll levels, and transmit the collected data to sink nodes via ZigBee or some other wireless communication protocols. Actuator nodes execute commands from the upper layers. Communication between a sink node and sensor or actuator nodes is usually point-to-point. A sink node collects data from a group of sensor nodes and sends collected data to the base station or passes execution commands from upper layers to actuator nodes, via mobile communication networks (2G/3G/4G). The base station is connected to a system server through the Internet. The system server stores and processes the received data from the base station, completes data analyses according to corresponding applications, sends commands to actuator nodes according to the pre-defined rules as appropriate. Various kinds of user terminals (desktops, laptops, pads, and smart phones, etc.) connect users to the system server over the Internet (including mobile Internet).

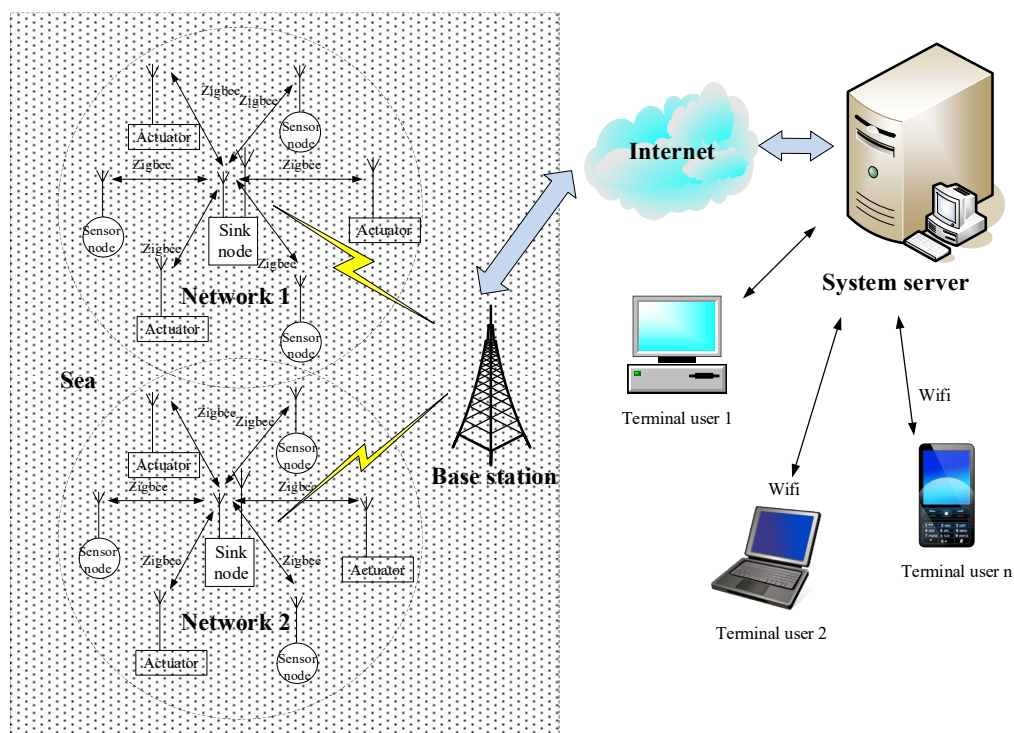


Figure 2. Common physical architecture of IoT-based marine environment monitoring and protection systems.

The design and development of a durable and scalable IoT system for marine environment monitoring and protection should carefully consider a number of factors: The harsh marine environment, the communication protocols and network topology, the number and distribution of nodes, buoys and mooring systems, oceanographic sensors, energy supply and harvesting options, and so on.

As described above, an IoT system consists of many sensor and actuator nodes and a gateway for the connection to the Internet. Usually, sensor and actuator nodes are organized into a connected network according to a certain topology. Physical topology and density are entirely dependent on applications [24], therefore the design and development of an IoT system need to consider its application and deployment environment. Even though more sensors can be densely deployed to enhance data availability and accuracy, a dense deployment also brings negative issues: High energy consumption, data collisions, and interferences, etc. [25]. Sensor nodes normally have four typical kinds of network topologies: Linear, star, cluster/tree, and mesh topologies, as shown in Figure 3.

While we have discussed in detail the star, cluster/tree, and mesh topologies in reference [18], the linear topology is a series connection of all the nodes. Its main advantage is that it has a simple structure thus that the cost is relatively low. As a consequence, the transmission volume is large, and the speed is slow.

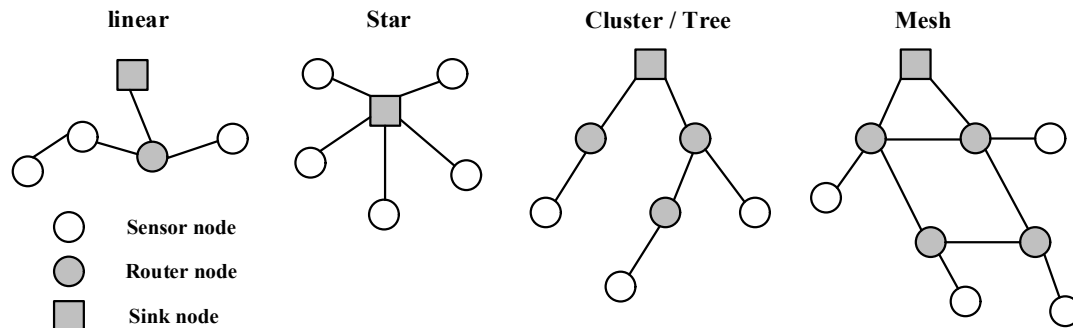


Figure 3. Typical wireless sensor network topologies.

The selection of a right network topology for a particular application depends on the amount and frequency of data to be transmitted, the distance of data transmission, the requirement of battery life as required for maintenance, and the mobility of the sensor nodes [26]. On the other hand, the physical network topology of an IoT system may change during its operation due to its energy availability, node position variations, equipment or sensor malfunction, node reachability (due to noise, severe weathers, moving obstacles, etc.), and task details of sensor nodes [27].

2.3. A General Marine Environment Monitoring Sensor Node

The general architecture of a marine environment monitoring sensor node is shown in Figure 4. It typically has a buoy device to protect electronic devices against water, and consists of the following four main modules [28]: A sensing module, a microcontroller, a wireless transceiver module, and a power supply module.

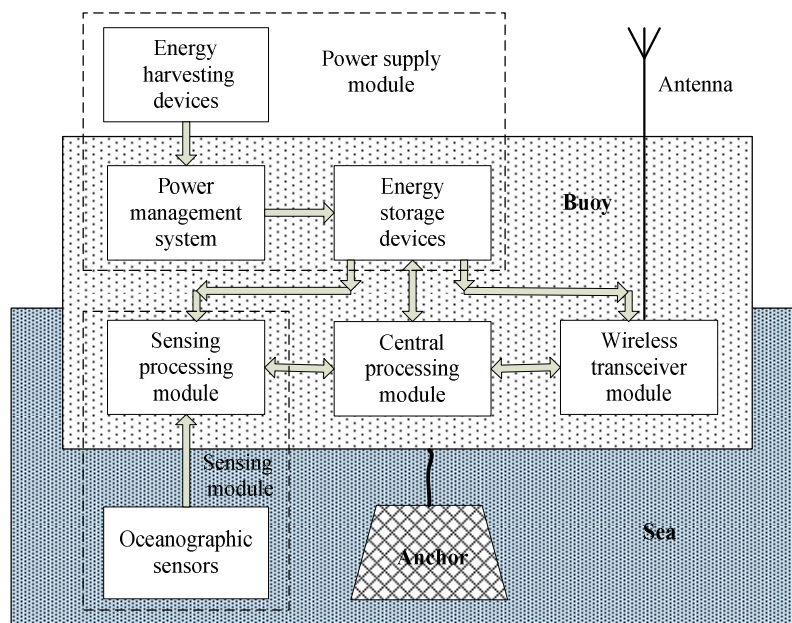


Figure 4. General architecture of a marine environment monitoring sensor node.

The sensing module obtains the environmental and equipment status (with associated amplifiers and analog-to-digital (A/D) converters). The microcontroller receives the data from the sensor and processes the data accordingly. A wireless transceiver module includes a radio frequency (RF) transceiver and an antenna. A power supply module includes energy storage devices (like rechargeable batteries), and a power management system with energy harvesting devices, which can be a solar panel, a wind energy harvesting device, a tidal power generator, or a seawater power generator. The buoy has an anchor to prevent it from moving.

2.4. Typical Sensors and Sensing Parameters

Sensors are used to respond to changes in their environments by producing electrical signals in the form of electrical voltage, current, or frequency [29]. There are typically two kinds of sensors: Physical sensors and chemical sensors. Physical sensors are used to measure different physical parameters like temperature, humidity, pressure, wind speed, and wind direction. Chemical sensors are used to measure various chemical parameters like salinity, turbidity, pH, nitrate, chlorophyll, and dissolved oxygen (DO). Details can be found in reference [18]. Decisions on the selection of sensors are made according to requirements related to the deployment area and season, measurement range, accuracy, resolution, and power consumption.

2.5. Wireless Communication Technologies

A sensor node requires a radio module for wireless communication. The access network, with a communication range from a few hundred meters to several kilometers, includes all the devices between the backbone network and user terminals.

For the IoT-based marine environment monitoring and protection systems, wireless communication networks have different requirements than other applications, because of the following reasons:

- (1) Reliability: Radio antenna oscillations and bad ocean weather conditions can cause instability of radio signals.
- (2) Energy efficiency: Low power consumption is the key to supporting long-flow and reduced maintenance costs in stand-alone battery-powered equipment. This is particularly critical for devices deployed in remote offshore areas that are difficult and costly to replace.

According to the current requirements for IoT applications, the development of wireless communication technologies has already made significant progresses. Various wireless communication standards and technologies have been proposed and developed, including WiFi, ZigBee, Bluetooth, GPRS, GSM, and WiMAX. A summary and brief comparison of these communication standards and technologies can be found in our previous survey paper [18]. Typically, multiple wireless communication technologies are used in an IoT-based marine environment monitoring and protection system. In some specific applications, underwater acoustic communication technologies are used for data collection and communication among underwater marine environment sensors [30–33]. Generally, a longer-range communication consumes more energy. Selection of the most appropriate wireless communication technology for an application depends on the transmitted data volume, transmission frequency, transmission distance, and available power supply.

3. A Review of Existing Marine Environment Monitoring Projects and Systems

Various systems have been developed and deployed for marine environment monitoring during the past couple of decades. Table 1 summarizes the features of reviewed projects and systems under five different application areas as defined in Section 2.1. For completeness and consistency, we kept all the projects and systems summarized in our previous survey [18] and added more from the recent research literature.

Table 1. Summary of existing marine environment monitoring projects and systems.

Reference	Country	Sensing Parameters	Comm. Protocols	Buoy	Energy Harvesting	Key Features (Including Testing and Deployment)
Application Area: Ocean Sensing and Monitoring						
Perez et al. [34]	Spain	T, P, salinity, nitrates, velocity, chlorophyll, and turbidity	GPRS, ZigBee	Special buoy	Solar	LabVIEW-based user interface using Google Maps; Solar energy harvesting; Special buoy; Deployed in a harbor
Voigt et al. [35]	Sweden; Germany	T, motion, vibration and sound	GPRS	Simple buoy and king's buoy	No	Design of an advanced low-cost buoy system; tested in real environment
Vesecky et al. [36]	USA	T, wave and location	900 MHz	Mobile minibuoy	No	An autonomous mini-buoy prototype (tested in a pool); GPS is used
Liu et al. [37]	China	T, Sea depth	ZigBee	Sensor floating	No	A Perpendicular Intersection (PI) mobile-assisted localization scheme; deployed in Hong Kong U of S&T campus and Tsingtao
Macias et al. [38]	Spain	T, Visible-field, sound	ZigBee and acoustic	?	?	Three-tier communication architecture; transmitting video streaming data; Tested on module of NS-3
Roadknight et al. [39]	UK	T, conductivity, water depth, turbidity	?	Single buoy	No	A multi-layered scalable and adaptive approach of data management; deployed off Scroby sands
Cella et al. [40]	Australia	T, illuminance	ZigBee	Cylinder waterproof buoys	Solar	Used underwater wireless communication; deployed in the Moreton Bay
Jiang et al. [41]	China	T, velocity and light	ZigBee	Lever buoy	No	The sleep mechanism and lever buoy; deployed off the seashore
Tao et al. [42]	China	Water T, DO and pH	ZigBee	Buoys with GPS and PEA	?	Position determination and location verification using GPS and PEA (positioning estimation algorithms); tested in two testbeds
Alippi et al. [43]	Italy	Seawater luminosity, T and moisture	ZigBee	Cylinder waterproof buoys	Solar	Optimal solar energy harvesting; power-aware and adaptive TDMA protocol; deployed in the Moreton Bay
De Marziani et al. [44]	Argentina	T, P, PAR radiation, pH and salinity	ZigBee	Cylinder waterproof buoys	Solar	A low cost reconfigurable WSN with solar panels; tested in San Jorge Gulf
Albaladejo et al. [45]	Spain	T, P	ZigBee	Special buoy	Solar	A new multisensory buoy system with solar panels; deployed in Mar Menor Lagoon
Al-Zaidi et al. [46]	UK	T, depth, wind speed and direction, humidity, salinity	MADNET routing protocol	Ship	?	Marine data acquisition and cartography system based on VHF; hybrid Mobile Ad-hoc/Delay Tolerant routing protocol (MADNET); tested in North Sea, and English Channel
Ferreira et al. [47]	Portugal	T, position	WiFi, GPRS/UMTS/LTE, Acoustic	Ship Buoy, ASV	No	Used autonomous underwater vehicles (AUV), and autonomous surface vehicles (ASV); tested in Portuguese coast
Kaur et al. [48]	India	Water T, P, wind speed, wind direction, humidity, cloud cover, turbidity	GPRS		?	SentiWordNet is used as an information retrieval tool for processing messages received from nearby marine areas

Table 1. Cont.

Reference	Country	Sensing Parameters	Comm. Protocols	Buoy	Energy Harvesting	Key Features (Including Testing and Deployment)
Hu et al. [49]	China	T, humidity and salinity	?	AUV	?	Ring Broadcast Mechanism is used to guide searching direction of sensor nodes; providing self-adaptive dynamic routing mechanism to search the alternative path
Mourya et al. [50]	UK	T, P, salinity, oxygen level	Acoustic	Anchors with acoustic modems	Solar	A framework for spatio-temporal monitoring of underwater acoustic sensor networks; anchors are deployed in the ROI inspired by compressive sensing
Morozs et al. [51]	UK	T, P, humidity, optical, distance, sound, magnetic field, motion	Acoustic	Autonomous surface vehicle (ASV)	No	Implementation of the TDA-MAC protocol in practice, and practical issues prompted several crucial modifications to the TDA-MAC protocol
Song et al. [52]	China	Water T, P, salinity and PH	Acoustic	Buoy	?	Underwater positioning algorithm of electing anchor nodes and the self-repairing localization algorithm based on anchor nodes failure
Application Area: Water Quality Monitoring						
Yang et al. [53]	USA	pH	RF and acoustic	PVC housing	No	Various interface circuits; 5 air-based sensor nodes; lab testing only
Seders et al. [54]	USA	T, pH, and DO	433 MHz	Box and polyethylene ring	No	A LakeNet sensor pod and an altered sampling strategy; tested a prototype in a small lake
Regan et al. [55]	Ireland	T, pH, turbidity, DO and conductivity	ZigBee	Inshore sensor buoys	Solar	A real-time heterogeneous water quality monitoring; deployed in five sites on the River Lee, Ireland
O'Connor et al. [56]	Ireland	T, conductivity and depth	?	Buoys	?	A multi-modal environment monitoring network based on WSN and visual image; tested in River Lee, Poolbeg Marina and Galway Bay
Hadjimitsis et al. [57]	Cyprus	T, P, salinity and turbidity	GPRS	Cylinder waterproof buoy	No	Integrated satellite remote sensing and WSN; deployed in a beach
Jin et al. [58]	China	T, pH, DO, and salinity	ZigBee GPRS	?	No	An early WSN-based water monitoring system
Alkandari et al. [59]	Kuwait	Water T, DO, and pH	ZigBee 802.11 Ethernet radio	?	Solar	Used ZigBee and 802.11 and a high capacity solar panel; tested in a pool
Adamo et al. [60]	Italy	T, salinity, conductivity, turbidity and chlorophyll-a	GPRS	Self-sufficient buoy	?	Two different probe solutions for field covering; tested in Apulia region
Application Area: Coral Reefs Monitoring						
Bromage et al. [61]	USA	T, P, pH, light, and conductivity	900 MHz	Watertight housing	No	Deployed in Monterey Bay
Berlian et al. [62]	Indonesia	T, ORP, pH, Electrical Conductivity, DO, audio/video	?	Remotely Operated Vehicle, buoy	No	Remotely Operated Vehicles with water quality sensors; Big Data analysis

Table 1. Cont.

Reference	Country	Sensing Parameters	Comm. Protocols	Buoy	Energy Harvesting	Key Features (Including Testing and Deployment)
Application Area: Fish Farm Monitoring						
López et al. [63]	Spain	T and pH	ZigBee	?	No	A sub-layer-based power consumption algorithm; tested in a pool
Yang et al. [64]	China	Water T, pH value, salinity, DO and COD	GPRS	?	Solar	Multi-hop communication protocol, multiple nodes, and SMT; tested in an aquatic experimental base
Leblond et al. [65]	France	T, depth, salinity, position, catches	GPRS	Vessels	No	Fixed on fishing gears, self-powered, autonomous; tested in Bay of Biscay
Lloret et al. [66]	Spain	Sediment depositions	Acoustic	Bouy	?	Ultrasonic sensor; tested through simulations
Meera et al. [67]	India	Sea surface T, quality of sea water, pH, chlorophyll	WiFi	Fishing vessels	No	A multi-level P2MP infrastructure network—OceanNet; protocol performance comparison of CoAP, AMQP and MQTT
Lloret et al. [68]	Spain	Amount of pollution	?	Buoy	?	A group-based underwater WSN for monitoring fecal waste and uneaten feed; tested on OPNET Modeler network simulator
Application Area: Wave and current monitoring						
Marimon et al. [69]	Philippines	Acceleration, angle	ZigBee, GSM/GPRS/EDGE	Buoy	Solar	Integrated different wave sensors; threshold values generated based on statistics; tested in Manila Bay
Chen et al. [70]	China	Current velocities	?	?	No	A temporal evolution model to describe the ocean current process based on the temporal correlation of the current velocity.

Notes: “T”: Temperature; “P”: Pressure; “DO”: Dissolved Oxygen; “COD”: Chemical Oxygen Demand; “No” under Energy Harvesting: Battery power is used; “?”: Related information is not available from the source.

From this long list of systems, we can see that most of the efforts are related to general ocean sensing and monitoring [34–44,46–48] and water quality monitoring [53–60,62]. Some specific efforts have been made for fish farm monitoring [63,65,66,68], coral reef monitoring [61,62], wave and current monitoring [69], and marine shellfish monitoring [64]. Several projects focus on specific technologies or devices, e.g., buoys [35,36,44,45,47,64], energy saving and harvesting [34,40,43–45,55,59,63,64], routing protocols [46,64], data transmitting approaches [38,46,47], and data analysis [62].

It can also be found that testing places are different among these systems. About half of the developed systems have been tested or deployed in real marine or river environments [34,35,37,39–41,44–47,55–57,60,61,65,69]; a number of them were experimented in lab settings or indoor environments [38,53,64,68]; some were tested in outdoor pools or small ponds/lakes [36,54,59,63]; and several of them were only tested through simulations [38,66,68].

It is very interesting to note that most projects, systems, and applications have been developed by research groups in a small number of countries, including China [37,41,42,58,64], USA [36,53,54,61], Spain [38,45,63,68], UK [46], Indonesia [62], Philippines [69], India [48], Portugal [47], France [65], and Ireland [55,56]. In these completed systems, most of them occurred on water surfaces [34,35,37,39–46,48,54–65,68,69] and only a few of them were deployed under water [38,47,53,66]. Please note that underwater wireless sensor network is not the focus of this paper and we will briefly discuss this later in the paper.

Even though many researchers discussed different options for energy harvesting including solar, wind, waves, and ocean currents, only solar energy has been used in about a third of these systems and applications.

While most systems use GPRS and/or ZigBee for wireless communications, a few systems use underwater acoustic communication [38,47,53,66].

Offshore and deep-sea fish farming has been emerging, which provides more opportunities and also challenges, not only for marine environment monitoring but also for operation controls and marine environment protection.

4. New Technologies for Marine Environment Monitoring and Protection

4.1. Data Analysis

Recent fast development and deployment of IoT technologies in marine environment monitoring created huge amounts of data, while the recent advancement of Big Data analytics facilitated the analysis of these marine environment data.

As in many other IoT-based data collection systems, dealing with marine environment data also faces some major challenges, particularly the large amount of data and significant bad data. Researchers around the world have been trying to address these challenges. Yang et al. [71] proposed a method to quickly describe the contour of data collected over the Internet of Things (IoT). The distribution of contour lines can be calculated accurately in a short time.

Blix and Eltoft [72] proposed an automatic model selection algorithm (AMSA) to determine the best model for a given matchup dataset. It can automatically choose between regression models to estimate the parameter of interest. It also finds out the number and combination of features to be used for obtaining the best model. They used four Machine Learning feature ranking methods and three Machine Learning regression models to estimate oceanic chlorophyll-a in the global and optically complex waters.

Zhong et al. [73] developed a fast fuzzy C-means clustering algorithm to analyze water environment monitoring data of the Three Gorges Reservoir Area. The hard cluster center can be treated as the initial value of the fuzzy cluster center to accelerate the speed of convergence and reduce the number of iterations.

Addison et al. [74] summarized the challenges in marine environment data management and interpretation caused by the implications of Big Data. They suggested a solution for the management

of Big Data, which requires new collaborations between marine practitioners and data scientists with expertise in programming languages and packages like R and Python.

Belghith et al. [75] proposed a deep learning-based approach in a marine Big Data setting that enables to classify these diverse acoustic sounds not only considering marine mammals signals.

Li et al. [76] presented a support vector regression architecture with smoothness priority for marine sensor data prediction to handle the abruptly fluctuating, multi-noise, non-stationary and abnormal data. The smoother plays the role of preprocessing to handle the outliers and noises in marine sensor data, providing stable initialization values for the next nonlinear approximation based on support vector machines.

Since the existing current correlation analysis method for ocean monitoring big data is time consuming and stability is poor, Song et al. [77] developed a new method by sending the collected data to the cloud storage system. Based on the global and local Moran index calculations, the ocean big data of relatively high correlation were saved to the adjacent data center, which reduces the ocean monitoring data correlation analysis time.

Radeta et al. [78] developed a low-cost passive acoustic monitoring (PAM) system for nautical citizen science and real-time acoustic augmentation of whale-watching experiences. This paper used machine learning identify vocal acoustic samples of common cetaceans like whales and dolphins with acoustic features (clicks, moans or whistles).

It is clear that research and development efforts on the application of Big Data analytics in marine environment monitoring have been growing recently.

4.2. Network Topology Control

In wireless sensor networks (WSNs), network topology control capability is a key factor in the performance of the entire network. A reasonable WSN topology control structure can effectively improve the efficiency of network communication protocols and the overall performance of the network. A good network topology helps extend the overall life cycle of the network. Therefore, WSN topology control optimization technology is the key to determining the overall performance of the network, including the network coverage and connectivity. The node selection policy changes the state of the node itself and avoids the communication link redundancy among nodes to form a performance-optimized network structure. Table 2 summarizes the advantages and disadvantages of different topology control algorithms for underwater wireless sensor networks (UWSNs) [79].

Table 2. Classification of Topology Control Algorithms for underwater wireless sensor networks (UWSNs).

Category	Main Idea	Advantages	Disadvantages
Power control based	The proper transmission power level is assigned to each node to guarantee enough signal strength at the receiver that it can successfully receive and decode the transmitted message	Simple; scalable; conserves energy; does not change the sensing coverage; can overcome time-varying acoustic channel quality.	May diminish the network connectivity; increases the number of hops and end-to-end delay.
Wireless interface mode management based	The wireless interface of nodes alternates between active, sleeping, and powered-off modes. This change reduces the amount of unnecessary time a node spends listening to the channel.	Simple; scalable; conserves energy relative channel polling; does not change sensing coverage.	Changes network density; changes routing paths from time to time; increases delay.
Mobility assisted based	Some mobile nodes are moved to new locations in different depths or with a predetermined trajectory, creating new interconnections.	Improves network connectivity; deals with network partitions; improves data collection from hop spots.	Needs trajectory planning procedures; increases energy cost for mobility; may change sensing coverage.

Even though this classification of topology control algorithms is for underwater wireless sensor networks, it can be well applied to the IoT-based marine environment monitoring in general. Please note that UWSN is not the focus of this paper. For comprehensive reviews of underwater sensor networks and applications, please refer to references [80,81]. It is also interesting to check out the concept of Internet of Underwater Things (IoUT) and its potential applications [82].

The rest of this section provides a review of some examples of interesting approaches proposed and developed in the literature on network topology control. Kim et al. [83] used power control to achieve reliable data delivery based on the sea surface movement that affects the surface signal reflection and the strength of the received signal at a node. Bai et al. [84] proposed an approach to reduce link interference and achieved high throughput by using a correlation matrix to describe the source-destination relationship and conflict relationships among the links. Power control is used to achieve a Signal to Noise Ratio (SNR) larger than the decoding threshold.

Su et al. [85] proposed a cycle difference set-based protocol to determine the number and positions of active and sleep intervals in one cycle to guarantee that both the transmitter and receiver are awake for communication. Coutinho et al. [86] developed an optimization model to investigate the performance of the on-the-fly adjustment of the sleep interval in duty-cycled UWSNs to achieve a balanced energy consumption.

Khan et al. [87] presented a method to incorporate the AUV resurfacing time in the VoI function and AUV path planning algorithm. Coutinho et al. [88] designed the distributed topology control (DTC) and centralized topology control (CTC) depth adjustment-based topology control algorithms for disconnected and void nodes repositioning to improve network connectivity and data routing.

4.3. New Communication Routing Protocols

With the advancement of communication technologies, new communication routing protocols have been proposed and developed for marine environment monitoring systems. Faheem et al. [89] proposed a chromosome (routing path), which consists of sequences of non-negativity integers that denote the IDs of genes (CH nodes) through which a routing path passes. In a routing path, the

order of each CH is represented by the locus of chromosome in which the gene of the first locus is always reserved for the source CH nodes. A looping feature of selection (ϕ_s), crossover (ϕ_c), and mutation (ϕ_m) operators is applied on each individual to improve the quality of the solution through the pre-defined probabilities (ϕ_p) until the termination criterion is satisfied. The probability of the mutation rate increases from an extremely low value of 0.01 to its maximum value of 0.05. Highly stable small clustering mechanism is used to organize sensor nodes into a connected hierarchy for distributing energy and data traffic load evenly in the network.

The protocol proposed by Javaid et al. [90] is called balanced energy adaptive routing (BEAR). It exploits the location information, selects the neighbors, chooses the facilitating and successor nodes based on cost function value, and finally selects the forwarder node that has residual energy more than the average residual energy of the network. BEAR allows nodes to communicate directly with the sink node leading to an increase in the number of packets being dropped.

Alageswaran and Swapna [91] proposed an enhanced duty cycled multiple rendezvous multichannel media access control (DMM-MAC) for handling more volume of data in multi-hop underwater wireless sensor networks (UWSNs) for marine eco systems. This protocol distributes bursty sensor data by dynamically tuning the duty cycles, and hence network performance is enhanced dramatically. It is equipped with one modem and the propagation delays or relative distances need not to be known by the other nodes in the network. Less energy is consumed, and it is easy to forward the data based on priority. One cycle consists of several frames and is subdivided into an active section and a sleep section. Each node can use the other time slot when the time slot is free.

5. Major Challenges and Opportunities

5.1. Energy Management

Because marine environment monitoring and protection systems work on harsh environments as mentioned above, battery replacement is difficult and costly. It is therefore very important to have an efficient energy management system together and preferably with an energy harvesting capability.

Typical energy options for sensor nodes include batteries, capacitors, fuel cells, and energy harvesting. A battery is widely used in sensor nodes, and in fact in over half of the systems reviewed under this paper. However, using batteries in sensor nodes has a number of issues [92] including the replacement difficulty, risk of losing power during operation, and environment contamination. It is therefore important to explore alternative power supply options for sensor nodes. Energy harvesting is a natural way to go. In marine environments, energy harvesting options including solar, wind, water waves, and currents. The most outstanding energy harvesting at the moment is photovoltaics (solar energy) [92].

Usually, network energy consumption is increasing at a very high rate due to an increase in data rates, increase in the number of Internet-enabled services and rapid growth of Internet connected edge-devices [21]. The drastic increase in IoT devices requires efficient fabrication of batteries because of the uncertainty in battery dissipation. On the other hand, the devices in marine environment monitoring and protection systems are heterogeneous in nature, each with various capabilities and numerous requirements [93]. Hence, the requirement of cost effective and energy efficient routing strategy in terms of space and time arises in future wireless communication networks.

5.2. Standardization

With the wider scope of marine environment monitoring and protection, international cooperation projects are emerging one after another. According to the interests of the projects, various application platforms and methods have their own characteristics and cannot be compatible with each other. Even though a number of networking protocols and standards for Internet of Things have been proposed and developed [94,95], they are not sufficient for applications in marine environment monitoring. In addition to the standards of IoT networking, it is also important to provide the industry with standards

for IoT devices, equipment, and platforms for marine environment monitoring and protection applications, and providing different levels of governments and marine environment management agencies with standards for marine environment data management, analysis, and reporting. As a result, the standardization of platform development for marine environment monitoring and protection systems brings challenges, including:

1. Standardization of IoT devices specifically for marine environment monitoring and protection, including sensor and actuator nodes, routers and gateways.
2. Standardization of IoT platforms and system technologies for marine environment monitoring and protection, including communication network structures, protocols, and algorithms.
3. Standardization of computing and data storage technologies for marine environment monitoring and protection, including cloud, fog, and edge computing mechanisms, data archiving and warehousing techniques.
4. Standardization of data analysis outputs and reporting formats for exchange among different organizations and governments.

5.3. Marine Environment Protection

Currently, the environmental protection issue is one of the most important issues around the world. The ultimate goal of monitoring the ocean is to protect the marine environment. Most of the current marine environment monitoring applications collected and analyzed massive data from the ocean, but there is no action on protection controls yet. With the advancement and sophistication of IoT and Big Data technologies and their wide applications in marine environment monitoring and protection, we are confident that active marine environment protection measures, technologies, and systems will be developed and deployed in the near future. Massive data collected from marine environments will be analyzed using advanced Big Data analytics and the results will be sent to the related marine environment management agencies/control centers for quick decision making and real time manual interventions in order to protect the marine environment from some disasters (e.g., oil spills and bad weather). On the other hand, some autonomous vessels or other devices (actuators) can also automatically and quickly react to disasters and other events in order to protect the marine environment. In the case of a fishing farm application, data collected from the fish farm environment can be analyzed and used to provide the best fish growth condition and the best feeding.

6. Conclusions

During the past couple of decades, marine environment monitoring has attracted wide attention. Governments and research organizations have heavily invested in the research and development of new technologies in this area. Advanced information and communication technologies have been applied to the development of various marine environment monitoring technologies and systems. Internet of Things has played an important role in this area. This paper presents an updated review of the related technologies and systems on the application of the Internet of Things in marine environment monitoring.

A comprehensive review of about 40 related projects revealed that most systems and applications developed thus far are for ocean sensing and monitoring and water quality monitoring. Some specific efforts have been made for fish farm monitoring, coral reef monitoring, wave and current monitoring. Several projects focused on specific technologies or devices like buoys, energy saving and harvesting devices, routing protocols, data transmitting mechanisms, and data analysis techniques.

It can also be noted that testing places are very much different among these projects. About half of the developed systems have been tested or deployed in real marine or river environments; a number of them were experimented in lab settings or indoor environments; a few others were tested in outdoor pools or small ponds and lakes; and several of them were only tested through simulations.

While most systems use GPRS and/or ZigBee for wireless communication, a few systems used underwater acoustic communication.

Emerging offshore and deep-sea fish farming provides more opportunities and also challenges, not only for marine environment monitoring but also for operation controls and marine environment protection, which is a new area of research.

The energy management issue is mainly considered from two aspects: Reducing energy consumption and using alternative renewable energy sources. Optimizing network topologies and developing advanced routing protocols are the ways to reduce energy consumption. Even though many researchers discussed different options for energy harvesting including solar, wind, waves and ocean currents as alternative renewable energy sources, only solar energy has been used in about a third of the developed systems.

The results of Big Data analytics can be used not only for feedback to marine environment management agencies and control centers for quick decision making and real time manual interventions but also for autonomous vessels and remotely deployed devices to take real time actions in order to protect the marine environment from some disasters (e.g., oil spills and bad weather). It is a growing area of research and development.

This review also identified several research challenges and opportunities, including energy management, standardization of system platforms and technologies, and marine environment protection measures.

Author Contributions: All the co-authors have contributed to the conceptualization of the paper and the review of the cited references. As the first author, G.X. prepared the first draft based on his previously published survey paper. X.S. searched and collected the most recent references and provided a first review of these references. Y.S. contributed to the review of new IoT and wireless communication technologies. W.S. defined the paper structure, re-organized the content, and edited the whole paper.

Funding: This work was supported in part by the Science Foundation for Enhancing School with Innovation of Guangdong Ocean University (Grant Nos. GDOU2015050207).

Conflicts of Interest: No conflict of interests exists in the submission of this manuscript, and the manuscript has been approved by all the co-authors for publication. We would also like to declare that the work described in this paper is original and it has not been published previously, and not under consideration for publication elsewhere, in whole or in part.

References

1. Van Kranenburg, R. *The Internet of Things: A Critique of Ambient Technology and the All-Seeing Network of Rfid*; Institute of Network Cultures: Amsterdam, The Netherlands, 2008.
2. Yang, C.; Shen, W.; Wang, X. Internet of Things in manufacturing: An overview. *IEEE SMC Mag.* **2018**, *4*, 6–15.
3. Cook, D.J.; Crandall, A.S.; Thomas, B.L.; Krishnan, N.C. Casas: A smart home in a box. *Computer* **2013**, *46*, 62–69. [[CrossRef](#)]
4. Finch, E. Is ip everywhere the way ahead for building automation? *Facilities* **2001**, *19*, 396–403. [[CrossRef](#)]
5. Komninos, N.; Philippou, E.; Pitsillides, A. Survey in smart grid and smart home security: Issues, challenges and countermeasures. *IEEE Commun. Surv. Tutor.* **2014**, *16*, 1933–1954. [[CrossRef](#)]
6. Talcott, C. Cyber-physical systems and events. In *Software-Intensive Systems and New Computing Paradigms*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 101–115.
7. Yongfu, L.; Dihua, S.; Weining, L.; Xuebo, Z. A service-oriented architecture for the transportation cyber-physical systems. In Proceedings of the 31st Control Conference (CCC), Hefei, China, 25–27 July 2012; pp. 7674–7678.
8. Ungurean, I.; Gaitan, N.-C.; Gaitan, V.G. An IoT architecture for things from industrial environment. In Proceedings of the 10th International Conference on Communications (COMM), Bucharest, Romania, 29–31 May 2014; pp. 1–4.
9. Wang, C.; Bi, Z.; Da Xu, L. IoT and cloud computing in automation of assembly modeling systems. *IEEE Trans. Ind. Inform.* **2014**, *10*, 1426–1434. [[CrossRef](#)]

10. Catarinucci, L.; De Donno, D.; Mainetti, L.; Palano, L.; Patrono, L.; Stefanizzi, M.L.; Tarricone, L. An IoT-aware architecture for smart healthcare systems. *IEEE Internet Things J.* **2015**, *2*, 515–526. [\[CrossRef\]](#)
11. Yan, Y.; Qian, Y.; Sharif, H.; Tipper, D. A survey on smart grid communication infrastructures: Motivations, requirements and challenges. *IEEE Commun. Surv. Tutor.* **2013**, *15*, 5–20. [\[CrossRef\]](#)
12. Dameri, R.P.; Rosenthal-Sabroux, C. *Smart City: How to Create Public and Economic Value with High Technology in Urban Space*; Springer: Cham, Switzerland, 2014.
13. Boonma, P.; Suzuki, J. An adaptive, scalable and self-healing sensor network architecture for autonomous coastal environmental monitoring. In Proceedings of the IEEE Conference on Technologies for Homeland Security, Woburn, MA, USA, 16–17 May 2007; p. 18.
14. Hadim, S.; Mohamed, N. Middleware: Middleware challenges and approaches for wireless sensor networks. *IEEE Distrib. Syst. Online* **2006**, *7*, 1. [\[CrossRef\]](#)
15. Albaladejo, C.; Sánchez, P.; Iborra, A.; Soto, F.; López, J.A.; Torres, R. Wireless sensor networks for oceanographic monitoring: A systematic review. *Sensors* **2010**, *10*, 6948–6968. [\[CrossRef\]](#)
16. Alippi, C.; Camplani, R.; Galperti, C.; Roveri, M. Effective design of wsns: From the lab to the real world. In Proceedings of the 2008 3rd International Conference on Sensing Technology (ICST), Tainan, Taiwan, 30 November–3 December 2008; pp. 1–9.
17. Cardei, M.; Wu, J. Energy-efficient coverage problems in wireless ad-hoc sensor networks. *Comput. Commun.* **2006**, *29*, 413–420. [\[CrossRef\]](#)
18. Xu, G.; Shen, W.; Wang, X. Applications of wireless sensor networks in marine environment monitoring: A survey. *Sensors* **2014**, *14*, 16932–16954. [\[CrossRef\]](#)
19. Zhang, X. Application of Internet of Things technology in agricultural production. In Proceedings of the 2018 the International Symposium on Big Data and Artificial Intelligence, Hong Kong, China, 29–30 December 2018; pp. 269–274.
20. Antão, L.; Pinto, R.; Reis, J.; Gonçalves, G. Requirements for testing and validating the industrial Internet of Things. In Proceedings of the 2018 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW), Vasteras, Sweden, 9–13 April 2018; pp. 110–115.
21. Khan, R.; Khan, S.U.; Zaheer, R.; Khan, S. Future Internet: The Internet of Things architecture, possible applications and key challenges. In Proceedings of the 2012 10th International Conference on Frontiers of Information Technology (FIT), Islamabad, India, 17–19 December 2012; pp. 257–260.
22. Yang, Z.; Yue, Y.; Yang, Y.; Peng, Y.; Wang, X.; Liu, W. Study and application on the architecture and key technologies for IoT. In Proceedings of the 2011 International Conference on Multimedia Technology (ICMT), Hangzhou, China, 26–28 July 2011; pp. 747–751.
23. Wu, M.; Lu, T.-J.; Ling, F.-Y.; Sun, J.; Du, H.-Y. Research on the architecture of Internet of Things. In Proceedings of the 2010 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE), Chengdu, China, 20–22 August 2010; pp. V5:484–V485:487.
24. Li, M.; Yang, B. A survey on topology issues in wireless sensor network. In Proceedings of the 2006 International Conference on Wireless Networks (ICWN), Las Vegas, NV, USA, 26–29 June 2006; p. 503.
25. Zeng, Y.; Sreenan, C.J.; Xiong, N.; Yang, L.T.; Park, J.H. Connectivity and coverage maintenance in wireless sensor networks. *J. Supercomput.* **2010**, *52*, 23–46. [\[CrossRef\]](#)
26. Flammini, A.; Ferrari, P.; Marioli, D.; Sisinni, E.; Taroni, A. Wired and wireless sensor networks for industrial applications. *Microelectron. J.* **2009**, *40*, 1322–1336. [\[CrossRef\]](#)
27. Kim, S.-Y.; Guzide, O.; Cook, S. Towards an optimal network topology in wireless sensor networks: A hybrid approach. In Proceedings of the ISCA First International Conference on Sensor Networks and Applications, San Francisco, CA, USA, 4–6 November 2009.
28. Anastasi, G.; Conti, M.; Di Francesco, M.; Passarella, A. Energy conservation in wireless sensor networks: A survey. *Ad Hoc Netw.* **2009**, *7*, 537–568. [\[CrossRef\]](#)
29. Porter, J.; Arzberger, P.; Braun, H.-W.; Bryant, P.; Gage, S.; Hansen, T.; Hanson, P.; Lin, C.-C.; Lin, F.-P.; Kratz, T. Wireless sensor networks for ecology. *AIBS Bull.* **2005**, *55*, 561–572. [\[CrossRef\]](#)
30. Jiang, Z. User density in a cluster underwater acoustic network. *Int. J. Intell. Control Syst.* **2012**, *17*, 31–40.
31. Jiang, Z. Underwater acoustic networks—issues and solutions. *Int. J. Intell. Control Syst.* **2008**, *13*, 152–161.
32. Zhu, Y.; Peng, Z.; Cui, J.-H.; Chen, H. Toward practical mac design for underwater acoustic networks. *IEEE Trans. Mobile Comput.* **2015**, *14*, 872–886. [\[CrossRef\]](#)

33. Zixuan, Y.; Zhifang, W.; Chang, L. Research on marine environmental monitoring system based on the Internet of Things technology. In Proceedings of the IEEE International Conference on Electronic Information and Communication Technology (ICEICT), Harbin, China, 20–22 August 2016; pp. 121–125.
34. Pérez, C.A.; Jimenez, M.; Soto, F.; Torres, R.; López, J.; Iborra, A. A system for monitoring marine environments based on wireless sensor networks. In Proceedings of the OCEANS 2011 IEEE—Spain, Santander, Spain, 6–9 June 2011; pp. 1–6.
35. Voigt, T.; Osterlind, F.; Finne, N.; Tsiftes, N.; He, Z.; Eriksson, J.; Dunkels, A.; Bamstedt, U.; Schiller, J.; Hjort, K. Sensor networking in aquatic environments-experiences and new challenges. In Proceedings of the 32nd IEEE Conference on Local Computer Networks, Dublin, Ireland, 15–18 October 2007; pp. 793–798.
36. Vesecky, J.F.; Laws, K.; Petersen, S.I.; Bazeghi, C.; Wiberg, D. Prototype autonomous mini-buoy for use in a wireless networked, ocean surface sensor array. In Proceedings of the 2007 IEEE Conference on Geoscience and Remote Sensing Symposium (IGARSS), Barcelona, Spain, 23–27 July 2007; pp. 4987–4990.
37. Liu, K.; Yang, Z.; Li, M.; Guo, Z.; Guo, Y.; Hong, F.; Yang, X.; He, Y.; Feng, Y.; Liu, Y. Oceansense: Monitoring the sea with wireless sensor networks. *ACM SIGMOBILE Mobile Comput. Commun. Rev.* **2010**, *14*, 7–9. [[CrossRef](#)]
38. Macias, E.; Suarez, A.; Chiti, F.; Sacco, A.; Fantacci, R. A hierarchical communication architecture for oceanic surveillance applications. *Sensors* **2011**, *11*, 11343–11356. [[CrossRef](#)]
39. Roadknight, C.; Parrott, L.; Boyd, N.; Marshall, I.W. A layered approach to in situ data management on a wireless sensor network. In Proceedings of the 2004 Intelligent Sensors, Sensor Networks and Information Processing Conference, Melbourne, VIC, Australia, 14–17 December 2004; pp. 85–90.
40. Cella, U.M.; Shuley, N.; Johnstone, R. Wireless sensor networks in coastal marine environments: A study case outcome. In Proceedings of the Fourth ACM International Workshop on UnderWater Networks, Berkeley, CA, USA, 3 November 2009; p. 8.
41. Jiang, M.; Guo, Z.; Hong, F.; Ma, Y.; Luo, H. Oceansense: A practical wireless sensor network on the surface of the sea. In Proceedings of the IEEE International Conference on Pervasive Computing and Communications(PerCom), Galveston, TX, USA, 9–13 March 2009; pp. 1–5.
42. Tao, C.; Hanfei, Z.; Ming, C.; Guofu, F. Implementation study of a home position monitoring system for marine environment. In Proceedings of the 2010 2nd IEEE International Conference on Information Management and Engineering (ICIME), Chengdu, China, 16–18 April 2010; pp. 224–231.
43. Alippi, C.; Camplani, R.; Galperti, C.; Roveri, M. A robust, adaptive, solar-powered wsn framework for aquatic environmental monitoring. *IEEE Sens. J.* **2011**, *11*, 45–55. [[CrossRef](#)]
44. De Marziani, C.; Alcoleas, R.; Colombo, F.; Costa, N.; Pujana, F.; Colombo, A.; Aparicio, J.; Alvarez, F.; Jimenez, A.; Urena, J. A low cost reconfigurable sensor network for coastal monitoring. In Proceedings of the OCEANS 2011 IEEE—Spain, Santander, Spain, 6–9 June 2011; pp. 1–6.
45. Albaladejo, C.; Soto, F.; Torres, R.; Sánchez, P.; López, J.A. A low-cost sensor buoy system for monitoring shallow marine environments. *Sensors* **2012**, *12*, 9613–9634. [[CrossRef](#)]
46. Al-Zaidi, R.; Woods, J.C.; Al-Khalidi, M.; Hu, H. Building novel vhf-based wireless sensor networks for the Internet of Marine Things. *IEEE Sens. J.* **2018**, *18*, 2131–2144. [[CrossRef](#)]
47. Ferreira, H.; Silva, F.; Sousa, P.; Matias, B.; Faria, A.; Oliveira, J.; Almeida, J.M.; Martins, A.; Silva, E. Autonomous systems in remote areas of the ocean using bluecom+ communication network. In Proceedings of the OCEANS—Anchorage, Anchorage, AK, USA, 18–21 September 2017; pp. 1–6.
48. Kaur, S.; Jain, S.; Virmani, D. Deployment of wireless sensor networks for intelligent information retrieval in marine environment. In Proceedings of the 2015 IEEE International Conference on Control System, Computing and Engineering (ICCSCE), George Town, Malaysia, 27–29 November 2015; pp. 371–376.
49. Hu, Y.; Zheng, Y.; Liu, H. A route maintenance method for marine mobile sensor network based on ring broadcast mechanism. In Proceedings of the 2018 18th IEEE International Conference on Communication Technology (ICCT), Chongqing, China, 8–11 October 2018; Institute of Electrical and Electronics Engineers Inc.: Chongqing, China, 2018; pp. 533–537.
50. Mourya, R.; Saafin, W.; Dragone, M.; Petillot, Y. Ocean monitoring framework based on compressive sensing using acoustic sensor networks. In Proceedings of the OCEANS 2018 MTS/IEEE Charleston, Charleston, SC, USA, 22–25 October 2018; Institute of Electrical and Electronics Engineers Inc.: Charleston, SC, USA, 2018.

51. Morozs, N.; Mitchell, P.D.; Zakharov, Y.; Sherlock, B.; Neasham, J.A.; Tsimenidis, C.C.; Mourya, R.; Petillot, Y.R.; Gibney, T.; Dragone, M.; et al. Robust tda-mac for practical underwater sensor network deployment: Lessons from usmart sea trials. In Proceedings of the 13th ACM International Conference on Underwater Networks and Systems, WUWNet 2018, Shenzhen, China, 3–5 December 2018; Association for Computing Machinery: Shenzhen, China, 2018.
52. Song, Y.; Zhang, X.; Zhang, M.; Wang, J. Research of node localization algorithm based on wireless sensor networks in marine environment monitoring. *J. Comput. Methods Sci. Eng.* **2018**, *18*, 69–83. [\[CrossRef\]](#)
53. Yang, X.; Ong, K.G.; Dreschel, W.R.; Zeng, K.; Mungle, C.S.; Grimes, C.A. Design of a wireless sensor network for long-term, in-situ monitoring of an aqueous environment. *Sensors* **2002**, *2*, 455–472. [\[CrossRef\]](#)
54. Seders, L.A.; Shea, C.A.; Lemmon, M.D.; Maurice, P.A.; Talley, J.W. Lakenet: An integrated sensor network for environmental sensing in lakes. *Environ. Eng. Sci.* **2007**, *24*, 183–191. [\[CrossRef\]](#)
55. Regan, F.; Lawlor, A.; Flynn, B.O.; Torres, J.; Martinez-Catala, R.; O'Mathuna, C.; Wallace, J. A demonstration of wireless sensing for long term monitoring of water quality. In Proceedings of the 2009 IEEE 34th Conference on Local Computer Networks, Zurich, Switzerland, 20–23 October 2009; pp. 819–825.
56. O'Connor, E.; Zhang, D.; Smeaton, A.F.; O'Connor, N.E.; Regan, F. Multi-modal sensor networks for more effective sensing in irish coastal and freshwater environments. In Proceedings of the 2012 Oceans, Hampton Roads, VA, USA, 14–19 October 2012; pp. 1–9.
57. Hadjimitsis, D.G.; Hadjimitsis, M.G.; Themistocleous, K.; Agapiou, A. Integration of micro-sensor technology and remote sensing for monitoring coastal water quality in a municipal beach and other areas in cyprus. In Proceedings of the Remote Sensing for Agriculture, Ecosystems, and Hydrology XI, Berlin, Germany, 1–3 September 2009; p. 74720p.
58. Jin, N.; Ma, R.; Lv, Y.; Lou, X.; Wei, Q. A novel design of water environment monitoring system based on wsn. In Proceedings of the 2010 International Conference on Computer Design and Applications (ICCD), Qinhuaingdao, China, 25–27 June 2010; pp. V2:593–V592:597.
59. Alkandari, A.; Alabduljader, Y.; Moein, S.M. Water monitoring system using wireless sensor network (wsn): Case study of kuwait beaches. In Proceedings of the 2012 Second International Conference on Digital Information Processing and Communications (ICDIPC), Klaipeda City, Lithuania, 10–12 July 2012; pp. 10–15.
60. Adamo, F.; Attivissimo, F.; Carducci, C.G.C.; Lanzolla, A.M.L. A smart sensor network for sea water quality monitoring. *IEEE Sens. J.* **2015**, *15*, 2514–2522. [\[CrossRef\]](#)
61. Bromage, M.; Obraczka, K.; Potts, D. Sea-labs: A wireless sensor network for sustained monitoring of coral reefs. In *NETWORKING 2007. Ad Hoc and Sensor Networks, Wireless Networks, Next Generation Internet*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 1132–1135.
62. Berlian, M.H.; Sahputra, T.E.R.; Ardi, B.J.W.; Dzatmika, L.W.; Besari, A.R.A.; Sudibyo, R.W.; Sukaridhoto, S. Design and implementation of smart environment monitoring and analytics in real-time system framework based on Internet of Underwater Things and Big Data. In Proceedings of the 2016 International Conference on Electronics Symposium (IES), Denpasar, Indonesia, 29–30 September 2016; pp. 403–408.
63. López, M.; Gómez, J.; Sabater, J.; Herms, A. IEEE 802.15. 4 based wireless monitoring of ph and temperature in a fish farm. In Proceedings of the MELECON 2010 15th IEEE Mediterranean electrotechnical conference, Valletta, Malta, 26–28 April 2010; pp. 575–580.
64. Yang, H.; Wu, H.; He, Y. Architecture of wireless sensor network for monitoring aquatic environment of marine shellfish. In Proceedings of the 2009 7th Asian Control Conference (ASCC), Hong Kong, China, 27–29 August 2009; pp. 1147–1151.
65. Leblond, E.; Lazure, P.; Laurans, M.; Rioual, C.; Woerther, P.; Quemener, L.; Berthou, P. The recopesca project: A new example of participative approach to collect fisheries and in situ environmental data. *Mercat. Ocean-Q. Newsl.* **2010**, 40–48.
66. Lloret, J.; Garcia, M.; Sendra, S.; Lloret, G. An underwater wireless group-based sensor network for marine fish farms sustainability monitoring. *Telecommun. Syst.* **2015**, *60*, 67–84. [\[CrossRef\]](#)
67. Meera, M.S.; Rao, S.N. Comparative analysis of IoT protocols for a marine IoT system. In Proceedings of the 2018 7th International Conference on Advances in Computing, Communications and Informatics (ICACCI), Bangalore, India, 19–22 September 2018; Institute of Electrical and Electronics Engineers Inc.: Bangalore, India, 2018; pp. 2049–2053.

68. Lloret, J.; Sendra, S.; Garcia, M.; Lloret, G. Group-based underwater wireless sensor network for marine fish farms. In Proceedings of the GLOBECOM Workshops (GC Wkshps), Houston, TX, USA, 5–9 December 2011; pp. 115–119.
69. Marimon, M.C.; Tangonan, G.; Libatique, N.J.; Sugimoto, K. Development and evaluation of wave sensor nodes for ocean wave monitoring. *IEEE Syst. J.* **2015**, *9*, 292–302. [[CrossRef](#)]
70. Chen, H.; Chen, H.; Xie, L.; Wang, K. Temporal tracking of ocean current field using the underwater acoustic sensor network. In Proceedings of the 2018 OCEANS—MTS/IEEE Kobe Techno-Oceans, Kobe, Japan, 28–31 May 2018; Institute of Electrical and Electronics Engineers Inc.: Kobe, Japan, 2018.
71. Yang, J.; Wen, J.; Jiang, B.; Lv, Z.; Sangaiah, A.K. Marine depth mapping algorithm based on the edge computing in Internet of Things. *J. Parallel Distrib. Comput.* **2018**, *114*, 95–103. [[CrossRef](#)]
72. Blix, K.; Eltoft, T. Machine learning automatic model selection algorithm for oceanic chlorophyll-a content retrieval. *Remote Sens.* **2018**, *10*, 775. [[CrossRef](#)]
73. Zhong, Y.; Zhang, L.; Xing, S.; Li, F.; Wan, B. The big data processing algorithm for water environment monitoring of the three gorges reservoir area. *Abstr. Appl. Anal.* **2014**, *2014*, 698632. [[CrossRef](#)]
74. Addison, P.; Collins, D.; Trebilco, R.; Howe, S.; Bax, N.; Hedge, P.; Jones, G.; Miloslavich, P.; Roelfsema, C.; Sams, M. A new wave of marine evidence-based management: Emerging challenges and solutions to transform monitoring, evaluating, and reporting. *ICES J. Mar. Sci.* **2017**, *75*, 941–952. [[CrossRef](#)]
75. Belghith, E.H.; Rioult, F.; Bouzidi, M. Acoustic diversity classifier for automated marine big data analysis. In Proceedings of the 2018 30th International Conference on Tools with Artificial Intelligence (ICTAI), Volos, Greece, 5–7 November 2018; IEEE Computer Society: Volos, Greece, 2018; pp. 130–136.
76. Li, Z.; Wang, N.; Li, Y.; Sun, X.; Huo, M.; Zhang, H. Collective efficacy of support vector regression with smoothness priority in marine sensor data prediction. *IEEE Access* **2019**, *7*, 10308–10317. [[CrossRef](#)]
77. Song, J.H.; Xie, H.; Feng, Y. Correlation analysis method for ocean monitoring big data in a cloud environment. *J. Coast. Res.* **2018**, *24*, 24–28.
78. Radeta, M.; Nunes, N.J.; Vasconcelos, D.; Nisi, V. Poseidon—Passive-acoustic ocean sensor for entertainment and interactive data-gathering in opportunistic nautical-activities. In Proceedings of the 2018 Designing Interactive Systems Conference (DIS), Hong Kong, China, 9–13 June 2018; Association for Computing Machinery, Inc.: Hong Kong, China, 2018; pp. 999–1012.
79. Coutinho, R.W.; Boukerche, A.; Vieira, L.F.; Loureiro, A.A. Underwater wireless sensor networks: A new challenge for topology control-based systems. *ACM Comput. Surv. (CSUR)* **2018**, *51*, 19. [[CrossRef](#)]
80. Gkikopoulou, A.; Nikolakopoulos, G.; Manesis, S. A survey on underwater wireless sensor networks and applications. In Proceedings of the 2012 20th Mediterranean Conference on Control & Automation (MED), Barcelona, Spain, 3–6 July 2012; pp. 1147–1154.
81. Felemban, E.; Shaikh, F.K.; Qureshi, U.M.; Sheikh, A.A.; Qaisar, S.B. Underwater sensor network applications: A comprehensive survey. *Int. J. Distrib. Sens. Netw.* **2015**, *1*–14. [[CrossRef](#)]
82. Domingo, M.C. An overview of the Internet of Underwater Things. *J. Netw. Comput. Appl.* **2012**, *35*, 1879–1890. [[CrossRef](#)]
83. Kim, S.; Park, S.; Yoo, Y. Dynamic transmission power control based on exact sea surface movement modeling in underwater acoustic sensor networks. In Proceedings of the 2014 IEEE 10th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), Larnaca, Cyprus, 8–10 October 2014; pp. 666–672.
84. Bai, W.; Wang, H.; Shen, X.; Zhao, R. Link scheduling method for underwater acoustic sensor networks based on correlation matrix. *IEEE Sens. J.* **2015**, *16*, 4015–4022. [[CrossRef](#)]
85. Su, R.; Venkatesan, R.; Li, C. An energy-efficient asynchronous wake-up scheme for underwater acoustic sensor networks. *Wirel. Commun. Mobile Comput.* **2016**, *16*, 1158–1172. [[CrossRef](#)]
86. Coutinho, R.W.; Boukerche, A.; Vieira, L.F.; Loureiro, A.A. Modeling the sleep interval effects in duty-cycled underwater sensor networks. In Proceedings of the 2016 IEEE International Conference on Communications (ICC), Kuala Lumpur, Malaysia, 22–27 May 2016; pp. 1–6.
87. Khan, F.A.; Khan, S.A.; Turgut, D.; Boloni, L. Optimizing resurfacing schedules to maximize value of information in uwsns. In Proceedings of the 2016 IEEE Global Communications Conference (GLOBECOM), Washington, DC, USA, 4–8 December 2016; pp. 1–5.
88. Coutinho, R.W.; Boukerche, A.; Vieira, L.F.; Loureiro, A.A. A novel void node recovery paradigm for long-term underwater sensor networks. *Ad Hoc Netw.* **2015**, *34*, 144–156. [[CrossRef](#)]

89. Faheem, M.; Tuna, G.; Gungor, V.C. Qerp: Quality-of-service (qos) aware evolutionary routing protocol for underwater wireless sensor networks. *IEEE Syst. J.* **2018**, *12*, 2066–2073. [CrossRef]
90. Javaid, N.; Cheema, S.; Akbar, M.; Alrajeh, N.; Alabed, M.S.; Guizani, N. Balanced energy consumption based adaptive routing for IoT enabling underwater wsns. *IEEE Access* **2017**, *5*, 40–51. [CrossRef]
91. Alageswaran, R.; Swapna, P. Enhanced dynamic duty cycled multiple rendezvous multi-channel media access control (dmm-mac) protocol for underwater sensor network based marine eco system. *Biomed. Res.* **2016**, 118–122.
92. Knight, C.; Davidson, J.; Behrens, S. Energy options for wireless sensor nodes. *Sensors* **2008**, *8*, 8037–8066. [CrossRef]
93. Javaid, N.; Sher, A.; Nasir, H.; Guizani, N. Intelligence in IoT-based 5G networks: Opportunities and challenges. *IEEE Commun. Mag.* **2018**, *56*, 94–100. [CrossRef]
94. Salman, T.; Jain, R. Networking protocols and standards for Internet of Things. 2017. Available online: <https://onlinelibrary.wiley.com/doi/abs/10.1002/9781119173601.ch13> (accessed on 17 March 2019). [CrossRef]
95. Vidales, M. 802.15.4 Wireless for Internet of Things Developers. Available online: <https://blog.helium.com/802-15-4-wireless-for-internet-of-things-developers-1948fc313b2e> (accessed on 17 March 2019).



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

TITLE/ AUTHOR	Internet of Underwater Things and Big Marine Data Analytics—A Comprehensive Survey / M. Jahanbakht, W. Xiang, L. Hanzo and M. Rahimi Azghadi
SOURCE	IEEE Communications Surveys & Tutorials Volume 23, No. 2, Pages 904-956, Secondquarter 2021 https://doi.org/10.1109/COMST.2021.3053118 (Database: ResearchGate)

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/346901591>

Internet of Underwater Things and Big Marine Data Analytics – A Comprehensive Survey

Article in IEEE Communications Surveys & Tutorials · January 2021

DOI: 10.1109/COMST.2021.3053118

CITATIONS

52

READS

613

4 authors:



Mohammad Jahanbakht

Islamic Azad University

28 PUBLICATIONS 157 CITATIONS

[SEE PROFILE](#)



L. Hanzo

University of Southampton

2,790 PUBLICATIONS 53,164 CITATIONS

[SEE PROFILE](#)



Mostafa Rahimi Azghadi

James Cook University

86 PUBLICATIONS 1,673 CITATIONS

[SEE PROFILE](#)



Wei Xiang

La Trobe University

362 PUBLICATIONS 6,893 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Physical layer security [View project](#)



Improper Gaussian Signaling [View project](#)

Internet of Underwater Things and Big Marine Data Analytics – A Comprehensive Survey

Mohammad Jahanbakht^{ID}, *Student Member, IEEE*, Wei Xiang^{ID}, *Senior Member, IEEE*,
Lajos Hanzo^{ID}, *Fellow, IEEE*, and Mostafa Rahimi Azghadi^{ID}, *Senior Member, IEEE*

Abstract—The Internet of Underwater Things (IoUT) is an emerging communication ecosystem developed for connecting underwater objects in maritime and underwater environments. The IoUT technology is intricately linked with intelligent boats and ships, smart shores and oceans, automatic marine transportations, positioning and navigation, underwater exploration, disaster prediction and prevention, as well as with intelligent monitoring and security. The IoUT has an influence at various scales ranging from a small scientific observatory, to a mid-sized harbor, and to covering global oceanic trade. The network architecture of IoUT is intrinsically heterogeneous and should be sufficiently resilient to operate in harsh environments. This creates major challenges in terms of underwater communications, whilst relying on limited energy resources. Additionally, the volume, velocity, and variety of data produced by sensors, hydrophones, and cameras in IoUT is enormous, giving rise to the concept of Big Marine Data (BMD), which has its own processing challenges. Hence, conventional data processing techniques will falter, and bespoke Machine Learning (ML) solutions have to be employed for automatically learning the specific BMD behavior and features facilitating knowledge extraction and decision support. The motivation of this paper is to comprehensively survey the IoUT, BMD, and their synthesis. It also aims for exploring the nexus of BMD with ML. We set out from underwater data collection and then discuss the family of IoUT data communication techniques with an emphasis on the state-of-the-art research challenges. We then review the suite of ML solutions suitable for BMD handling and analytics. We treat the subject deductively from an educational perspective, critically appraising the material surveyed. Accordingly, the reader will become familiar with the pivotal issues of IoUT and BMD processing, whilst gaining an insight into the state-of-the-art applications, tools, and techniques. Finally, we analyze the architectural challenges of the IoUT, followed by proposing a range of promising direction for research and innovation in the broad areas of IoUT and BMD. Our hope is to inspire researchers, engineers, data scientists, and governmental bodies to further progress the field, to develop new tools and techniques,

as well as to make informed decisions and set regulations related to the maritime and underwater environments around the world.

Index Terms—Internet of Things, Big Data, Underwater Network Architecture, Data Acquisition, Marine and Underwater Databases/Datasets, Underwater Wireless Sensor Network, Image and Video Processing, Machine Learning, Deep Neural Networks.

NOMENCLATURE

ASIC	Application-Specific Integrated Circuit
AUV	Autonomous Underwater Vehicles
BER	Bit Error Rate
BMD	Big Marine Data
CNN	Convolutional Neural Networks
DBN	Deep Belief Network
DCS	Distributed Computing System
DL	Deep Learning
FPGA	Field-Programmable Gate Array
GIS	Geographic Information System
GPU	Graphics Processing Unit
IaaS	Infrastructure as a Service
IoT	Internet of Things
IoUT	Internet of Underwater Things
IP	Internet Protocol
MAC	Media Access Control
MEC	Mobile Edge Computing
ML	Machine Learning
NN	Neural Network
PaaS	Platform as a Service
QoS	Quality of Service
RBM	Restricted Boltzmann Machine
RFID	Radio Frequency Identification
RNN	Recurrent Neural Network
RNTN	Recursive Neural Tensor Network
ROV	Remotely Operated Vehicles
SaaS	Software as a Service
SDN	Software-Defined Network
SLAM	Simultaneous Localization and Mapping
SNR	Signal-to-Noise Ratio
SVM	Support Vector Machine
TCP	Transmission Control Protocol
ToF	Time of Flight
UWSN	Underwater Wireless Sensor Network

This work is funded by the Australian Government Research Training Program Scholarship, and in part by Beijing Natural Science Foundation under Grant L182032 (Corresponding authors: Wei Xiang and M. Rahimi Azghadi).

M. Jahanbakh and M. Rahimi Azghadi are with the College of Science and Engineering, James Cook University, Queensland, Australia (e-mail: mohammad.jahanbakht@my.jcu.edu.au; mostafa.rahimiazghadi@jcu.edu.au).

Wei Xiang is with the School of Engineering and Mathematical Sciences, La Trobe University, Melbourne, VIC 3086, Australia (e-mail: w.xiang@latrobe.edu.au).

L. Hanzo is with the School of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, U.K. (e-mail: lh@ecs.soton.ac.uk).

L. Hanzo would like to acknowledge the financial support of the Engineering and Physical Sciences Research Council projects EP/N004558/1, EP/P034284/1, EP/P034284/1, EP/P003990/1 (COALESCE), of the Royal Society's Global Challenges Research Fund Grant as well as of the European Research Council's Advanced Fellow Grant QuantCom.

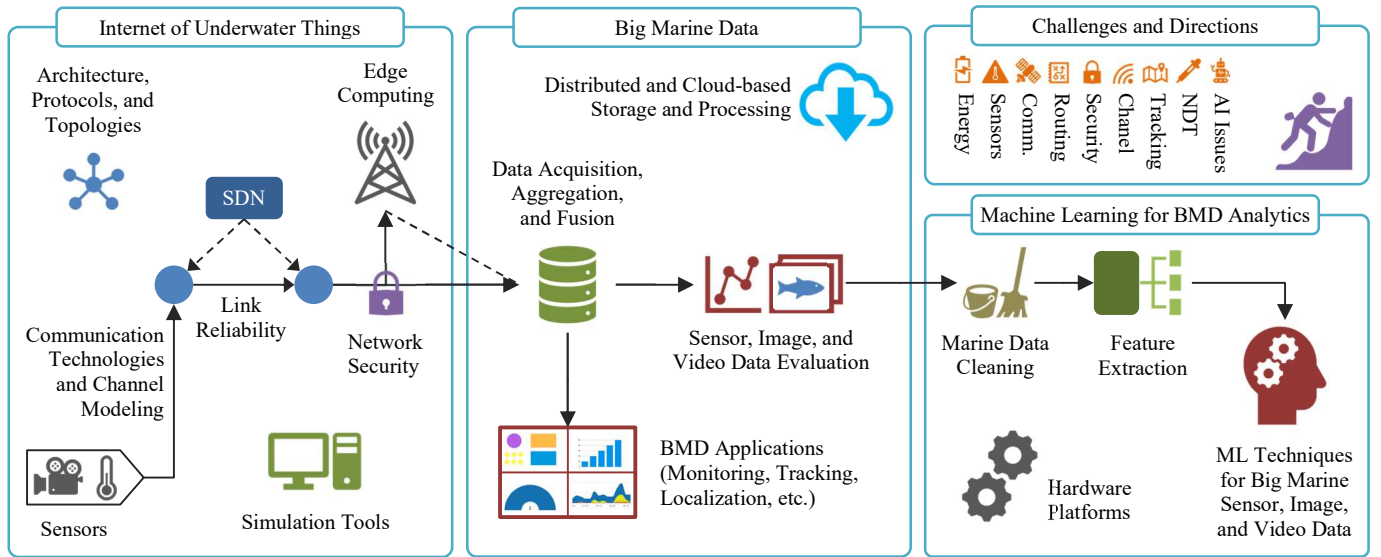


Fig. 1. Qualitative relationship between the diverse system components of the IoUT and BMD analytics, starting from underwater sensors and ending up to ML solutions and future directions.

I. INTRODUCTION

THE INTERNET OF THINGS (IoT) augmented with machine intelligence and big data analytics is expected to transform and revolutionize the way we live in almost every technological area. Broadly speaking, IoT can be defined as an infrastructure of the information society that connects equipment/devices (things) to the Internet and to one another. By this means, the IoT could connect devices in any place on earth to help us have better interaction with our living environment [1].

To date, the existing networks in terrestrial and urban areas have been the domain of influence for the IoT and have been researched extensively. This has made a fairly strong foundation for the industrial IoT developments, which are emerging with an astonishing pace at the time of writing [2]. However, the underwater section of IoT, i.e. IoUT has not attracted as much attention as it deserves and it is a rather unexplored research area. This is mainly because underwater applications are still in their infancy and the new era of scientific endeavor to better understand, control, and interact with the oceans and seas through underwater technologies is yet to flourish.

Although 44% of the earth's population lives within 150 km of the sea, 95% of sea area remains unexplored by the humankind [3]. Oceans cover more than 70% of the earth's surface and 90% of international trades are through nautical transportations [4]. Astonishingly, 12 people have spent 300 hours on the surface of the moon, while only 3 people have spent about 3 hours at 6 km depth of the ocean. In addition, about 90 million tons of salt-water fish are caught worldwide each year, and the coral reefs are estimated to provide food for almost 500 million people [5]. Hence, underwater research and development could have a significant impact on many aspects of human's life by establishing and rolling out the IoUT.

"On the surface of the ocean, men wage war and destroy each other; but down here, just a few feet beneath the surface, there is a calm and peace, unmolested by man."

— Jules Verne

Although the IoUT has many technical similarities with its ground-based counterpart (IoT) such as its structure and function, it has many technical differences arising from its different communication/telecommunication environments, computational limitations, and constrained energy resources. To address these gaps between the IoT and IoUT, technical concepts in the field of IoUT will be extensively discussed. These include both the underwater communications [6]–[8], as well as the underwater sensors and devices [9].

By connecting an increasing number of devices and machines to the Internet, the IoT and IoUT ecosystems produce enormous amounts of data. This high volume of data is referred to in parlance as *big data*. Big data is currently being generated by various technological ecosystems and perhaps the most ubiquitous data types in today's world is the data produced throughout the IoT. This is also set to increase, since the number of Internet-connected devices is projected to increase from the current 30 billion to over 50 billion by 2020 [10].

In the age of sparse data production, analytical mathematics and statistical techniques, widely known as data mining, were employed to infer knowledge from data. However, in the current era of data proliferation, when the produced data volume in the last five years exceeds the whole amount of data generated before that, conventional data processing techniques will soon fall short [11], [12]. These traditional big data handling methods relying on statistical descriptive, predictive, and prescriptive analytics usually suffer from the lack of generalization. That is, they cannot automatically learn the behavior and features in smaller datasets and use them in big data scenarios.

Internet of Underwater Things and Big Marine Data Analytics	
I.	Introduction
II.	Internet of Underwater Things
A	Underwater Sensors and Devices for Data Collection
B	IoUT Communications
C	Improving IoUT Link Reliability
D	IoUT Network Architecture
E	Underwater Channel Modeling
F	IoUT Network Topologies
G	Underwater Network Simulation Tools
H	IoUT Network Security
I	Software-Defined Networks in IoUT
J	Edge Computing in IoUT
K	Section Summary
III.	Big Marine Data
A	Marine Data Acquisition
B	Marine Data Classification
C	Distributed and Cloud-based BMD Processing
D	Marine Data Applications
E	Section Summary
IV.	Machine Learning for BMD Analytics
A	Machine Learning Techniques
B	Deep Learning Frameworks and Libraries
C	Marine Data Cleaning for Machine Learning
D	Feature Extraction for Marine Data Analytics
E	Hardware Platforms for ML in IoUT
F	ML Techniques in BMD Applications
G	Section Summary
V.	Challenges and Future Directions in IoUT and BMD
A	Underwater Network Management System
B	Energy conservation and harvesting in IoUT devices
C	Development of Low-cost and Affordable Sensors
D	Large-scale IoUT Underwater Communications
E	Dynamic IoUT Signal Routing and Traffic Control
F	Securing Underwater Networks
G	Deleterious Effects of Imprecise Channel Modeling
H	Sparse and High-maintenance Sensing Devices in IoUT
I	Poor Underwater Positioning and Navigation
J	Non-Destructive Testing in Underwater Applications
K	Lack of Strong Data Leveraging Tools
L	Training Deep Networks
M	Degraded Underwater Images
VI.	Conclusion

Fig. 2. Outline at a glance.

To address this significant problem, machine learning has risen as one of the practical solutions. ML has been created to facilitate an automatic approach to learning and extracting knowledge from data. This could revolutionize various aspects of our lives, ranging from treating formidable diseases, to boosting the economies, to understanding the universe, to defense and military decisions [13]–[15]. As listed below, ML has also been used in a variety of sparse and big underwater data applications, including:

- *Evaluation and discoveries*: Examples of these include the evaluation of corals and their inhabitants [16], seabed analysis and mapping (photo-mosaicking) [17], object classification and discovery [18], [19], plant identification [20], the automatic recognition of fish [14], [21], lobster [22], [23], plankton [24], and other species, as well as tracking and direction finding [25].
- *Monitoring and management*: Examples of these include environmental monitoring (e.g. water quality and pollution) [26], fish farming [4], pipeline monitoring and corrosion investigation (e.g. in oil and gas industry) [27], [28], harbor security and military surveillance [3], navigation assistance [29], marine forecast and warning systems (e.g. tsunami, red-tide, flood) [30], and maritime geographic information systems [31].

ML as an indispensable tool in the IoUT, offers intelligent solutions for analyzing BMD, and thus it will be thoroughly investigated in this paper.

This paper is motivated by the fact that the IoUT, BMD, and machine/deep learning are salient topics emerging in the scientific literature. Nevertheless, there is no comprehensive survey to cover the joint applications of these three. In other words, many previous articles can be found in the literature that cover IoT [1], [10], big data [11], the joint aspects of IoT and big data [32], and even big data analytics in IoT [12], [33]. In a clear contrast, the amount of research published on the IoUT and BMD is very limited. Of these limited publications, some cover the IoUT [3], [4], while others cover BMD [6], [34]. However, to the best of our knowledge, this paper is the first survey article that provides a comprehensive overview of the IoUT and BMD analytics relying on the most recent radical machine/deep learning approaches. This makes the present article beneficial for data scientists, ML engineers, data analyst, big data engineers, and policy makers in the marine-related disciplines.

To boldly and explicitly illustrate the contributions of this paper in the fields of IoUT and BMD analytics, Table I contrasts our unique contributions to other published treatises in the area. For all other articles in this table, a tick mark (✓) is granted, even if those dedicated only a few relevant sentences to the given subject. However, none of those surveys are similar to ours in terms of their coverage. Putting the depth of each discussion in other papers aside, there are many topics that are only covered in this paper on IoUT and BMD analytics.

The synthesis of research directions in this paper is handled in a smooth progression, starting from IoUT data collection, followed by networking, through to big data analytics and BMD processing. This flow of logic is illustrated in Fig. 1,

TABLE I
THE CONTRIBUTION OF THIS PAPER IN THE FIELD OF IOUT AND BMD ANALYTICS, COMPARED TO OTHER PREVIOUSLY PUBLISHED WORKS

Discussed Subjects	Domingo [4], 2012	Huang <i>et al.</i> [34], 2015	Kao <i>et al.</i> [3], 2017	Li <i>et al.</i> [6], 2018	Xu <i>et al.</i> [35], 2019	Jouhari <i>et al.</i> [36], 2019	Qiu <i>et al.</i> [37], 2020	Raj <i>et al.</i> [38], 2020	Khalil <i>et al.</i> [39], 2020	This Work
Internet of Underwater Things										
Sensors for Marine Data Collection	✓	✓	✓						✓	✓
Undersea Non-Destructive Testing										✓
Energy Consumption and Harvesting	✓	✓		✓	✓			✓	✓	✓
Communication Technologies	✓	✓	✓			✓	✓	✓	✓	✓
Link Reliability and Routing Improvement	✓	✓	✓	✓			✓	✓		✓
Network Architecture	✓	✓	✓	✓			✓	✓	✓	✓
Communication Protocols	✓	✓		✓	✓	✓	✓	✓		✓
Wired and Wireless Channel Modeling			✓	✓		✓				✓
Underwater Network Simulation Tools										✓
Security and Cryptography in IoUT	✓	✓	✓				✓	✓		✓
Software-Defined Networks (SDN)								✓		✓
Edge Computing in IoUT							✓			✓
Big Marine Data										
Data Acquisition, Aggregation, and Fusion	✓	✓					✓			✓
Sensor, Image, and Video Data Evaluation										✓
Open Access Databases					✓					✓
Distributed and Cloud-based Data Proc.		✓					✓			✓
Applications (Monitoring, Tracking, etc.)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Machine Learning and Deep Learning for BMD Analytics										
Machine Learning Techniques Briefing										✓
Sensor, Image, and Video Data Cleaning										✓
Feature Extraction from BMD				✓						✓
Hardware Platforms in BMD Analytics										✓
ML for Sensor, Image, and Video BMD	✓		✓	✓		✓				✓

which evolves from the distinct research directions to their amalgamation into big data analytics for IoUT. The four blue blocks labeled in this figure correspond to the remaining Sections II, III, IV, and V. Additionally, any text or symbol inside these blocks represents a dedicated subsection. Accordingly, the reader will gradually become familiar with state-of-the-art tools and techniques, whilst gaining an insight into the challenges and opportunities in the broad areas of IoUT, BMD, and processing BMD in IoUT.

This treatise is organized as illustrated in Fig. 2. Section II presents an overview of the IoUT by associating its ecosystem to the concepts and methodologies defined for IoT. In Section III, we discuss the usual challenges in the field of BMD and provide insights concerning oceanic sensors, image, and video data, which are widely available through several databases. In order to advance our knowledge and coordinate efforts in the field of BMD, sophisticated data analytic techniques and methodologies are required. As mentioned, one of the main approaches to meet this demand is to use

ML techniques. Section IV reviews several ML techniques conceived for automatic data leveraging from growing big marine databases. Finally, in Section V, the challenges and opportunities in the emerging fields of IoUT, BMD, and underwater data processing are discussed, whilst offering further insights into the opportunities and potential solutions to the challenges. The paper is concluded in Section VI.

II. INTERNET OF UNDERWATER THINGS

The concept of networks is broadly defined as a collection of independent machines, which exchange meaningful data through pre-arranged technologies (e.g. Ethernet, Wi-Fi, Bluetooth). Accordingly, the Internet (Worldwide Interconnected Networks) can be considered as a distributed network, or simply a network of networks. This network has an open standard and constitutes a widely accessible ecosystem with lots of users and a variety of applications. Within the Internet, IoT is the largest sub-ecosystem, which connects devices in any place on earth to the World Wide Web.

Similar to the definition of IoT, IoUT may also be defined as worldwide interconnected networks of digitally identified underwater objects, which all obey the communication protocols of a pre-specified reference model such as TCP/IP or OSI [1]. Based on this definition, a detailed discussion of the IoUT objects (e.g. sensors) and underwater communications will be provided. Additionally, the family of IoUT network standard models and protocols will be surveyed.

A. Underwater Sensors and Devices for Data Collection

Our knowledge of the underwater environment is rather limited. This is a consequence of having underdeveloped monitoring technologies for this environment. In addition, due to the large operational areas, sea and coastal monitoring tend to suffer from sparse sensor deployment [40]. To overcome these shortcomings, low-energy sensors that are capable of working in the vast, hostile, and dynamic underwater conditions are required.

As mentioned in the introduction section, the perception achieved by understanding the data collected using sensors in oceanic areas are essential both to human life and to environmental sustainability. These sensors, for instance, can evaluate the impact of human activities on resources in marine ecosystems and also make us aware of the amount of pollution dumped into the sea [41]. At the time of writing small-scale Underwater Wireless Sensor Networks (UWSN) and hydrographic research vessels that contain a variety of marine sensors are deployed locally to assess environmental pollution and also to evaluate the seawater quality. However, the main disadvantage of these UWSNs is their small coverage area, which cannot cover the seas in scales of thousands of square kilometers [42].

Accordingly, it is important to connect all the existing sensor networks to the Internet, giving birth to the IoUT, in order to create an infrastructure for monitoring marine life on a global scale. This internationally accessible IoUT that measures essential chemical and physical parameters at sea, provides both historical and real-time measurements from

myriads of marine locations worldwide. The collated oceanic sensor data will help experts predict future phenomena and also help policy makers ratify informed decisions [43].

Therefore, the IoUT infrastructure should consist of sensing objects and communication components in its underwater layers of the architectural model. These objects and components are known as nodes and sinks [3]. To elaborate [35],

- Underwater endpoint nodes are the end devices at the underwater side of the network, including various types of sensors, cameras, hydrophones, data storage micro-chips, actuators, acoustic tags, radio frequency tags, tag-readers, etc.
- Underwater mid-layer nodes are deployed above the underwater endpoint nodes, and are composed of data redistribution points, modems, gateways, repeaters, relays, etc.
- Sink nodes in IoUT terminology are the overwater nodes along with the land-side facilities, like buoys, exploration platforms, ships, satellites, onshore stations, etc.

Perhaps, the most important data collection components among all the different IoUT endpoint nodes are sensors, which not only collect data, but help activate other underwater components such as cameras, hydrophones, data storage micro-chips, and actuators. Some of the most popular environmental parameters in underwater applications are listed in Table II. Typical industrial sensors that offer accurate measurement of these parameters are also listed. The BMD generated from the continuous operation of these sensors is transferred, stored and processed in the IoUT ecosystem.

In Sections III and IV, further discussions on marine sensors, as well as on sensor data storing and processing, will be provided. In addition, a number of IoUT-based observatory systems will be introduced. Before any sensor can be deployed in an underwater environment, efficient and reliable underwater data communications should be realized. This poses one of the greatest challenges for the pervasive sensor deployment in IoUT ecosystems due to the extremely low acoustic and electromagnetic channel capacities and high signal attenuations over long maritime distances, which affect reliable underwater data communications. The next section will discuss and survey data transportation as well as communication methods and protocols of the IoUT.

B. IoUT Communications

Today's marine vehicular communication systems utilize the Very High Frequency (VHF) automatic identification system to provide essential shipping information [42] (e.g. vessel name, position, speed, destination, etc.). In addition, high-speed satellite communications are available as an expensive alternative to existing VHF systems. However, none of these systems are capable of supporting long-range underwater applications, where the acoustic waves are the dominant communication media. Nonetheless, these acoustic carriers also suffer from high propagation delay, fading, narrow-bandwidth, and high-attenuation. Therefore, in underwater applications, it is sometimes inevitable to use a combination of different technologies

(i.e. acoustic, electromagnetic, and optical) to overcome the communication challenges such as signal attenuation.

There is always a high level of signal attenuation, when passing through water. This attenuation affects every telecommunication technology in a different way. The signal attenuation is directly related to the main design constraints such as the maximum reliable data-rate and the maximum possible communications distance. In the following subsections, the signal attenuation is discussed in detail.

1) *Electromagnetic signal attenuation*: Electromagnetic waves and radio frequency signals do not propagate well underwater. This is mainly due to the high conductivity of seawater. The penetration depth of electromagnetic waves is inversely proportional both to the conductivity (σ [S/m]) and to the frequency (f [Hz]) [45]:

$$\delta \approx \frac{1}{\sqrt{\pi \mu f \sigma}}, \quad (1)$$

where μ is the water permeability and the penetration depth expressed in δ [m] is defined as the distance that an electromagnetic wave travels before becoming attenuated to e^{-1} of its initial amplitude. Based on this formula, only low frequency signals (with small channel capacity) can travel long distances in seawater, before they completely fade out. Therefore, the antenna size (L [m]) increases, as the frequency decreases [45]:

$$L \propto \frac{v}{f}, \quad (2)$$

where v is the speed of the electromagnetic wave in water that is almost equal to the speed of light in free space.

2) *Acoustic signal attenuation*: Due to the hostile electromagnetic underwater environment, most communications in the IoUT are based on acoustic links, which also suffer from a narrow frequency bandwidth. However, the attenuation of low frequency acoustic waves (α [dB/km]) is lower than that of electromagnetic waves [46]:

$$\alpha \approx F_1(f, pH) + F_2(f, T, S, z), \quad (3)$$

where f is the frequency in [KHz], pH is the water acidity (i.e. almost 6.0 ~ 8.5 for both freshwater and seawater), T is the water temperature in centigrade, S is the water salinity in parts-per-thousand (ppt), and z is depth in kilometer.

3) *Optical signal attenuation*: Another possible communication technology for underwater environments relies on an optical channel. However, similar to the previous pair of channel types, optical channels also suffer from signal attenuation.

Absorption and scattering are the two main causes of optical signal attenuation under water. Several previous studies have performed numerical simulations to estimate the attenuation [8], [47], [48]. These include solving the radiative transfer equation, which is time-consuming and complex, but precise; or using simplified models (e.g. Monte Carlo), which are typically fast but imprecise [8].

One of the most successful examples of these simplified Monte Carlo-based approaches was proposed by Gabriel *et al.* [47], where the spectral beam attenuation coefficient is calculated as,

$$c(\lambda) = a(\lambda) + b(\lambda), \quad (4)$$

TABLE II
IOUT SENSORS TO MEASURE UNDERWATER PHYSICAL , OPTICAL , FLUID , AND CHEMICAL PARAMETERS [9], [28], [35], [42], [44]

Environmental Parameters	Typical Underwater Sensor Products	Brief Description
Temperature ($^{\circ}C$)	• RBR Coda T by RBR • SBE series by Sea-Bird • T Xchange by AML Oceanographic	Digital (RS232) underwater temperature readings in the range of $-5 \sim 45^{\circ}C$ with an accuracy of $\pm 0.001^{\circ}C$ and up to 10,000 <i>m</i> depth rating
Conductivity (<i>S/m</i>) and Salinity (ppt)	• TTurb by TriOS • 4319 and 4419 by Aanderaa • C Xchange by AML Oceanographic • SBE 4 series by Sea-Bird	Digital (RS232 and LAN) conductivity measurement in the range of $0 \sim 200 mS/cm$ with an accuracy of more than $\pm 0.01 mS/cm$ and up to 6000 <i>m</i> depth rating (conductivity value is used in some products to indirectly derive salinity)
Depth (<i>m</i>) and Pressure (Bar) for Bathymetry	• PTM, PR36, and 2600 Series by Omni • 8000 series by Paroscientific • P Xchange by AML Oceanographic • LMK and LMP Series by BD Sensors • Nortek Scour Monitor by OSIL	Pressure-based Analog (current and voltage) and digital (RS232) sensors can measure water depth ($0 \sim 7000 m$), underwater absolute pressure ($0 \sim 700 Bar$), and underwater differential pressure ($0 \sim 6000 Bar$) with an accuracy of more than $\pm 0.1\%$. Some other acoustic sensors (e.g. Nortek Scour Monitor) can measure depth by using echo sounders.
Hydrophone ($dBV/\mu Pa$)	• Variety of products by Benthowave Instrument • DH-4 and TH-2 by Sonotronics • C and CR series by Cetacean Technology • TC series by etec electronic engineering • icListen and icTalk series by Ocean Sonics	Analog (voltage) and digital (LAN) $10 mHz \sim 2 MHz$ active and passive hydrophones with 1° to Omnidirectional beamwidth for applications up to 2000 <i>m</i> underwater and versatile beam shapes (e.g. Single and Multibeam) with $-230 \sim -110 dBV/\mu Pa$ voltage sensitivity
Turbidity and Visibility (NTU)	• OBS series by Campbell Scientific • InPro 8000 series by Mettler Toledo • WQ730 by xylem • 4112 by Aanderaa	Analog (current and voltage) and digital (RS232) $0 \sim 4000 NTU$ turbidity measurement (using forward scatter, side scatter light, and backscatter light) with an accuracy of $\pm 2\%$ in up to 6000 <i>m</i> depth rating
Optical Attenuation (Absorption) (m^{-1})	• FAS series by Sea-Bird • OLAS by Werne & Thiel	Scanning of the light spectrum between $400 \sim 730 nm$ with $\pm 0.01 m^{-1}$ accuracy in up to 5000 <i>m</i> depth rating with $10 \sim 25 cm$ light path length
Optical Backscatter or Volume Scattering ($sr^{-1}m^{-1}$)	• HydroScat series by HOBI Labs • Hyperion by Valeport • ECO series by Sea-Bird	Digitally (RS232) sensing of $0 \sim 5 sr^{-1}m^{-1}$ backscatter in $420 \sim 880 nm$ with $0.003 sr^{-1}m^{-1}$ scattering sensitivity and 140° nominal backscattering angle at up to 6000 <i>m</i> depth rating
Photosynthetic PAR ($\mu mol m^{-2} s^{-1}$)	• ECO and RXA series by Sea-Bird • LI-192 by LI-COR • RBR Coda PAR by RBR	Analog (voltage) and digital (USB) sensors in $389 \sim 700 nm$ spectral range for linear PAR measurement with an accuracy of $\pm 2\%$ in up to 7000 <i>m</i> depth rating
Spectral Irradiance (e.g. Fluorometers, Radiometers, etc.) (application-specific physical units)	• Cyclops series by Turner Designs • ECO, SeaOWL, and WETStar Fluorometers and Multispectral and HyperOCR Radiometers by Sea-Bird • Photo-, Fluoro-, and Radiometers by TriOS • AP Series by Aquaread	Continuous scanning of the entire ultraviolet, visible, and infrared spectrum between $190 \sim 950 nm$ with an accuracy of $\pm 0.2 nm$ in up to 6000 <i>m</i> depth rating for versatile applications such as chlorophyll, phycocyanin (Freshwater Algae), Phycoerythrin (Marine Algae), CDOM/FDOM, dye tracing (fluorescein, PTSA, and rhodamine), hydrocarbons (oil and fuel), nitrate, turbidity, wastewater monitoring, etc.
Water Flow and Current Velocity (<i>m/s</i>)	• Current meters by Valeport • Water flow meters by OTT HydroMet • DCPS and ZPulse DCS by Aanreraa • ISM series by HS Engineers	Analog (voltage) and digital (RS232) $0 \sim 15 m/s$ current flow sensors, using mechanical (impeller), magnetic induction, acoustic Doppler, and K-band radar Doppler technologies with an accuracy of $\pm 0.5\%$ and up to 6000 <i>m</i> depth rating
Tide and Wave Elevation (<i>m</i>) and Direction ($^{\circ}$)	• 4648, 5218, and 44xx series by Aanderaa • S500 by NexSens Technology • SVS-603 by Seaview Systems • ACM-PLUS series by Falmouth Scientific	Digital (RS232) tide height, wave height, and wave direction read with an accuracy of $\pm 0.5 cm$ in height and $\pm 2^{\circ}$ in direction and $\pm 1\%$ in period, plus $0 \sim 30^{\circ}$ wave tilt angle read with an accuracy of $\pm 0.5^{\circ}$ for versatile applications (e.g. energy period, steepness, irregularity, wave Fourier spectrum, moment spectral, tide pressure, etc.)
Nutrients (e.g. Nitrate (NO_2 , NO_3), Phosphates (PO_4), etc.) (ppm)	• AP Series by Aquaread • Ion-Selective Electrodes by Eureka Water Probes • SUNA and HydroCycle series by Sea-Bird • Marine Nutrient Analyzer by SubCtech	Nutrient sensors can be made by solid-state, liquid membrane, or UV spectrum technologies to measure $0 \sim 30,000 ppm$ nitrates with an accuracy of $\pm 2 \mu Mol/L$ in up to 2000 <i>m</i> depth rating and also to measure phosphates in up to 200 <i>m</i> depth rating. Other included ion-selective electrodes are ammonium, ammonia, chloride, fluoride, calcium, sodium, silica, etc.
Potential of Hydrogen (pH)	• pH Sensor by SubCtech • DP-HP5 by PreSens Precision Sensing • SP series by SensorLab • TpH and TpH-D sensors by TriOS	Analog (current and voltage) and digital (RS232 and USB) conventional and differential pH sensors in the range of $0 \sim 14 pH$ with an accuracy of more than $\pm 0.01 pH$ in up to 2000 <i>m</i> depth ratio
Dissolved Oxygen ($\mu Mol/L$)	• SBE and Hydro-CAT series by Sea-Bird • Oxygen Optode series by Aanderaa • AP Series by Aquaread • Oxygen sensors by AMT	Analog (current and voltage) and digital (RS232) dissolved oxygen sensor in the range of $0 \sim 1600 \mu Mol/L$, which is equal to 500% of surface saturation in all natural waters (fresh and salt) with an accuracy of $\pm 1\%$ and up to 12,000 <i>m</i> depth rating
CO_2 and pCO_2 (ppm)	• C-sense by Turner Designs • pCO_2 series by SubCtech • CO_2 series by Sunburst Sensors	Analog (current and voltage) and digital (RS232) CO_2 and pCO_2 sensors in the range of $0 \sim 10,000 ppm$ with an accuracy of $\pm 1\%$ in up to 6000 <i>m</i> depth rating

where $a(\lambda)$ and $b(\lambda)$ are the spectral absorption and spectral scattering coefficients, respectively, and $\lambda[m]$ is the wavelength. Both of these coefficients constitute intrinsic optical properties and are calculated by simple volume integration over a solid angle (Ψ).

Alternatively, the relationship between the received light intensity ($L(t, r, \theta, \varphi)$) and the transmitted optical power ($S(t)$) expressed in spherical coordinates (r, θ, φ) can be defined by the well-known radiative transfer differential equation [49]. One of the most accurate solvers for this differential equation in underwater optical communications was proposed by Illi *et al.* [48], who have formulated their time-domain approach as,

$$\left[\frac{1}{v} \frac{\partial}{\partial t} + \vec{n} \cdot \nabla \right] L = -c(\lambda) L + S + \iint VSF \times L \sin(\varphi') d\theta' d\varphi', \quad (5)$$

where v is the speed of light, $\vec{n}$ denotes the direction of propagation, and $c(\lambda)$ accounts for the same absorption and scattering coefficients of (4). The Volume Scattering Function (VSF) will be discussed in Section III-B2.

The improvement in solving (5) was achieved by enhancing the finite-difference method as well as by proposing a better approximation for the definite integral. Again, this is an accurate, but computationally expensive method of optical signal attenuation characterization.

The signal attenuation encountered by different telecommunication technologies discussed above directly affect the bit error rate and the overall underwater link reliability in the IoUT, which is discussed in the following section.

C. Improving IoUT Link Reliability

The underwater channel quality is significantly affected by several dynamic factors, including tidal-waves, pressure gradients, temperature gradients, floating sediments, and changes in water ingredients (chemical compounds). These channel dynamics affect the signal amplitude (distortion), frequency (dispersion), and speed (refraction) [50]. They also result in different delays, corresponding to delay jitter. Due to these challenges, all underwater channels are considered as unreliable links, which requires us to define a reliability metric and then, use this metric for quantifying and optimizing the quality of service in our IoUT network [51].

Previous contributions in this field tend to assess the link reliability either by software-based or hardware-based metrics [51]–[54]. Hardware-based metrics tend to measure Signal-to-Noise Ratio (SNR) and signal-to-interference ratio as their reliability metrics [52]. The signal strength values can be directly read from the hardware transceiver. On the other hand, software-based metrics predominantly rely on comparing the overall end-to-end data delivery of the communication systems [51]. The channel or link reliability metric is defined as the ratio of bits as well as packets that are delivered successfully through the link. This metric can be evaluated either at the bit-

or the packet-level. At the bit-level, the Bit Reception Ratio (BRR) is the reciprocal of the Bit Error Rate (BER):

$$BER = \frac{N_{Errorous\ Receptions} [bits]}{N_{Total\ Transmissions} [bits]}, \quad (6)$$

$$BRR = 1 - BER. \quad (7)$$

By contrast, for a packet of m bytes (i.e. $8 \times m$ bits), successful reception of a packet means that all the m bytes were received correctly. Thus, the link reliability metric at the packet-level can be defined by the Packet Reception Ratio (PRR) as [51]:

$$PRR = (1 - BER)^{8m}. \quad (8)$$

Another commonly used alternative to the BER of (6) for defining a link reliability metric is the Required Number of Packet transmissions (RNP), defined as [53]:

$$RNP = \frac{N_{Total\ Transmissions} [packets]}{N_{Correct\ Receptions} [packets]}. \quad (9)$$

In contrast to the BER that is measured at the receiver side, the RNP metric is designed to be measured at the transmitter side.

Another software-based link reliability metric, namely the Expected Transmission Count (ETX), is defined as the number of expected transmissions that a node requires for successful delivery of a packet [54], which can be directly calculated as the reciprocal of the PRR value of:

$$ETX = \frac{1}{PRR}. \quad (10)$$

Based on the band-limited nature of both the overwater and underwater telecommunication channels, the BER, RNP, and ETX values that have been respectively calculated in (6), (9), and (10) are related to the originally transmitted data-rate, initially transmitted power, as well as to the distance between the consecutive transceiver pairs [45]. In this regard, reliable communication in the presence of random background noise, requires a certain minimum received power. This guarantees reliable data flow throughout the IoUT infrastructure for the ensuring big marine data processing.

The data-rate in bits-per-second (bps) and the transmission range, alongside the other characteristics of communication technologies are shown in Table III. The simplified simulation plot of attenuation vs. transmission range in this table is calculated using the same conditions as in the studies discussed in Section II-B. Here, the electromagnetic conductivity of the seawater and freshwater are considered to be 44,000 and $100 \mu S/cm$, respectively [55]. The salinity values of the seawater and freshwater are also considered to be 35 and 0.4 ppt. The water depth only has a minor effect on the results and it is considered to be 1 km and 1 m for seawater and freshwater, respectively. The water temperature and its pH value are assumed to be $10^\circ C$ and 7.7.

By comparing the details provided in Table III, one can readily conclude that reliable underwater wireless communications are restricted to low data-rate acoustic waves for long

TABLE III
DATA-RATES AND OTHER CHARACTERISTICS OF COMMUNICATION TECHNOLOGIES IN UNDERWATER APPLICATIONS [8], [36], [56]

Communication Technology	Data Rate	Latency	Advantages Disadvantages	Simplified Simulation of Attenuation vs. Transmission Range
	Tx Range	Efficiency		
	Attenuation	Main Channel Parameters		
Acoustic	~ kbps	High	- Proven and widely used technology - Long communication distances - Works in non-line-of-sight - Low data rate and bandwidth - Harmful on marine life - Large latencies	
	~ Kilometers	~ 100 bits/Joules		
	< 150 dB/km	- Temperature - Salinity - Pressure		
Electromagnetic	~ Mbps	Low	- Works in non-line-of-sight - Very limited range	
	~ Tens of meters	N.A.		
	< 100 dB/m	- Conductivity - Permittivity		
Optical	~ Gbps	Low	- Ultra-high data rate and bandwidth - Low cost - Highly affected by environmental parameters - Requires a line-of-sight - Limited range	
	~ Hundreds of meters	~ 30,000 bits/Joules		
	< 15 dB/m	- Absorption - Scattering (Turbidity) - Organic Matters		

distances or high data-rate optical rays for short distances. Short-distance and low data-rate electromagnetic waves are substantially outperformed by the other two technologies.

Despite the restrictive nature of the above-mentioned underwater communication technologies, innovative techniques can be developed to boost both the software-based and the hardware-based reliability metrics of underwater communication links. These techniques include *ad hoc* routing improvements and hop-count optimization, which are discussed in the following subsections.

1) *Routing improvement*: The specific choice of the link reliability metrics to be optimized has a substantial influence on the routing design of *ad hoc* networks [57]. Therefore, the design and implementation of a reliable network under the above-mentioned restrictive conditions of underwater channels requires an efficient data routing scheme [40]. One such scheme has been proposed by Rani *et al.* [58] for UWSN, which can be adapted to IoUT nodes as well. As shown in Fig. 3, the whole underwater wireless network in [58] is divided into multiple sub-regions or clusters. In every cluster, relay nodes cooperate with the local normal nodes to forward data to the cluster-heads. The cluster-heads of each cluster are responsible for routing and transmission of data to the next

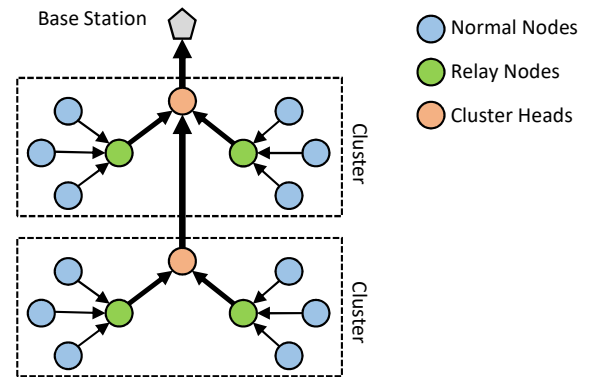


Fig. 3. Reliable and energy-efficient multi-cluster network topology [58].

cluster-head situated in the upper sub-region, termed as *cluster coordinator*.

The scheme presented in [58] relies on a pair of efficient algorithms. The first algorithm constitutes a location-free and energy-based policy that is devised to group individual nodes into cluster sets, aiming to increase SNR reliability metric. The second algorithm involves allowing cluster-heads and

relay nodes to store data before routing and transmission. This helps the whole system to increase the RNP metric in (9), by avoiding retransmission of the same data packets. By implementing these ideas, the protocol tends to transmit fewer packets of data with a higher reliability and also carries out load balancing in packet routing throughout the network.

Another noteworthy data routing scheme was proposed by Tran-Dang *et al.* [59] for underwater acoustic sensor networks. Their work follows the same structure as [58] in Fig. 3. By contrast, their cooperative routing algorithm does not divide the UWSN into multiple clusters. Instead, every normal node independently selects its own relay node and cluster-head. This selection is based on the SNR link reliability metric of Section II-C, on the physical distances represented by hop-count in Section II-C2, and the time of arrival, which is extracted from the timestamps of the packages. Moreover, this algorithm constantly monitors the environment in UWSN to estimate the SNR, hop-count, and time of arrival throughout the network. These parameters are frequently updated to adapt to the dynamic underwater environment. However, the simulation results reported in this paper require experimental fieldwork for validation.

More discussions on the effects of network topology on the software-based link reliability metrics (e.g. BER) are provided in Section II-F. In the next section, a hardware-based reliability metric (i.e. SNR) will be used for the evaluation and optimization of the hop-counts in IoUT.

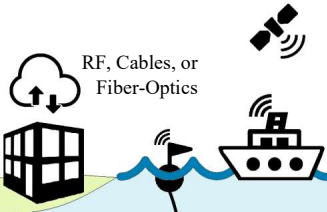
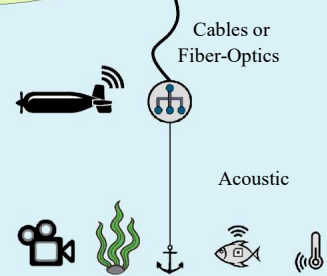
2) *Hop-count optimization*: Another subcategory of methods capable of increasing the underwater link reliability is to optimize the hop-count, which is defined as the number of intermediate hardware devices (Modem, Gateway, Switch, Router, HUB, and Repeater) conveying the data between successive sources and destinations [60]. For example we could minimize the number of hops, provided that the transceiver is capable of reliably communicating over higher distances with the aid of higher transmit power and or more sophisticated receivers.

To characterize the effect of the number of hops on the system's reliability, we have to consider the fact that almost all of the wireless underwater endpoint and mid-layer nodes are battery-limited and the adequate operation of the entire network depends on the charge of their batteries. On the other hand, increasing the transmission distance, which is required for most underwater applications, decreases the effective bandwidth. This in turn increases the power consumption of delivering the payload at a minimum SNR.

To address the challenges imposed by battery constrained underwater nodes on the link's SNR reliability metric, Li *et al.* [6] suggest reducing the hop-distance by deploying relay nodes along the underwater link to improve the overall transmission performance. This relay-aided transmission scheme is based on power-bandwidth-range dependency, and at the time of writing it offers an energy-efficient method supporting sustainable high data-rate delivery in underwater scenarios.

To establish reliable communication in any IoUT system, a suitable network architecture should be developed. This architecture has to define the model layers, connection topologies, communication protocols, and related equipment sets. All

TABLE IV
MATCHING IOUT ARCHITECTURES TO THE 5-LAYER TCP/IP MODEL [61]

IoUT	TCP/IP	Schematic	Equipment
Overwater Layers	Application, Transport, Network, Data Link, and Physical		Endpoint Computers, Routers, Switches, Modems, HUBs, and Repeaters
Underwater Layers	Physical		Wired or Wireless
	Data Link and Network		Repeaters, HUBs, and Modems
	Transport and Application		Gateways, Endpoint Sensors, and Actuators

these establishments have a direct influence on our underwater communication paradigm and its subsequent link reliability metrics. This architecture and its influence on all network aspects will be discussed in the next section.

D. IoUT Network Architecture

Despite the fact that every standalone network can have its own architecture, the Transmission Control Protocol/Internet Protocol (TCP/IP) is the most ubiquitous network architecture. This reference architecture consists of 5 distinct layers, namely the application, transport, network, data link, and physical layers [61]. The network layer is also known as the internetwork or Internet layer in some literature. Similarly, the data link and physical layers are sometimes combined for forming a 4-layer TCP/IP model, in which the combined data link and physical layer is referred to as the network access or network interface layer [62].

The TCP/IP layers can be arranged as shown in Table IV to be matched to the IoUT network architecture. This table presents the IoUT network architecture in a very general form. Here, both the land-side computers and the underwater sensors are considered as network endpoints, located at the highest TCP/IP layer (i.e. the application layer). This architecture covers the entire network, from the underwater application layer up to the overwater application layer. The four rows in this table, from top to bottom, represent the sink nodes, physical channel, underwater mid-layer nodes, and underwater endpoint nodes, based on the definition of those nodes in Section II-A. It should be noted that the standard TCP/IP layer numbers (i.e. 1 for physical, 2 for data link, 3 for network, 4 for transport, and 5 for application) are not used anywhere in this paper. Instead, the layer names are used to specify the corresponding TCP/IP layers.

Every TCP/IP layer shown in Table IV relies on some protocols to govern the communications between the IoUT devices, regardless of their underlying structure and design.

Based on these protocols, every single entity in the network will be aware of the data formats, communication syntax, synchronization methods, security concerns, and error control schemes utilized. A layer-wised collection of these protocols is referred to in parlance as *protocol stack*. Similar to the IoT [32], the IoUT ecosystem does not require a universal protocol stack. It can rely on multiple protocols that co-exist in the model layers of a single project, as a result of diverse demands and requirements. These layer-wise stack of protocols will be discussed in the following subsections, along with other layer-wise security protocols in Section II-H2. These protocols are required for supporting different aspects of an IoUT infrastructure to meet various challenges, including [63]:

- Connecting to the existing IoT and WEB protocols;
- Providing an acceptable latency;
- Harmonizing the packet lengths with the channel's coherence times;
- Overcoming the significant signal attenuations;
- Matching the bit error rate to the application;
- Accommodating the low underwater bandwidth;
- Resisting the everyday security threats.

In the following subsections, the layers shown in Table IV along with their relevant protocols are discussed, starting from the application and transport layers.

1) *Underwater application layer*: The underwater application layer in IoUT is mainly responsible for identifying each individual object (i.e. sensor's id, sensor type, sensor location, etc.) and then, gathering data, processing information, and delivering commands. Gathering data in application layer includes sensing, tracking, recording, and streaming of (live) data, which will be discussed in Section III. Data processing on the other hand, will take the gathered data and process them, which results in the information sought. This topic will be discussed in more detail in Section IV. Some actuators may also belong to this layer to react to the environment as instructed by the commands of scientists, visitors, policy-makers, educators, owners, or even from a ML model. The action of these actuators is recordable and controllable through the IoUT.

The existing IoT ecosystem is well equipped with application layer protocols, covering all technological demands in the diverse IoT applications. These protocols are comprehensively reviewed by Al-Fuqaha *et al.* [1]. By studying these IoT protocols it is readily seen that not all of them are suitable for IoUT applications. Some of these protocols rely on wideband operation, while others have protocol headers that are redundant in underwater communications, without any significant effect on the link reliability.

For example, while the HyperText Transfer Protocol (HTTP) is widely used in the application layer of the worldwide web to deliver web-pages, its internal separation into header section and body section is not necessary in a simple sensor data transmission in IoUT. By this means, some suitable protocols to be used in the underwater application layer include but are not limited to the Inter-Module Communication (IMC), Serial Line Interface Protocol (SLIP), eXtensible

Markup Language (XML), Message Queuing Telemetry Transport (MQTT), Data Distribution Service (DDS), and Modbus [1]. Another protocol is the Teletype Network (Telnet), which is considered as the origin of modern application layer protocols [64]. The simple 8-bit and text-oriented data structure of Telnet makes it suitable for IoUT applications.

2) *Underwater transport layer*: As already mentioned in Section II-A, the sensors, cameras, and actuators in the application layer are typically considered as underwater endpoint nodes, which offer a specific service in IoUT. Considering the nature of data produced by any specific service, the transport layer will be responsible for splitting the data into packets prior to its transmission through the network gateway. This layer also takes into account the importance of data order and any potential data loss.

In this regard, TCP is a reliable transport layer protocol that has widely been used in the Internet. In contrast to the concept of a *reliable link*, which has been studied in Section II-C, a *reliable protocol* (e.g. TCP) is defined as a protocol with handshaking, error detection, and error correction capabilities. Based on this definition, reliable protocols typically bear more overhead than unreliable protocols, and as a result, grow in the required channel capacity. This is not a desirable feature in IoUT, where a few lost sensor data packets can be acceptable, in return of freeing up some channel resources. Therefore, User Datagram Protocol (UDP) is more recommended in underwater applications, as it does not use acknowledgments at all.

In addition to the TCP and UDP, the Datagram Congestion Control Protocol (DCCP) is another transport layer protocol, which is message-oriented and reliable. DCCP offers congestion notification and congestion control that is useful in UWSNs.

Finally, nodes and gateways in the transport layer may have wired or wireless modes. In the case of wireless modes, the undersea nodes should be as multi-modal as possible for considering the node size, weight, complexity, price, and energy consumption [6]. Relying on multi-mode operation, access to the IoUT networks is possible by relying on multiple marine telecommunication technologies (i.e. acoustic, electro-magnetic, and optical).

3) *Underwater network layer*: The major tasks of the network layer include bi-directional data packet handling between individual endpoints as well as protocol translation between adjacent layers (i.e. transport and data link). These responsibilities are addressed in the network layer by implementing both the Internet protocol and the data routing.

The Internet Protocol version 4 (IPv4) and Internet Protocol version 6 (IPv6) are the main network layer Internet protocols. However, IPv6 is not recommended in IoUT, as it is more verbose than IPv4 owing to its excessive header. Additionally, if DCCP is used in the previous transport layer, using Explicit Congestion Notification (ECN) protocol in the network layer is recommended to allow end-to-end notification of network congestion, without dropping packets.

Alongside the Internet protocols, there are network layer routing protocols that specify how the routers communicate and collaborate with each other. An efficient data routing

scheme is an essential part of any UWSN. In addition to the discussed algorithms and protocols in Section II-C1, some other well-designed routing protocols for underwater communications are E-CARP [40] and E-CBCCP [58].

The IP and routing protocols in the network layer, along with the scheduling protocols in the next section, provide the access of the sink nodes (e.g. surface-floated buoys, ships, satellites, etc.) to the endpoint nodes (e.g. underwater sensors), through the mid-layer nodes [3]. These mid-layer nodes are not always required, but in the case of long-distance communication and/or in the presence of high signal attenuation, they play an essential role in extending the range.

Given the natural difficulties involved in recharging those endpoint nodes, mid-layer nodes, and sink nodes, underwater networks should have an efficient energy management strategy. Many previous studies have investigated different techniques of minimizing the energy consumption in underwater networks [11], [58], [65], [66].

A plausible power conservation strategy was introduced by Koseoglu *et al.* [66] for underwater networks. This strategy assigns more network-layer resources (i.e. route accesses and scheduling) to the nodes and sinks that have to transmit over a longer distance and/or have less efficient physical layers. By involving cross-layer optimization of the channel access rate instead of the separate optimization of the layers, a 66% less energy consumption per successfully transmitted bit could be achieved. However, this study only considered a single-hop scenario, where the nodes do not perform multi-hopping. Whilst the holistic optimization has to resolve the design-dilemma of having more short hops at a higher delay or less hops at a higher power. Therefore, this solution needs further investigation of multi-hopping and to strike a trade-off in terms of the above dilemma.

4) *Underwater data link layer*: As already mentioned in the previous section, data link layer supports the access of underwater sensors to the surface stations by appropriately sending/receiving data from the physical layer, as well as coordinating the data departures. This data exchange with the physical medium requires a conversion between data frames and electronic signals, depending on the underlying communication technology (i.e. acoustic, electromagnetic, and optical).

The nodes and sinks that use any of these technologies, usually share the physical telecommunication channel, and therefore, scheduled access to the shared channel becomes a mandatory action in every UWSN [65]. This necessity gives rise to the concept of Media Access Control (MAC) in the data link layer. The scheduled access, which is provided by MAC helps to avoid or manage data collision and to complete reliable data transmission in real-time. Access regulation in MAC becomes even more important, when connecting the existing UWSNs to the IoUT infrastructure.

In this regard, MAC scheduling protocol can be either collision-free [67] or contention-based [68]. Due to the unique characteristics of underwater acoustic channels, employing contention-based MAC protocols in IoUT are inefficient and costly. In contrast, collision-free MAC protocols achieve higher performance by consuming lower energy and offer-

ing better network throughput [67]. Time-Division Multiple Access (TDMA), which is also known as a slotted MAC is the mainstream collision-free scheduling protocol in most of the UWSN standards [69]. Based on the network topology, as discussed in Section II-F, in the slotted MAC, a cluster-head undertakes the responsibility of data frame coordination. On the other hand, using ALOHA as a contention-based MAC protocol is also common in UWSN applications [66], which requires no coordinator node at all.

Additionally, due to the peculiar features of underwater communication channels (as will be discussed in Section II-D5), conventional MAC designs and protocols cannot work well under water [70]. Hence, in UWSN, designers should minimize the MAC and Ethernet frame size, while satisfying all other protocol requirements. Accordingly, research has been conducted to propose compliant underwater MAC protocols including TRMAC [60], UWOR-MAC [71] and EAST [65].

In addition to shortening the MAC frame size in the light of the limited underwater bandwidth, we also have to consider limited underwater energy resources. The problem of providing a scheduling service for sensory data transmission while conserving energy is addressed in [65] by designing an energy-aware scheduling protocol, which is a compelling solution that offers a reasonable network throughput at the time of writing. However, this method has a complex probability-based model, which relies on time-consuming initialization. As a further constraint, the probabilistic state transition mechanism has to be regularly updated. Nonetheless, these adjustments can be carried out automatically.

Among all the underwater communication challenges, the lack of continuous network connectivity between nodes and sinks is the main reason of the notorious unreliability of links. In computer-networking terminology, communicating links associated with intermittent connectivity are termed as *delay tolerant networks* [42]. To address this, Li *et al.* [72] introduce a delay tolerant MAC protocol for underwater wireless networks relying on m -fold repeated transmission of every packet. The value of m is optimized with respect to the successful transmission probability determined by the channel's reliability.

Another important IoUT data link layer protocol termed as *Dolphin* is introduced by Fujihashi *et al.* [73], which supports high-quality video transmission, based on the video-on-demand streaming model. This protocol transmits multi-view videos using acoustic technology. To handle multiview video transmission in low data-rate acoustic channels, the underwater systems are categorized into quality-sensitive and delay-sensitive applications, which are supported by the so-called S-Dolphin or G-Dolphin mode of operations. Both of these methodologies exploit time-shifted slot assignments between the underwater video communication pairs to avoid data collisions, albeit they have not been implemented in practice.

5) *Physical layer in IoUT*: The nature of the underwater physical layer can be completely different in the case of wired or wireless communications. In a wired communication regime, we can use either cables or optical fibers. The use

Related to Ambient	Suspended Sediments	Related to Signal Path	High Attenuation
	Corrosions and Erosions		Fading and Reverberation
	Nodes and Sinks Mobility		Long Propagation Delay
	Lack of Energy Resources		Variable Delay (Jitter)
Related to Noise	Thermal and Impulse	Related to Frequency	Small Channel Capacity
	Cross Talk		Large Antenna Size
	Intermodulation		Distortion and Dispersion

Fig. 4. Major signal impairment causes in the underwater physical layer.

of cables is limited to wired energy transfer [9] as well as narrow-bandwidth data transmission in underwater wired sensor networks [74]. On the other hand, the use cases of optical fibers in underwater physical layer communications include but are not limited to:

- *Wideband image and video transmission:* As it will be discussed in Section III-B1, there are many open access observatories worldwide, which provide live image and video data using their IoUT infrastructure. Some of these observatories (e.g. [OBSEA](#), [MARS](#), and [ONC](#)) rely on wired data transmission through fiber-optics.
- *Telephone and Internet delivery worldwide:* In 1988, TAT-8, the first transatlantic fiber-optic cable, was laid between United States, Britain, and France [75]. Nowadays, we rely almost entirely on 550,000 miles of underwater optical fibers for inter-continental Internet transactions, as they are faster and cheaper than satellite communications.

Despite the benefits of using wired communications in IoUT, they have some major drawbacks, resulting in avoiding them as much as possible. Some of these disadvantages include slow and tedious installation, higher implementation and maintenance costs, hard to locate and repair faults, less flexibility as a result of fixed mechanical junctions, being subject to rotate and twist failures in high underwater nodes mobility, lack of security in either case of intentional damage or unauthorized access, being subject to physical and chemical environmental impairments, being subject to shark bite related damages, etc. [76].

To avoid the above-mentioned disadvantages of wires in the underwater physical layer, wireless communications offer a promising solution. However, underwater wireless signal transmission and wave propagation is more challenging than its aerial/terrestrial counterparts. This is mainly due to the harsh environmental conditions at deep-sea ($> 100\text{ m}$), shallow water ($< 100\text{ m}$), or even freshwater. Some of the phenomena imposing signal impairments on the underwater physical layer are listed in Fig. 4.

Each and every challenge shown in Fig. 4 has a different impact on the different underwater applications. For

instance, wireless data communication in shallow water is mainly affected by surface noise and multipath fading. By contrast, in the deep-water applications, the attenuation, the narrow-bandwidth, and the high sensitivity to environmental parameters such as temperature are responsible for the most grave signal impairments.

Dut to these challenges, the existing overwater wireless protocols are not readily suitable for undersea applications. Some of these overwater protocols belong to the [IEEE 802](#) family and to their derivatives (e.g. [Zigbee](#)) as well as to the Low-Power Wide Area Networking (LPWAN) protocols (e.g. Long Range ([LoRa](#)) and Narrowband IoT ([NB-IoT](#))). These protocols normally use electromagnetic waves as their transmission medium. Hence, they will have a very short communication range in underwater environments. Accordingly, the use cases of these electromagnetic wave based protocols would be limited to a few specific scenarios in IoUT, like:

- *Short-range underwater transmission:* As depicted in the right figure in Table III, electromagnetic waves can only propagate for tens of centimeters in seawater and for couple of meters in freshwater. This propagation distance increases, if the carrier frequency is reduced. Therefore, whenever no long-distance transmission is required, the aforementioned protocols can still be employed in IoUT solutions [77].
- *Long-range overwater transmission:* [LoRa gateway buoys](#) and [Underwater NB-IoT](#) solutions of the sea mammal research unit rely on LPWAN for their overwater data submission. Another potential solution is to construct an *ad hoc* point-to-point link, based on the ITU-R M.1842-1 recommendation. This approach uses the VHF radio solution of the Automatic Identification System (AIS) for conveying data packets [42].

It is worth noting that wired protocols - such as the IEEE 802.3 for LAN (Ethernet) - can still be employed in IoUT projects after modest reconfiguration [78].

Nevertheless, if long-distance underwater wireless communications are required, acoustic waves constitute the only viable option. An efficient underwater acoustic physical layer IoUT protocol has been conceived by Marchetti and Reggiani [79]. They achieved a beneficial signal-to-noise ratio gain, by taking advantage of the energy gleaned from all propagation paths, instead of only the strongest path.

As will be discussed in Section II-G, to overcome the challenges of wireless communications portrayed in Fig. 4, we basically need to undertake an analytical or numerical approach, to computationally evaluate data transmission throughout the network. Taking this mathematical approach will consequently empower us to optimize our communication system in the favor of having more reliable IoUT infrastructure. However, analyzing and optimizing the physical layer of the network requires us to provide the simulation tool with an appropriate underwater physical channel model, which is the topic of the next section.

E. Underwater Channel Modeling

As discussed in the previous sections, the underwater physical layer exhibits different behavior in response to both differ-

ent propagation modes and different channel types. For example, the signal attenuation was found out in Section II-B to be calculated differently for electromagnetic, acoustic, and optical carriers. The physical layer investigated in Section II-D5 also exhibits different behavior in the case of wired and wireless channels. Regardless of the channel types and propagation technologies, we require to have an appropriate channel model to have a better insight on underwater data transmission. This proper model can then be used for predicting the performance of our communication system, designing the optimum underwater location of nodes, and decreasing the overall energy consumption of the system, before its actual deployment [80].

The channel models of wired and wireless networks vary with the choice of the communication technology. In other words, it is not possible to design an accurate channel model for universal employment in every application [50]. On the other hand, a feasible model has to undergo some degrees of simplifications, which is strongly correlated with the requirements of the problem itself [81]. As an example, we might simply neglect the water salinity in an acoustic channel model, while this simplification is not possible in underwater electromagnetic propagation.

Considering the aforementioned factors, channel models are somehow tailored to their own specific use cases. Accordingly, one usually needs to modify and combine the main features of multiple stand-alone models to come up with a dedicated solution to a specific application. For instance, each of the following works considered a specific aspect of the acoustic wave propagation underwater (i.e. layered ocean water, particles in seawater, and the slope in seabed) and neglecting the rest, for simplification.

- It is very common in underwater channel modeling to consider a constant phase velocity throughout the entire medium (isovelocity). But in the contribution of Naderi *et al.* [82], the non-isovelocity acoustic wave propagation in shallow-water environments has been evaluated. They split ocean water into multiple layers of piece-wise linear sound speed profiles. Afterward, they use traditional sound wave propagation techniques to extract the time-variant channel transfer function. The complexity level of their solution can be adjusted by increasing the number of the linear layers. This could easily result in instability or inaccuracy in case of extra-thin or extra-thick layers, respectively.
- Relatively long wavelength of acoustic waves compared to the floating particles in ocean, makes them less vulnerable to backscatter and forward scatter phenomena. This is not the same for optical waves, as will be studied in Section III-B2. In a relevant study, Zhou *et al.* [81] have considered the impact of the scattering particles on the propagation characteristics of acoustic waves in underwater environments. They randomly distribute those particles on an assumptive rectangular cross section of ocean. Despite their interesting method, two main drawbacks exist in their statistical approach. First, their evaluation is only a 2D vertical cross section of seawater, neglecting realistic 3D sections. Second, they do not consider the stochastic size of the scattering particles.

- It is generally considered safe to assume the sea surface as a flat plane in underwater acoustic wave propagation, but this is not the case for the seafloor. In the work done by Naderi *et al.* [83], the up/down slopes in seabed have been taken into account. This important consideration is particularly essential in the case of shallow waters in the presence of coral reefs and plants, and cannot be neglected.

As mentioned, in order to have a desired customized and simplified underwater channel model, relevant existing models can be combined and unified. The majority of acoustic channel models (including all of the above-mentioned) follow the same mathematical approach. This approach relies on the superposition of a single Euler wave ray, over all the possible propagation paths in the time- or frequency-domain. This superposition relying on resolving the time-variant channel impulse responses can be written as [82],

$$h(t) = \sum_n c_n e^{-j(2\pi f_0 t + \theta_n)} \delta(t - \Delta t_n), \quad (11)$$

$$\theta_n = \frac{2\pi f_0}{v_0} \Delta L_n, \quad (12)$$

where f_0 and v_0 denote the frequency and the acoustic wave speed in water, Δt_n and ΔL_n stand for the propagation delay and the difference in propagation path length, θ_n is the propagation phase shift, and $\delta(\cdot)$ is the Dirac delta function. Here in (11), c_n encapsulates both the initial gain and the subsequent losses by incorporating the concept of statistical random variables to model the stochastic environmental parameters (i.e. effects of scattering particles, motion-induced Doppler shifting, location uncertainty of nodes, changes in received power, dynamic seabed topology, and other natural variables). Selection of these environmental parameters for encapsulating in c_n differs from one use case to another, depending on the scale of these parameters' effect in every case [80].

The same concept of acoustic superposition in (11) can be used to model electromagnetic channels. But as discussed in Section II-B, high-attenuating electromagnetic waves are not common in underwater data transmission and therefore, they are not evaluated here. In a clear contrast, using line-of-sight optical communications as well as fiber-optic channels (as discussed in Section II-D5) are very popular in high data-rate underwater communications [8]. In this regard, the subject of optical fibers is out of the scope of this article and the reader is encouraged to follow it up from other resources [84]. Meanwhile, studying the line-of-sight optical channel properties in underwater applications will be conducted in Section III-B2.

As stated at the beginning of this section, the channel modeling is the backbone of every IoUT network simulation to evaluate its proper operation and to optimize its parameters [85]–[87]. One of the very basic parameters in IoUT networks that relies on channel modeling is the node location and it is referred to in parlance as *network topology*, which will be discussed in the next section. Later on, in Section II-G we will review software tools that offer wires and wireless underwater

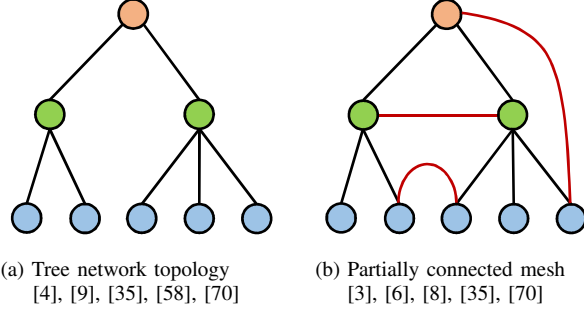


Fig. 5. Reported network topologies in IoUT applications.

channel simulation, along with network topology design and optimization.

F. IoUT Network Topologies

By utilizing TCP/IP as the reference model for the IoUT network architecture, almost all of the known network topologies can be used in gateway-enabled underwater applications [35], [70]. Two of these topologies (i.e. tree and mesh), which have high potential and are suitable for implementation in IoUT are shown in Fig. 5.

As discussed previously in Section II-D5, the frequency bandwidth is limited undersea and energy is hard to harvest. Therefore, advanced distributed topologies are rarely used in underwater applications, instead, the conventional tree and mesh topologies are dominant. The tree topology is typically used in small networks relying on one-way protocols [4], [9], [58], [70]. The tier (client-server) negotiation in this topology is based on a request and response process. On the other hand, the mesh topology tends to be the option of choice in sophisticated networks to meet the high traffic requirements, while using all available signal routes and frequency bands efficiently. This ensures that the limited channel capacity of underwater environments is efficiently exploited [3], [6], [8], [70].

It is worth noting that any signal route of Fig. 5 has its own reliability quantified in terms of its overall bit error rate in any typical IoUT network. This assists us in beneficially choosing a suitable network topology for any application, given its specific bit error rate target.

However, choosing a suitable network topology, designing a multi-layer communication network, and selecting an appropriate protocol for each layer of its architecture is not always a straightforward task. This job requires a tedious work to continuously design, test, debug, integrate, and deploy the newer versions of the network. It will get even more challenging in the harsh underwater conditions. To make this process a lot easier, some simulation tools are available, which can be used in underwater network design as well as layer-wise protocol testing. These software tools will be studied in the next section.

G. Underwater Network Simulation Tools

In IoUT applications, an improperly configured network can cost a lot of time and money [88]. The best way to avoid these

costs is to simulate and test every aspect of the network, prior to its first deployment and to any potential re-configuration. To do so, a wide range of open-source and commercial simulation tools are available. Table V lists a number of the tools that have the capability to model underwater channels and protocols.

All of the tools listed here are discrete-event network simulators, which model a system as a sequence of event triggers, assuming that no changes occur between consecutive events. This highly-repeatable simulation method is easier than analytical models to implement [89]. The level of network support in these tools varies from wired and wireless *ad hoc* networks to more advanced cellular and satellite communications.

The first item in this table is the well-known Network Simulator (ns) software and its ns-1, ns-2, ns2-Miracle, and ns-3 upgrades, which have been used by more than 600 academic as well as industrial institutions in more than 50 countries worldwide [90]. Some other tools that rely on ns for their underwater network simulations have also been studied in this table (i.e. AQUASIM, SUNSET, WOSS, and DESERT Underwater). They are all equipped with underwater channel modeling, acoustic transmission, and relevant protocols.

Based on their license level (i.e. commercial, research, education, or military), the simulation tools in Table V can undertake a verity of analysis and design tasks, including [91]:

- Channel modeling;
- Layer-wise protocol design and testing;
- Data transmission, traffic, and routing;
- Spatially arranging the network topology;
- Simulating network elements and commercial devices;
- Working with heterogeneous network technologies;
- Analyzing the overall network performance.

Additionally, with two exceptions for AQUASIM and OM-Net++, all other software products in this table can perform IoUT network emulation, in conjunction with available hardware as well as software solutions. In a contrast to network simulators that are an abstract mathematical description of a communication system, network emulators are virtual networks that mimic the behavior and also mirror the functionality of a real network, to which the end-systems will physically connect [92].

Network emulation provides a more realistic characterization of the underlying network. Therefore, it is highly recommended in IoUT applications to continue with an emulation step, right after network simulation and just before their actual deployment. By undertaking this simulation middle-step we can assure the quality of the IoUT network, by testing and troubleshooting it in a close to real-life environment.

It is worth mentioning that the accuracy of the simulation tools discussed in this section, as well as of the numerical channel models in Section II-E depend on the level of details provided in their inputs. In addition, relying on mathematical estimations, approximations, and statistics to model various aspects of a given natural environment, usually results in limited accuracy. For example, the number of natural parameters that can affect a typical underwater acoustic or optical channel can be huge. Some of these unpredictable parameters include noise, distortion, uncertain electronic specifications,

TABLE V
SIMULATION TOOLS FOR ANALYZING AND DESIGNING UNDERWATER COMMUNICATION NETWORKS AND PROTOCOLS

Software Name	Owner	Initial Release	Based On	Operating System(s)	Language(s)	Main Application(s)	Real-Time	Emulation	Open-Source	Other Specifications
Network Simulator (ns)	U.S. National Science Foundation	1990s	—	Windows, Linux	TcL, C++	Research and Education	✓ — ✗	✓	✓	- It has had four releases at the time of writing (i.e. ns-1, ns-2, ns2-Miracle, and ns-3). - All releases are object-oriented discrete-event simulators. - ns products are reliable basis for many other open-source products (e.g. AQUASIM, SUNSET, WOSS, DESERT Underwater, etc.).
Aquatic Simulator (AQUASIM)	Swiss Federal Inst. of Aquatic Sci. & Tech.	2013	ns-2	Windows, Linux, Mac	TcL, C++	Aquatic Research and Education	✓ — ✗	✓	✓	End-users can define the spatial topology of the communication network.
Sapienza Univ. Net. Framework for Underwater Sim., Emu., and Testing (SUNSET)	Sapienza University of Rome, Italy	2012	ns2-Miracle	Windows, Linux	TcL, C++	Underwater Acoustic Sensor Networks	✓ — ✓	✓	✓	It provides a complete toolset of pre-deployment (i.e. simulation and emulation) and deployment-time (i.e. real-time on sea) testing of the communication protocols [90].
World Ocean Simulation Sys. (WOSS)	University of Padova, Italy	2009	—	Windows, Linux, Mac	C++	Underwater Acoustic Networks	✓ — ✗	✓	✓	It integrates with the existing underwater channel simulators and improves their simulation results.
DESERT Underwater	University of Padova, Italy	2012	ns2-Miracle	Windows, Linux, Mac	TcL, C++	Underwater Networks	✓ — ✓	✓	✓	- The full name is Design, Simulate, Emulate and Realize Test-beds for Underwater network protocols. - The provided libraries extend the ns2-miracle simulator.
Riverbed Modeler	Riverbed Technology	1986	—	Windows, Linux	C++	Commercial Networks	✓ — ✗	✗	✗	- Riverbed Modeler was formerly known as OPNET Modeler Suite. - It is an object-oriented discrete-event simulator. - Huge libraries of accurate network hardware models are available [93].
QualNet	SCALABLE Network Technologies	2000	GloMoSim	Windows, Linux	C++	Commercial Networks	✓ — ✗	✗	✗	The statistical graphing tools in QualNet can display hundreds of metrics collected during the network simulation process.
NetSim	TETCOS	2002	—	Windows	Java	Research and Military	✓ — ✗	✗	✗	NetSim is an end-to-end packet level network simulator.
Objective Modular Net. Testbed in C++ (OMNeT++)	OpenSim	1997	OMNEST	Windows, Linux, Mac	C++	Kernel of Other Simulators	✗ — ✗	✓	✓	- Objective Modular Network Simulation Testbed (OMNEST) is the commercially distributed product of the OpenSim. - OMNeT++ does not contain protocol models and for this, relies on external frameworks, such as INET. - There are other simulators like Castalia, which are based on OMNeT++ and are useful in simulating low-power underwater wireless networks.
Java Simulator (J-Sim)	Several Contributors	2001	—	Windows, Linux, Mac	TcL, Java	Research and Education	✓ — ✗	✓	✓	- J-Sim was formerly known as JavaSim. - It is an object-oriented discrete-event simulator. - J-Sim uses much smaller memory footprint to carry out a similar simulation, compared to ns-2 [93]. - Visualization of the network operations in J-Sim is relies on the ns-2 network animator.

seabed topography, water currents, microscopic organisms, distribution and movement of species, etc. Therefore, the simulation and modeling results should always be supported by experimental fieldworks to discover their weaknesses and to validate the minimum acceptable accuracy of their outputs [94].

Finally, in addition to the general-purpose software packages and frameworks seen in Table V, there are some other simulation tools, like Network Security Simulator (*NeSSI²*), which are particularly designed to research and evaluate network's resistance against various security threats. These cybersecurity tests in IoUT are important, because underwater networks are often deployed in remote locations with difficult access that makes them vulnerable to cyberattacks.

In this regard, a safe underwater network with a secure communication protocol along with a strong data encryption algorithm can protect us against security threats, as discussed in the next section.

H. IoUT Network Security

Underwater networks are usually left unattended for long time intervals, exposing them to diverse potential threats. Due to the wide data broadcasting range of these networks, their physical channel (both wired or wireless) is prone to eavesdropping, hence cannot be kept secure. On the other hand, secure communication is crucial in many IoUT infrastructures [3] (e.g. harbor security, coastal defense, etc.), where unauthorized access can have serious consequences.

The security threats in underwater communications can be divided into passive and active attacks, in accordance to the actions taken by the intruders. In a passive attack, intruders simply try to obtain data, while in an active attack, they will attempt to inject, alter, or delete data by introducing malicious nodes [95]. In either case, the concept of authenticated data access offers a high-level solution, which can prevent IoUT network both from passive data-extraction and from active data-injection. A correct implementation of this concept will assure data confidentiality, data integrity, data freshness, user authentication, and non-repudiation of origin.

However, correctly implementing the concept of authenticated data access can be a challenging task in IoUT, when considering the mobility of network nodes, resource constraints in terms of communication as well as computation capacity, and shortage in energy supplies. This challenging task can be addressed by using cryptographic primitives as well as secure communications, when designing the network [96], which will be discussed in the following subsections.

1) *Cryptography*: Cryptography in network security is the science of encrypting data with a symmetric or asymmetric key, prior to its submission, and in a way that only the authorized receiver can decrypt it [95]. Despite the fact that asymmetric cryptography is more suitable in high-security applications, using symmetric encryption, wherever possible, has substantial benefits for resource-constrained underwater environments. This is because, at a given security level, a symmetric key will be significantly shorter than an asymmetric key [97], which consequently eliminates the need for public key

broadcasting as well as sophisticated algorithm deployment and execution.

Some symmetric cryptography algorithms that are suitable in underwater applications are *Blowfish* and the associated fish products *Twofish* and *Threefish*, as well as the Advanced Encryption Standard (*AES*), Rivest Cipher 5 (*RC5*), and Rivest Cipher 6 (*RC6*).

For asymmetric cryptography on the other hand, using Elliptic-Curve Cryptography (*ECC*) is recommended, as it is energy-efficient with less computation complexity than other popular algorithms, e.g. Rivest–Shamir–Adleman (*RSA*) [97]. *ECC* algorithm has also been shown to require smaller keys to provide equivalent security.

In both symmetric and asymmetric cryptography, a common key has to be shared between the communicating pairs. While the classical key-sharing protocols are common in overwater communications, they cannot be readily adopted by IoUT because of their network infrastructure requirements, complexity, and high resource demands [98]. By contrast, the physical layer key generation seems to offer a promising solution in IoUT. This method relies on the local generation of keys between two legitimate nodes, without broadcasting it in the channel. For example, a multi-party secret key generation scheme is proposed for underwater acoustic channels in [98]. Their highly confidential key extraction scheme is designed for the circumstances of underwater multipath and Doppler effects. Additionally, the simulation results of [98] demonstrate a secure communication against the key leakage in active attacks.

2) *Secure communications*: In contrast to cryptographic primitives, which secure an IoUT network by encrypting any data prior to its submission, covert communication techniques as well as secure protocols may also be applied [97]. The idea behind covert communication in IoUT is to reduce the average SNR in the propagation medium, which consequently reduces the chances of eavesdropping. Covert communication in underwater applications may rely on the following techniques [99]:

- Using an acoustic phased-arrays to direct data propagation beam toward a desired receiver only;
- Using frequency-hopping to repeatedly change the carrier frequency in an unpredictable fashion;
- Using spread spectrum and code division multiplexing to spread the transmitted energy over the frequency band;
- Using analog network coding with concurrent signal transmissions to have intentionally interfering signals.

In addition to the above methods, camouflaged data transmission is another creative technique of covert communications. Generally speaking, the eavesdropping systems classify engine sounds, environmental noises, and biological voices as ocean noise and try to filter them [100]. Here, the underwater acoustic communications can benefit from shaping their output signals in accordance to the time and frequency features of a targeted chaotic signal. Recent studies have verified the efficiency and feasibility of this scheme in IoUT, when the transmitted signal is camouflaged with marine mammal voices (e.g. killer whales) [100], ship-radiated noise [101], etc.

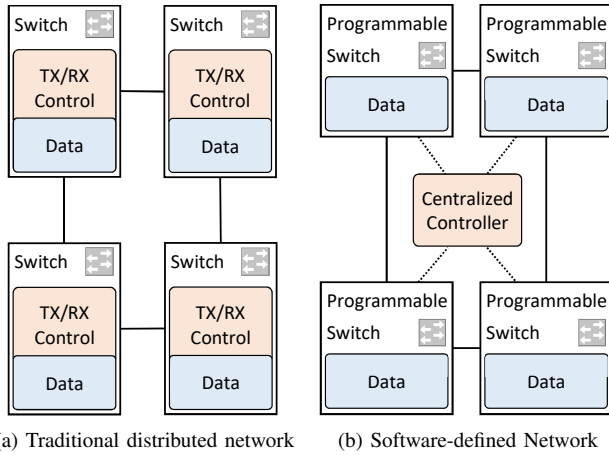


Fig. 6. Traditional vs. software-defined network management.

On the other hand, secure protocols can also be employed by IoUT architectural layers for ensuring network-threat-resilient underwater communications. These layer-specific as well as application-oriented secure protocols can be summarized as [97]:

- *Application layer protocols*: Remote Authentication Dial-in-User Service ([RADIUS](#)) and [Diameter](#);
- *Transport layer protocols*: Secure Socket Layer ([SSL](#)) and Transport Layer Security ([TLS](#));
- *Network layer protocols*: IP Security ([IPSec](#)), Secure FLOOD (SeFLOOD) [102], Resilient Pressure Routing (RPR) [103], and Reputation-based Channel Aware Routing Protocol (R-CARP) [104];
- *Data Link and physical layers protocols*: Extensible Authentication Protocol ([EAP](#)), IEEE 802.1X EAP over LAN ([EAPOL](#)), and IEEE 802.1AE MAC Security ([MACsec](#)).

All the aforementioned secure protocols, along with the network protocols studied in Section II-D, can be categorized under the distributed network management methodologies. In contrast to these distributed controls, a novel centralized approach will be introduced in the next section, which can boost the security of IoUT networks.

I. Software-Defined Networks in IoUT

As seen in Fig. 6, there is a novel centralized management technique, termed as *Software-Defined Network (SDN)*, which is practically beneficial for IoUT [105]. To elaborate, in contrast to distributed networks, where every network node makes its own decisions based on the locally configured routing tables, in SDNs, centralized control nodes continuously monitor the network's traffic flow and manage the network's configuration in order to improve its overall performance.

In contrast to the isolated UWSNs having inflexible hardware-dependent architectures, SDNs tend to be more suitable for the IoUT infrastructure. Some of the beneficial features of SDNs are:

- Programmable and hardware-independent;
- Automatically manageable and reconfigurable;

- Multi-modal (i.e. multi-technology) operation;
- Scalable and adaptable in terms of size and topology;
- Service-oriented (instead of application-oriented) design to offer multi-application functionality;
- Resource-sharing capability;
- Convenient troubleshooting.

In addition to the above-mentioned attributes, SDNs also support spectrum-aware communications, which is an outstanding benefit in underwater networks [105]. Dynamic spectrum management in SDNs is based on its cognitive radio procedure, which includes informed spectrum sensing, intelligent decision-making, and dynamic spectrum sharing. According to these features, a variety of underwater use cases are reported, including software-defined acoustic modems [106], underwater cognitive networks [107], UWSN virtualization [108], and underwater sensor cloud [109].

SDNs are also proved to have a better protection against malicious attacks, as a benefit of their global view over the entire system [97]. Basically, the centralized controller node can monitor and detect the irregular behavior of other nodes as well as suspicious data transmissions. It can then mitigate these abnormalities by instructing the network to fix the problem, or simply by ignoring the untrusted nodes.

SDNs have been also conceptualized to provide a multi-mode opto-acoustic technology in IoUT [110]. Celik *et al.* have used optical base stations as well as acoustic access points in their infrastructural underwater network. These opto-acoustic mid-layer nodes are used to provide coverage to other endpoint nodes in their vicinity, and to serve as a gateway to the rest of the network [110]. They have used SDN as a management enabling technique, both for 'hybridizing' optics and acoustics and for adapting their proposed architecture to the dynamically changing underwater environment. However, this envisioned contribution requires further practical verification, before being accepted as the basis for the future *ad hoc* or infrastructural IoUT designs. Such an experimental study can also reveal the usefulness of this architectural paradigm in support of the IoT and IoUT integration.

Another conceptual hybrid network architecture is proposed by Lal *et al.* [111] for enhancing the security of underwater acoustic networks. This hybrid architecture incorporates SDN, context-awareness, and cognition to boost the physical layer security. The efficiency of their proposed architecture in tackling underwater security situations was evaluated for various attacks including jamming, wormhole and ID-spoofing, black-hole and sinkhole, as well as replay and resource exhaustion attacks. Similar to the contribution in [110], this conceptual hybrid SDN can inspire further practical investigations to secure a network, while maintaining reliable communications.

All the approaches discussed above are introduced to improve the IoUT network architecture and protocols to consequently enhance underwater communication and to address its challenges. However, if the devices and nodes in IoUT are capable of local computations at the edge of the network, the underwater network traffic will be significantly reduced and further improvements can be made to the IoUT ecosystem. The concept of these edge computing nodes and their beneficial

effects on underwater data traffic will be discussed in the next section.

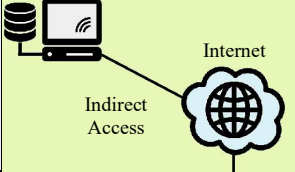
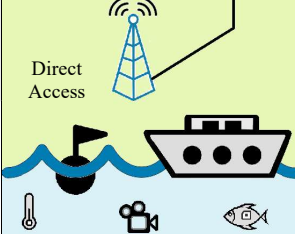
J. Edge Computing in IoUT

The concept of edge computing was originally introduced in IoT as an alternative to cloud computing. As the terminology suggests, in edge computing, the endpoint (edge) devices perform all or part of the required computations, so the need for data transfer and communication becomes less of a challenge. This sparse data transfer is ideal for the IoUT ecosystem, which suffers from the hostile communication medium. Within IoUT, edge computing can be defined as a distributed and elastic computing paradigm, in which computing is predominantly carried out in the edge-devices such as, underwater endpoints, mid-layer nodes, and data acquisition tools. In the absence of edge computing capability, the processing should be performed on local computers, servers, or by the centralized clouds [112], all of which require shuttling data back and forth, hence rendering it unsuitable for IoUT [113]. In edge computing, devices have to expand their capabilities and in addition to data collection and communication, engage in data processing as well. This feature will shift the services from a single centralized point to numerous distributed nodes and closer to the physical world. The major advantages of using such a decentralized edge computing technology in the IoUT ecosystem are data-rate reduction, latency reduction, and prompt inner-network decisions making [113]–[115].

However, to turn these advantages into reality, edge computing has to tackle significant challenges. As stated in Section II-D5, the energy resources of the edge-devices are limited. Furthermore, sustainable power cannot be readily delivered to the processing units of underwater applications. Even though edge computing has been the subject of extensive research in conventional IoT applications, it has its own specific challenges in IoUT. Below, we briefly review two of the IoT edge computing paradigms presented in [116] that are suitable for IoUT.

- *Mobile Cloud Computing (MCC)*: It is defined in IoT ecosystem, as the combination of cloud computing and mobile nodes to execute computational- as well as storage-heavy mobile applications (e.g. machine learning) in the cloud. MCC can offer rich computational resources to scarce-resource underwater applications. However, its relevance to IoUT is low because of the long propagation distance from the distributed BMD sources to the remote cloud servers [117], as well as the narrow underwater bandwidth and limited access to energy. Another drawback of MCC is its service accessibility, which is via Internet connection only. This is in contrast to other edge computing paradigms that can offer a direct access [116].
- *Mobile Edge Computing (MEC)*: The acronym MEC represents mobile edge computing and multi-access edge computing, with the latter one being more inclusive. According to the descriptions offered by the European Telecommunications Standards Institute (ETSI), the MEC technique provides cloud computing capabilities at the edge of the network, with close proximity to the end-users

TABLE VI
LOCATION OF THE PROCESSING UNIT IN DIFFERENT IOUT EDGE COMPUTING PARADIGMS

Edge Paradigms	Edge Processing Locations
Mobile Cloud Computing (MCC)	 Centralized cloud servers at the land side of the network
Mobile Edge Computing (MEC)	 land stations at the edge (e.g. radio access networks) Overwater sink nodes or underwater mid-layer nodes (e.g. buoys and ships)

(i.e. underwater endpoint nodes). In IoUT, the edge of network represents both the land stations at the edge (e.g. cellular towers, data centers, Wi-Fi routers, etc.) [118] as well as the overwater sink nodes (i.e. floating buoys, floating vehicles, exploration platforms, etc.). These edge gateways will carry out some preliminary and short-term tasks, before handing their results to the cloud servers for more sophisticated and resource-intensive analysis [119]. MEC techniques invoked for the IoUT can offer low latency and reliability. Their other advantages in IoUT include their support of wired communications as well as their facilitated direct access (i.e. no Internet connection is required) [116].

Considering the above, MCC and MEC are two of the IoUT edge computation techniques that are practically realizable in IoUT, as shown in Table VI. Among them, MEC requires lower bandwidth, lower network traffic, low-latency, and low-power underwater operation, and more reliable access to underwater sensors and cameras in the IoUT [114]. Hence, MEC networks constitute the most attractive edge computing paradigm in IoUT applications [114], [115]. They can perform computations in the sink or mid-layer edge-devices that are placed or have access to above water, where for example solar energy may be harvested.

K. Section Summary

In this section, we studied the essential topics involved in developing the IoUT. We discussed various sensors in the IoUT and learned about their important features and roles in data collection. We then pointed out the major challenges in the IoUT domain and learned that electromagnetic, acoustic and optical signal attenuation significantly affect the IoUT communication and its link reliability. We also learned how to measure and improve the link reliability. Next, we discussed the IoUT architectural model and its layer-wised protocol stack, and learned how the TCP/IP model can be adapted to the IoUT architecture. In order to be able to gain better insight into

the network performance and to predict its reliability, network modeling was discussed. The lessons learned were that various channel models result in different levels of complexity and they should be carefully selected, based on the application concerned.

The IoUT network topology was the next topic we covered, where we learned that the tree and partially connected mesh are the most popular topologies in IoUT applications. We also discussed the underwater network simulation tools and how they can be used to facilitate the implementation of the defined protocol stack. We provided a list of these simulation tools, which can be helpful to the community when designing and analyzing underwater communication networks and protocols. Network security in IoUT was another salient topic discussed. We learned that there is a paucity of literature on this important issue, even though it is essential in critical scenarios such as harbor security. Another important topic studied in this section was software-defined networks in IoUT. We learned how it can help with the management of IoUT networks. Finally, we discussed edge computing in IoUT and learned that it may be even more important for IoUT devices to have edge computing capability than for their IoT counterparts, mainly due to the more challenging communications in underwater scenarios.

The IoUT discussed in this section generates a vast amount of data, which may be referred to in parlance as *big marine data*. In the following section, we discuss big marine data and cover its associated data sources, data collection tools, and data processing methods in the oceanic and underwater domain.

III. BIG MARINE DATA

This section is dedicated to underwater data types, which are envisioned to have the lion-share in the IoUT data transactions of the near future. We will also cover data acquisition tools, localization as well as tracking, and will introduce some ready-to-use big ocean sensory, imagery, video, and geographic databases, which are useful to researchers and practitioners in the IoUT Domain.

BMD can be succinctly defined as the vast amount of heterogeneous data collected from marine fields. The main characteristics of BMD are temporally long and spatially vast coverage, diverse nature of the data sources (e.g. sensors, cameras, tags, aerial remote sensing), and multi-disciplinary data types (i.e. physical, chemistry, biological, environmental, economical, etc.) [34].

The BMD system components and processing stages are as follows [32]:

- 1) *Acquisition*: Involves collecting raw data.
- 2) *Transportation and security*: Requires the data to be encrypted and transferred across different communication media to its target storage, while considering its reliability and security.
- 3) *Storage and privacy*: Deals with policies around data storage (legal concerns and users privacy) and its archival requirements (file formats, retention lifecycle, and replication).
- 4) *Special-purpose processing*: For complex datasets, bespoke software packages are required for searching, pre-processing by filtering and cleaning, recognition and

labeling, post-processing and visualization of results, and updating.

- 5) *Exploitation and leveraging*: Ensures that users gain benefit in terms of increased revenue from their data. Some benefits of underwater data exploitation are that of monitoring the water's vital cleanliness, help new businesses to grow, support experts by providing informative bespoke data, track worldwide maritime transportations, and protect the environment.

From the above five BMD system components, *transportation and security* was already discussed in Sections II. The third component, i.e. *storage and privacy* covers legal policies and data handling issues, which are beyond the scope of this paper. Below, we review the *data acquisition* stage and later in sections III-C and III-D we will discuss some aspects of *data leveraging* stage in more details. Additionally, discussions on *data processing* will be provided in Section IV.

A. Marine Data Acquisition

Data acquisition, which is the first component of any BMD solution can be discussed in three different stages including data gathering, data aggregation, and data fusion.

1) *Data gathering*: Data gathering in IoUT can be performed using a variety of tools, some of which are listed in Table VII. The data acquisition tools in this table are divided into vehicles and primary data sources. Any vehicle in this table can be considered as a data acquisition tool, if and only if it is equipped with one or more primary data sources and tools (e.g. mounted cameras, hydrophones, sensors).

As we will discuss in more details in the forthcoming sections, BMD processing and data leveraging (as the 4th and 5th BMD system components) are not possible without automatic ML approaches. Therefore, a survey was conducted to extract the contribution of data acquisition tools in state-of-the-art ML articles. The survey result is shown in the second and fourth columns of Table VII. In the second column, the inter-category contribution of each item is compared to the others. In the vehicular category, the unmanned ROVs and AUVs together were used in about 95% of the relevant ML publications.

In the fourth column of Table VII, an item-specific evaluation is performed to quantify the share of each item both in the old and new ML publications (before and after 2014). As can be seen in the table, more than 50% of all underwater research in the area of ML have been carried out after 2014. There is only one exception for human-occupied vehicles. Perhaps, this is due to the recent wide adoption of automated as well as of remote methods and owing to the reduction in academic usage of costly manned vehicles.

2) *Data aggregation*: Data aggregation is a statistical data processing stage before higher level calculations and/or before transmission over a band-limited communication channel [120]. The level of raw data summarization here is to create another set of shortened raw data, by implementing mathematical techniques, such as down-sampling, linear regression, etc.

Data aggregation can be carried out in UWSNs, for example to minimize the bandwidth utilization. It can also reduce

TABLE VII
LIST OF DATA ACQUISITION TOOLS IN UNDERWATER APPLICATIONS WITH THE PERCENTAGE OF USE OF EACH ITEM IN RELATED ML ARTICLES

Data Acquisition Tools	Share in ML	Types of Data Acquisition Tools	Used in ML Articles	
			Before	After 2014
Vehicles	50%	Remotely Operated Vehicles (ROV) • Free Swimming • Bottom Crawling • Structurally Reliant • Towed System • Hybrid Remotely Operated Vehicles		62%
	45%	Autonomous Underwater Vehicles (AUV) • Underwater AUVs • Underwater Gliders (Seaglider) • Autonomous Submersibles		54%
	5%	Human-Occupied Vehicles • Submersibles (civilian research submarines) • Atmospheric Diving Suits		88%
Primary Data Sources	65%	Sensors • Separately Listed in Table II		50%
	35%	Cameras • Fixed Cameras (digital still camera) • Baited Remote Videos • Diver Operated Videos • Drifting Underwater Camera System		57%

the energy consumption in the network-level, by striking a trade-off between low data processing power and high data communication energy. However, using this process in underwater applications may impose some unwanted drawbacks, like [121]:

- Increasing the energy consumption and processing requirements in the node-level;
- Increasing the overall network latency;
- Compromising data accuracy by shortening its volume.

To address these challenges, LEACH, PEGASIS, HEED, and APTEEN, which are some well-known aggregation protocols in wireless sensor networks can be used in UWSNs subject to modest adjustments [122]–[124]. Additionally, heavy computations and high energy demands in data aggregation processes can be mitigated by using edge computing paradigms, as described in Section II-J.

3) *Data fusion*: The BMD gathered and aggregated may be stored in distinct subsystems or in separate databases. Data fusion is defined as the combination of relevant data from different data sources into an integrated dataset, with the objective of searching for more consistent and more accurate information than that provided by any individual database [125].

Data fusion systems in IoUT mainly rely on edge computing devices as well as on cloud computing servers to handle large amounts of heterogeneous BMD (i.e. sensors, audios, videos, commands, etc.) [37]. Three main questions are raised when designing a successful data fusion system in marine applications. The first is the location of data fusing operation, the second is the level of abstraction in the fusion system, and

the third is the level of overlapping of the original data.

Considering the first question, data fusion can be performed either in a centralized node or in a distributed network [125]. In a centralized scheme, the data fusing node is usually located on overwater edge devices or in an inland facility with good access to energy resources [37]. The distributed data fusion will be hierarchically conducted over the entire system, where every cluster-head fuses data from its own cluster nodes, passing them to the next cluster-head. Distributed BMD processing will be discussed in more details in Section III-C. Considering the second question about abstraction level, the answers offered by [126] are:

- *Low level (i.e. sensor fusion)*: If the inputs to our fusion system is the raw data directly gathered from the sensors, cameras, etc.;
- *Medium level (i.e. feature fusion)*: If the inputs to our fusion system is the output of feature extracting blocks, as will be discussed in Section IV-D;
- *High level (i.e. decision fusion)*: If the inputs to our fusion system is the output of classification or clustering blocks, as will be discussed in Section IV-A.

Low level data fusion is not recommended in BMD, as a result of restricted underwater resources. The only exception is in edge computing, where the raw data is fused together on the edge device, right before extracting their features. On the other hand, medium and high level data fusion can help us with building more sophisticated models or with finding more complex solutions [127].

Considering the third question on the overlapping level, the lessons of [126] are:

- *Redundant*: If we have multiple datasets with the same data type, related to the same subject, in the same time interval, by different data acquisition tools or in different sampling time spots;
- *Complementary*: If we have multiple datasets of the same data type, related to the same subject, from different angles or in different time intervals;
- *Cooperative*: If we have multiple datasets of different data types, related to the same subject.

Fusing redundant or complementary data can increase our confidence over the original datasets. However, based on the expensive data transmission in UWSNs, neither redundant nor complementary levels of overlapping are recommended. In contrast, cooperative data fusion (e.g. couple of sensed parameters from Table II or associated audio and video data) can obviously increase our knowledge of the subject.

B. Marine Data Classification

Based on our findings in previous section, which has also been reflected in Table VII, cameras and sensors are considered as primary underwater data acquisition tools. Most maritime industrial, research, or observatory project might integrate these data acquisition tools into a wireless or wired sensor network. The sensor networks that are connected to the Internet, all together play a predominant role in the overall IoUT infrastructure. The data collected in the UWSNs within the IoUT may be classified according to its dimension as discussed below.

1) *One-dimensional time-series marine data*: As described earlier in Sections II-A and II-D1, a sensor is considered as an endpoint node in the application layer, which offers time-series data reception through the Internet according to the end-user preferences. For instance, a biologist might be interested in the water temperature, while an environmentalist might be interested in the water quality.

Conventional marine sensors and marine nano-sensors (as listed in Table II) measure a variety of oceanic signals and processes within a specific duration. Marine nano-sensors are popular owing to their low power consumption [128], [129]. The sequential data provided by sensors and nano-sensors are used in monitoring and surveillance applications to provide long-term and large-scale perception of the environment and to tackle unwanted environmental changes [130].

Implementation of these sensor nodes in small-scale research-based IoUT structures are repeatedly reported in literature. Some of these contributions are discussed below:

- A low-complexity VHF-based IoUT ecosystem is proposed by Al-Zaidi *et al.* [42] for marine data acquisition based on storage devices in the cloud. The proposed structure is equipped with temperature, depth, and wind sensors to produce a near real-time system.
- A network of sensor nodes, based on the IEEE 21451 standard is constructed by Adamo *et al.* [41] for continuous monitoring of the seawater quality. This system is devised as an IoUT network for making strategic decisions concerning a range of environmental issues.

- A low-cost technology for maritime environmental sensing is described by Wright *et al.* [131]. The technology relies on the IoUT for measuring parameters such as the optical properties of water, ocean temperature, and wave dynamics.
- The Great Barrier Reef of Australia is the largest coral system on planet Earth, spanning a distance of over 2300 km. Palaniswami *et al.* [132] constructed an UWSN to capture data from temperature, pressure, and humidity sensors in an IoUT platform, for monitoring the complex ecosystem of the Heron Island in the southern Great Barrier Reef. The big data of sensory time-series collected for their study, has been then analyzed by ML algorithms to detect underwater anomalies. Therefore, [132] is a notable application of big data analytics in IoUT with prime objective of predicting severe tropical cyclones.

In addition to these small-scale research units, a range of other observatory stations have been established by institutes and organizations, to provide both research-oriented and industrial access to the underwater sensory data. These observatory stations are located all around the planet and they provide a reliable access to BMD through the IoUT.

Here we provide a list of major observatory stations in Table VIII. The observatories are sorted in an ascending order from the smallest to the largest observatory coverage area. All the entries in this table are open access and indicate active projects with sustainable funding. This guarantees continuous data preparation and update. They are all accessible through a web-form or dedicated software.

There are some other observatory projects, which are not included in Table VIII, because they are no longer supported and their databases have become obsolete. For instance, the [POSEIDON system](#) in Greece (2008 to 2012) is no longer actively operating, but its atmospheric and marine data are still downloadable through the web. Another example is the [JGOFS](#) (Joint Global Ocean Flux Study) project which was funded by the international science council during 1987 to 2003. JGOFS was an international program with participants from more than 20 nations. The rich multi-disciplinary data of this project is also still available to download.

The data provided by any one of the observatory systems in this section, can be affected by environmental noise, outlier records, misread values, and missing quantities. To deal with these measurement errors, we require a series of techniques, which will be provided in Section IV-C1.

2) *Two-dimensional underwater image data*: Some observatory stations introduced under Table VIII, are equipped with cameras to provide ready-to-use 2D image data. In addition to those live IoUT images and videos, there exist a variety of other still-image databases. These new databases are listed in Table IX, and they are eminently suitable for supervised ML applications (as it will be discussed in Section IV-A), as a benefit of their additional expert labels and annotations. These databases are accessible through a web-form or a dedicated software, or even in the form of a downloadable dataset. However, only a few of them are active projects with continuous updates.

TABLE VIII
OPEN ACCESS AND ACTIVE OBSERVATORIES WORLDWIDE; PROVIDING UP-TO-DATE PRIMARY DATA

Project Name	Funder	Description
Expandable Seafloor Observatory (OBSEA)	University of UPC	A seafloor observatory located in a fishing protected costal area in Spain . It is connected to the coast by a mixture of energy and communication cables to deliver live video , acoustic hydrophone, and sensor data.
Woods Hole Oceanographic Institution (WHOI)	Non-profit Organization	An independent organization in Massachusetts, The USA that is dedicated to ocean research, exploration, and education. It owns some observatory stations (like MVCO) and provides real-time and archived coastal sensed data accessible by the public.
Monterey Accelerated Research System (MARS)	MBARI Institution	This cabled observatory provides power and data connections in the deep-sea (891 <i>m</i> below the surface of Monterey Bay in California, The USA) and provides camera views, videos , Acoustics , and sensor data (O ₂ , transmissivity, salinity, etc.) to study the ocean.
Lofoten Vesterålen (LoVe)	Statoil	This station provides a variety of data types, including camera views, active and passive acoustics , and other data gathered from many sensors positioned in the cold-water corals in Norway [133].
Next Generation of Sensors (NeXOS)	European Commission	Miscellaneous sensor data from 3 sites around Italy and Spain are gathered (acoustic, carbon, hydrocarbon, fisheries, etc.) [26].
Marine Network (MARNET)	BSH®	A long list of current and historical sensor data (chlorophyll, currents, salinity, turbidity, etc.) for the North sea and Baltic sea of Germany are available to view on-line or to purchase as digital datasets.
Channel Coastal Observatory (CCO)	DEFRA Department	The network of regional coastal monitoring program consists of six regional monitoring sites all around the UK . The sensory data gathered in this program include pressure, wave height, tide, etc.
Integrated Ocean Observatory System (IOOS)	NOAA	This project integrates some existing networks of instruments in the western hemisphere (e.g. OOI , NERACOOS , MARACOOS) and provides videos , tagged camera views, and sustainable sensed data from regions around The USA including Alaska, Caribbean, California, Gulf of Mexico, Mid-Atlantic, Northeastern-Atlantic, Northwest Pacific, and Southeast Atlantic.
Australian Ocean Data Network (AODN) and Australian Institute of Marine Science (AIMS)	Australian Government	According to AIMS, Australia's marine territory is the third largest on Earth. AODN and AIMS portals together, provide access to Australian marine and climate sensed data and metadata in coastal areas and the Great Barrier Reef. These databases are collected from miscellaneous sources including IMOS , NOAA, NASA, ARGO project, etc.
Ocean Networks Canada (ONC)	University of Victoria	This network continuously delivers data from cabled observatories , remote control systems, and interactive sensors to monitor the west and east coasts of Canada and also the Arctic. This platform merges all individual Canadian observatories.
European Multidisciplinary Seafloor and water column Observatory (EMSO)	European Union	EMSO consists of 8 observatories and 3 test sites that placed around Europe , from North East to the Atlantic, through the Mediterranean, to the Black Sea. These observatories are equipped with multiple biological, chemical, and physical sensors , placed along the water column and on the seafloor.
EU Marine Observatory and Data Network (EMODNet)	European Union	This European data network includes seven broad disciplinary themes of bathymetry , geology , physics , chemistry , biology , seafloor habitats , and human activities and provides a wide range of marine camera views, taxonomy, GIS, sensed data, etc.
Joint Commission... (JCOMM) in GOOS	UNESCO	A global scale and Intergovernmental sensor network that integrates many famous observatory programs worldwide, including ARGO , DBCP , OceanSITES , SOT , GO-SHIP , and GLOSS .
NCEI , Landsat Aquarius , SARAL CryoSat , Jason , HY2-A IRS	NOAA NASA ESA ISRO	Global sensing of ocean surface by USA , European , and Indian satellites, provides information about sea surface temperature, surface winds and wind stresses, surface Flux analysis, visible and infrared imagery and radiometry, surface salinity, surface topography, etc. Note that NCEI is a satellite constellation of JPSS, GOES, POES, DMSP.

The multidisciplinary data provided by these image databases can be used in different branches of science. For example, while an ecologist may be interested in counting and tracking of individual inhabitants, a data scientist or computer programmer might need these data to train a ML algorithm and to verify its performance.

When considering the usage of these imagery data, one should be aware of the lighting conditions in the underwater environments. This can be significantly different from over-water photography and it is expected to affect the images taken undersea. To understand the nature of light in undersea imaging systems, we have to discriminate passive and active images, i.e. whether the imaging equipment creates its own light or not. Although the physics of optics are the same in both cases, different sets of parameters have to be considered to provide a better understanding of the context of underwater optical imaging.

Active imaging relies on the explicit usage of artificial light in the process of underwater imaging. This type of photography benefits from a substantial improvement in image quality, especially when the light is appropriately controlled by an optimized hardware configuration. However, active imaging suffers from high underwater energy loss (especially in case of long-term illumination), reduced portability, and unpleasant inhomogeneous intensity and color of the final picture [135].

As depicted in Fig. 7, undersea active imaging always encounters 3 sources of light rays into the camera. Direct reflection from the target object is the desired signal, while the two other reflections, termed as *backscatter* (which has not interacted with the target) and *forward scatter* (or blur component) are both undesired.

Backscatter or Volume Scatter: In active imaging scenarios back scattered light is defined as the one which has never interacted with the target object and usually appears as bright points

TABLE IX
OPEN ACCESS DATASETS OF STILL-IMAGES, TAKEN FROM UNDERWATER ORGANISMS

Project Name	Funder	Description
Institute for Marine and Antarctic Studies (IMAS)	University of Tasmania	This Australian institute hosts three major image, taxonomy, and atlas database projects including Reef Life Survey (RLS), Temperate Reef Base (TRB), and Zooplankton .
Sea Life Base and Fish Base	Non-profit Organization	FishBase is a project of FIN [®] (Philippines) with around 59,000 images of fish. SeaLifeBase is a joint project of the University of British Columbia (Canada) and The FIN [®] (Philippines), which includes 12,500 images of different marine species.
World Register of Marine Species (WoRMS)	Life Watch	Provides an authoritative list of valid and vernacular names for 250,000 global marine organisms. This Belgium -funded infrastructure contains extra information such as 32,000 images , literature, biogeographic data, and parent taxon. It is generally used by scientists for classification of marine species.
Fish 4 Knowledge	European Union	Provides three different datasets with annotated ground-truth for fish recognition (30,000 images of 23 species), fish trajectory (9000 trajectory of 23 species), and benchmark for complex backgrounds (14 videos of 7 backgrounds). It also offers access to live video feeds from 10 underwater cameras in reefs of Taiwan .
Encyclopedia of Life (EOL)	Multiple Organizations	Provides 4,000,000 Images of all life-forms on Earth (e.g. animals, plants, bacteria), including 15,000 images and videos from marine environment. It is financially supported by institutions from Australia , The UK , The USA , Mexico , and Egypt .
Shape Queries Using Image Databases (SQUID)	University of Surrey	Includes 1100 images of marine creatures in a smooth background (one creature in each picture). The ground-truth coordination of boundaries of animal's body in all images is annotated (256 to 1,653 points for each creature). This database is located in Tje UK and will not be updated anymore.
RSMAS, EILAT, EILAT II, and SDMRI [16], [134]	Multiple Organizations	These independent datasets contain 766, 1123, 303, and 100 images of corals, which have been categorized by experts into 14, 8, 5, and 20 different classes of coral genera and non-coral, respectively. They are also equipped with color reference pallets for color enhancement.
Moorea Labeled Corals (MLC)	University of Berkeley	This is a subset of the MCR-LTER dataset and has 2055 coral reef images with almost 200 human expert point-annotations over each one of them. These images are taken from the island of Moorea in French Polynesia and contain 5 coral and 4 non-coral distinct classes. They also contain pallets with known reference colors; making them suitable for image correction and color restoration.

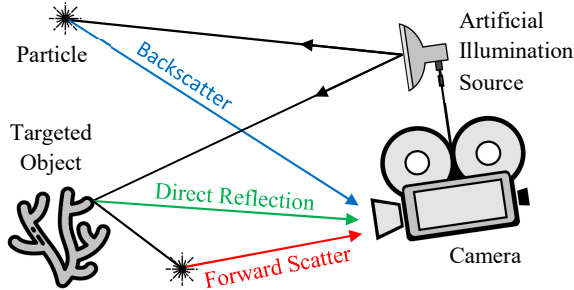


Fig. 7. Schematic view of three different optical rays, which are reflected back to the camera, in active underwater imaging.

in the output image. To accurately calculate the magnitude of light that is backscattered to the camera, we have to calculate or measure the light intensity ($I [w/m^2]$) first. Light intensity (or irradiance) is the power received by an illuminated surface perpendicular to the direction of propagation, per unit area. This parameter has to be evaluated for the specific volume that the camera is viewing. One of the popular methods to achieve this goal is proposed in [49], which discretizes the entire volume to small cubic cells and then calculates the Volume Scattering Function (VSF)¹ parameter for each cubic cell.

The VSF number of each unit cell is then used as a weighting function in the next step. Finally, the method of

[49] considers the VSF , the light intensity I , and the angle between the incident and the reflected light for numerically estimating the magnitude of light that is backscattered to the camera. Again, this is an efficient and accurate algorithm of estimating the magnitude of the backscattered light. However, it can be used as a fundamental step of any image enhancement procedure, especially in underwater applications, where the presence of backscattered light is indeed a significant challenge.

Forward Scatter or Blur: Forward scattered light is defined as a light beam, which interacts with the target and then it is indirectly reflected back to the camera. To approximate the output image of $E_{blur}(x, y)$ in a scattering environment, a common method is to convolve the original image $E_o(x, y)$ with the Point Spread Function ($PSF(r)$) as [49]:

$$E_{blur} = E_o \otimes PSF(r), \quad (13)$$

where r is the distance from the camera to the object and $\otimes$ is the convolution operator.

PSF represents the spatial impulse response of a scattering environment between a light source and the point of observation (e.g. camera), as already shown in Fig. 7. It is used in Fourier optics to calculate the output image of a linear imaging system. The formulation of PSF is simplified to [49]:

$$PSF(r) = [(e^{-Gr} - e^{-cr}) F^{-1} \{e^{-Brf}\}], \quad (14)$$

where c is defined in (4), G is an empirical constant ($|G| \leq c$), B is an empirical damping factor, and $F^{-1}(\cdot)$ is the inverse Fourier transform.

¹ VSF is an inherent optical property of water, which does not vary with the incident light field. A number of experimental methods are introduced in [49] to measure the VSF .

Both e^{-Gr} and e^{-cr} in (14) represent the forward scattering amplitude attenuation, while e^{-Brf} stands for the frequency-dependent damping. The use of the convolution operator in (13) indicates that this equation is only valid in linear optics, which is the case in the majority of underwater light-based experiments [49]. Although PSF in (13) is used for the forward scattering formulation, it can be employed to calculate the VSF in backscattering as well. The PSF formulation as well as its relationship to the VSF has been studied in great detail in [49].

The forward- and back-scattering process encountered in seawater disperse the light beam, hence resulting in blurred underwater images. Observe from (14) that the image gets more blurred, when the distance increases. One can exploit this equation to mitigate the blurring problem with the aid of image deblurring methods, which will be covered in Section IV-C2.

By contrast to active imaging, in passive imaging, i.e. where no artificial light is generated by the image capturer, even though the power consumption will be significantly lower, other concerns may be present. These concerns include limited visibility, contrast, and color distortion.

Visibility: The first concern in passive underwater imaging is always the visibility. In clear sunlit water, ambient sunlight provides a clear vision in relatively shallow water. In ordinary line-of-sight underwater applications, the distance of visibility can be modeled as,

$$d_{Visible} \approx \frac{5}{c - K(\theta, \varphi, z)\cos(\theta)}, \quad (15)$$

where K is the diffuse attenuation coefficient, which is an apparent optical parameter, while θ , φ , and z are the relative spherical coordinates of the subject of interest [136]. Based on this formula, underwater visibility decreases rapidly after a few meters. The only known remedy for this physical constraint in deep-waters is to use an artificial light source in active imaging [135].

Contrast: Contrast is defined as the color or gray-level difference between adjacent areas in the presence of light attenuation, optical noise, and vision blurring. If we consider a target object (O) against its background (B), the contrast of the object and its surrounding areas can be formulated as [137],

$$C_{OB} \triangleq \frac{L_O(\theta, \varphi, z) - L_B(\theta, \varphi, z)}{L_B(\theta, \varphi, z)}, \quad (16)$$

where L_O is the light level (radiance) of the object and L_B is that of background. The knowledge of C_{OB} will help us design contrast-enhancement algorithms for improving the contrast in underwater environments [138].

Color Distortion: As previously discussed in Section II-B and based on (4), light absorption (or attenuation) in underwater propagation strongly depends on the wavelength. Accordingly, all of the visible light wavelengths present in sea surface, provide a crystal clear view. By contrast, red light hardly penetrates below 10m, which is the reason of the greenish color of seawater. Colors having shorter wavelengths (i.e. blue color family) penetrate even deeper to 20 and up to 50 m in sea water, resulting in low-contrast bluish images.

Color distortion and the visibility are the main reasons why passive imaging is inefficient below 10 meters and also why the use of artificial light sources is inevitable in deep-water explorations. To address this important issue, the image enhancement or color restoration methods of Section IV-C2 have to be involved.

Having surveyed some of the 1D signals generated by sensors and the 2D image data types encountered in IoUT, let us now focus our attention on video or 3D data, that forms a large subset of IoUT datasets.

3) *Three-dimensional underwater video data:* As discussed, some of the observatory stations listed in Table VIII are capable of delivering live or archival ready-to-use video data. However, the main challenges in underwater video streaming are related to the limitations of undersea communication, discussed in Section II-B, aggravated by the underwater imaging difficulties studied in Section III-B2.

Although the underwater optical parameters and imaging constraints are all the same as in 2D imagery (i.e. contrast, color distortion, visibility, backscatter, and forward scatter), underwater video imaging has an extra intrinsic impediment that has to be addressed. This barrier, is of course, the continuously growing data volume produced by cameras. A typical camera operating at 1 frame/sec may generate 30 million frames (equal to almost 3 Tera bytes of data) per year [139]. This data volume is then multiplied by the number of cameras, mounted in a single observatory system.

As a rule of thumb, in the current version of the semi-automatic marine image-annotation software, every minute of video requires an expert to spend about 15 minutes for manual annotation [139]. Therefore, to analyze the video produced by a single camera in a single month, we approximately need 10,800 man-hour. Hence, there is a growing necessity to develop automatic video processing methods to deal with this excessive data volume. These automatic methods can be used in a variety of underwater video data applications such as, visible light video tracking [140], sonar video tracking [141], photo mosaicing underwater [20], and marine life studies [14].

In addition to the above-mentioned potential use cases of automatic video processing, the growing field of depth-based video may also significantly benefit from automatic processing. This technology uses optical multi-camera systems [142], [143], acoustic arrays [144], Time of Flight (ToF) depth sensors [145], and laser beams [146] to provide rich 3D information about the scene. This stereoscopic video technology has been used in underwater vehicles (e.g. ROVs, submersibles, etc.) [147], [148], in tethered underwater platforms [149], towed systems [140], and baited stations [150]. They are used in a wide variety of applications, including photogrammetric bundle adjustment [151], [152], 3D scene and organism reconstruction [143], [145], underwater 3D live tracking [140], [152], quantitative analysis and sizing of targets [140], [149], counting and measurement of marine creatures [150], as well as improving the segmentation and classification capabilities of traditional 2D algorithms [150], [153].

However, the sophisticated depth-based vision technology has its own challenges both in the context of acoustic and optical recording methods. While acoustic depth vision always

suffers from variable sound velocity, reverberation, as well as unwanted noise and echoes, optical stereoscopic cameras encounter blur and haziness in turbid water, aggravated by unstable illumination, and light refraction [140]. Perhaps, the most direct way of addressing this problem is to use improved acoustic and optical channel models for compensating the deficiencies of each technology [151]. Another solution is to combine both technologies into a unified opti-acoustic 3D imaging system. The latter solution is capable of offering improved precision, if both technologies cooperatively calibrate each other [148].

So far in this section, we have provided a detailed evaluation of the family of IoT sensors, as well as of image and video data sources. Additionally, some ready-to-use databases were introduced. Once the raw big data has been gathered, a high-performance data processing platform is required for inferring knowledge. Due to the scarcity of resources and owing to the limited power budget at the data collection points, processing cannot be performed locally. Therefore, the system has to rely on cloud-based or on distributed data processing platforms, which will be studied in the next section.

C. Distributed and Cloud-based BMD Processing

To meet the growing demand for big data processing, many high-quality software and services have been created to offer big data analytics. Some of these platforms are Apache, Amazon EMR, Microsoft Azure HDInsight, Cloudera, Hortonworks (which has recently been merged with Cloudera), SAP-Hana, HP-HAVen, 1010data, Pivotal data suite, Infobright, etc. [12], [32]. Each of these services responds to the big data processing demands by either providing distributed processing frameworks or cloud computing services.

Distributed processing systems consist of networked computers, which cooperate with each other to offer high performance data processing [154]. As seen in Fig. 8, among all distributed big data analytics frameworks, the Apache is the dominant platform, which has been used in about 95% of all reported scientific articles. Apache software foundation is a not-for-profit corporation, founded in 1999 to support more than 350 open-source Apache software projects, and 47 of them (about 13%) are directly related to big data analytics. More than 80% of the Apache platforms have been developed in only five languages (i.e. Java ($\approx 60\%$), different types of C ($\approx 10\%$), Python ($\approx 5\%$), JavaScript ($\approx 3\%$), and Scala ($\approx 3\%$)) [155].

Table X introduces the advantages and disadvantages of the four widely used Apache distributed processing frameworks including Spark, Hadoop, Storm, and Flink. All of these open-source projects are supported by Apache software foundation and they are eminently suitable for high-speed processing of IoT-generated BMD analytics.

The term *cluster* in Table X refers to any computer system or embedded system in the parallelized network of computers, each having its own processor, memory, and network IO. Furthermore, the directed acyclic graph scheduler is based on a specific type of directed mathematical graph having no cycles, in which it is impossible for the data to emanate from a

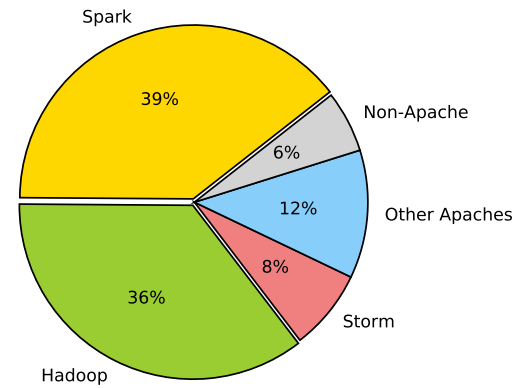


Fig. 8. Distributed big data frameworks used in data analytics papers published in IEEE and Web of Science databases.

vertex (i.e. any standalone computer) and pass over a non-zero number of graph edges (i.e. inter-computer connection cables), and to eventually loop back to the starting vertex again.

In addition to the locally distributed processing systems, there exist companies who offer cloud-based services for all aspects of big data storage, integration, streaming, ML, and *ad hoc* data analysis. According to the Statista, the share of software and services in overall big data revenue will double in the 6-year period spanning from 2019 to 2025. This means that the organizations will rely more on cloud services to handle their sophisticated use cases [156]. In other words, big data processing is expected to gradually shift from distributed to cloud-based services. This is an ongoing trend, as the number of businesses performing their big data analysis in the cloud has increased from 58% in 2017 to 73% in 2018 [157].

As already mentioned in Section II-J, BMD can partly be processed in the IoT-edge, before being transmitted to land. After transmission, there is no difference between the conventional IoT and IoT-based distributed or cloud-based data processing. Nevertheless, all available cloud services today are offered in accordance with one of the following models [158]:

- *Software as a Service (SaaS)*: This service model relies on centrally hosted software, which delivers specific services to licensed or subscribed clients and usually offers its dedicated functionality through a browser.
- *Platform as a Service (PaaS)*: This service model targets developers by providing them with operating systems, databases, software packages, application services, etc. It helps developers to focus on the development of their diverse applications, instead of software resource management and hardware maintenance of the underlying infrastructure.
- *Infrastructure as a Service (IaaS)*: This flexible low-level service model targets both developers and businesses by providing access to the underlying infrastructure (e.g. processors, data partitions, security, backup, etc.). This service tends to rely on high-level Application Programming Interfaces (API) in support of the network operations. While the service provider is responsible for the hardware maintenance, the users are required to configure

TABLE X
MAJOR APACHE FRAMEWORKS SUGGESTED FOR DISTRIBUTED PROCESSING OF BMD

Project Name	Interface Languages	Description	Advantages Disadvantages
Spark	- Scala - Java - Python - R - SQL	Spark uses a state-of-the-art directed acyclic graph scheduler along with a query optimizer. It also internally includes around 80 high-level mathematical operators (especially written in favor of building parallel applications) and many applicable libraries (e.g. SQL, MLlib, GraphX, Spark Streaming, etc.). All these operators and libraries can be combined seamlessly in every single application.	• Runs workloads in about 100× faster than Hadoop • Multiple compatible developing languages • Capable of arbitrary complex processing like machine learning • Suitable for real-time processing of batch data and streaming data • High memory requirement
Hadoop	- Java	Hadoop is a distributed batch-processor with 4 modules: Common (supports the other Hadoop modules), HDFS (distributed file system that provides data access), YARN (job scheduling and cluster management), and MapReduce (distributed parallel processing of large data sets).	• Suitable for real-time processing of batch data • High processing latency • Low encryption capabilities • High consumption of resources (i.e. IO and Memory)
Storm	- Java - Clojure	Storm is a distributed real-time computation system which is designed as a topology in the shape of a directed acyclic graph scheduler. It reliably processes unbounded streams of data, because it always schedules the streams in a resource-aware manner.	• Capable of arbitrary complex processing like machine learning • Stable integration with any database system • Suitable for real-time processing of streaming data • More dependency on system components
Flink	- Java - Scala	Flink is a distributed processing engine for state-based computations, which parallelizes applications (in any scale) into possibly thousands of tasks that are distributed and concurrently executed in clusters.	• Operable over unbounded (endless) data streams (e.g. IoUT data) • Suitable for real-time processing of pipelined streaming data • Possibility of latency over network communications

and maintain both operating system and the required software packages.

According to the aforementioned definitions, cloud computing is capable of processing BMD with the aid of all the above service models. Among them, PaaS strikes an attractive balance between convenience and cost, undertaking the management of the operating system, software packages, and hardware elements. It is worth mentioning that, Big Data as a Service (BDaaS), which has recently appeared in technical glossaries may indeed be classified as a specific form of PaaS, where the statistical analysis software packages are offered alongside the required databases and APIs.

On the other hand, IaaS constitutes a cost-effective solution, where the service providers are responsible for maintaining the underlying hardware. Some of the top PaaS and IaaS service providers are Amazon Web Services, Microsoft Azure, Google Compute Engine, DigitalOcean, Oracle Cloud Compute, Rackspace Cloud, IBM Cloud, Linode, HP Enterprise Converged Systems, Green Cloud Technologies, etc.

Despite the fact that SaaS providers tend to be user-oriented and application-specific, some of them also support underwater applications. For instance, marine image-annotation software are ready-to-use software products for semi-automatic annotation of videos and still-images. These software packages are comprehensively reviewed by Gomes-Pereira *et al.* [159], where 23 software products from more than 500 publications are summarized.

The semi-automatic solutions offered by marine image-annotation software are rarely applicable in real-time BMD analytics and they tend to rely on data analytic platforms as well as self-developed ML algorithms. As a result, every

practical use case of BMD, requires a distributed on-premise platform or cloud-based data processing software for glean-ing knowledge from BMD. Next, some of these IoUT data extraction applications are studied.

D. Marine Data Applications

The marine data collected by various sensors and devices of the IoUT ecosystem can be exploited by data processing platforms for compelling applications. These applications offer businesses benefits based on their own data [32]. In the following subsections, we discuss and evaluate some of these applications in the context of IoUT, including maritime applications, underwater localization, and marine life tracking.

1) *Coastal monitoring and GIS data:* By appropriately processing the available maritime sensory data, some secondary parameters can be collated with the aid of the IoUT. These include the accurate localization of marine vehicles, the provision of weather and climate data for specific oceanic locations, accessing biogeographic data such as the recognition, counting, and distribution of underwater species, etc. Hereafter, we will refer to all these secondary parameters as IoUT Geographic Data, because they are devoted to studying the physical environment, the inhabitants, locations, and things in the particular area of the sea.

Having accurate geographical data is essential in IoUT applications. For this type of IoUT data, there exist a number of open-source on-line geographical databases. These databases are mainly focused on maritime information systems, a particular type of Geographic Information System (GIS), which is indispensable in efficient international transportation [160].

Fortunately, most of these GISs can be readily merged with the IoUT infrastructure in support of maritime organizations with the aid of tracking and routing information, etc. They can also provide up-to-date access for researchers in order to monitor the global ecosystems.

Kalyvas *et al.* [31] surveyed more than 180 free real-world GISs having open access databases based on the most trustworthy on-line data sources. They have categorized the GISs into 19 classes, which are distilled into as few as four classes here for simplicity. These classes and their applications include:

- *IoUT tracking data*: For vessel monitoring, marine accidents, anti-shipping activities, and navigational aid systems;
- *Marine cartographic data*: For essential naval data (like borders), protected and sensitive areas, port status, port locations and facilities, coastline as well as land areas, and bathymetry maps;
- *Oceanic climate data*: For nautical weather, natural hazards, tides as well as eddies, and satellite imagery;
- *IoUT commerce data*: For shipping companies, flags of convenience, and marine conservation organizations.

An important category that could be added to the above-mentioned classes of Kalyvas *et al.* [31] is Biogeographical Data for addressing the geospatial distribution of underwater species. In order to define this new category, the open-access [OBIS](#) project (Ocean Biogeographic Information System), funded by UNESCO can be used. OBIS is connected to more than 500 databases in 56 countries and provides observation of 120,000 marine species down to 11,000 *m* depth, from bacteria to whales.

In addition to the aforementioned need for Biogeographical data, under the Marine Cartographic Data category detailed in [31], there is a missing sub-category for Georeferenced Locations. To elaborate a little further, georeferencing in GIS is a subcategory of navigational assistance that aims for precisely associating locations with their equivalent points on the physical maps, which is achieved by the [Marine Regions](#) project. This project is composed of a list of 55,000 georeferenced marine places, gathered from the [VLIMAR](#) Gazetteer and [MARBOUND](#) databases.

In addition to the GIS databases, there should also be a directory of applications, which require access to precise undersea locations. This is called underwater positioning. In the following sections, undersea localization of sensors and vehicles, along with the marine life tracking methods will be discussed in more details.

2) *Underwater localization*: A very useful data type that is included in IoUT is related to the positioning of undersea devices, systems, animal species, and data sources. This is even more critical when there is a need to geo-tag IoUT sensory and imagery data.

Underwater positioning is a challenging task, because the otherwise ubiquitous navigation signals of the GPS satellites do not penetrate seawater. Therefore, other underwater navigation methods should be used, including blind positioning relying on miscellaneous inertial sensors, acoustic transponders (with separate transmitters and receivers), ranging sonars (with

only a single standalone acoustic transceiver), image-based positioning (using cameras to localize), and Simultaneous Localization and Mapping (SLAM).

Due to the associated challenges including the long latencies, multipath fading, Doppler shifts, frequency limitations, sparse deployment of the nodes, and their high mobility in underwater networks, a single navigation technique will never offer a flawless performance [161], [162]. Therefore, all the vehicles, fixed stations, sinks, and nodes in IoUT applications usually combine some of these localization methods to achieve improved performance in underwater environments. Here, we briefly review the aforementioned localization techniques and provide insights concerning their advantages and limitations.

Blind positioning: which is also known as inertial navigation, is based on the knowledge of a device's relative orientation, acceleration, velocity, and gravity anomaly (i.e. difference between the observed gravity and the predicted value [163]). In this localization method, the underwater device has to determine its position automatically, i.e., without any positioning support from a ship or transponder. In this method, a combination of sensors is used to estimate the current location.

These sensors include a typical magnetic compass for direction detection, a pressure sensor for underwater depth estimation, a doppler velocity log for velocity measurement, a ring laser gyroscope or Micro Electro-Mechanical Systems (MEMS) gyroscope for angular velocity detection, and a pendulum or MEMS accelerometer for force and acceleration calculations [29].

However, these sophisticated blind positioning methods suffer from propagating errors. Even a small positioning error remains in the memory of the system and aggravates the future measurement errors, leading to an unbounded error propagation. Nevertheless, the method's estimation of the exact position can be improved by a variety of integrated sensors. Furthermore, blind positioning is a power-efficient method, compared to other positioning techniques [164]. Hence these positioning techniques contribute to almost all modern underwater positioning systems [161]–[165].

Acoustic transponders: This localization technique uses a transmitter and receiver pair for measuring the ToF of a ping signal in order to perform navigation. ToF [*sec*] is applicable both to the adjacent transmitter/receiver (i.e. where the transmitter and receiver are closely located) and separate transmitter/receiver (i.e. where transmitter and receiver are distantly located). It is also known as the time of arrival in separate transmitter/receiver scenarios [162]. By measuring the ToF and the acoustic wave speed, one can precisely measure the distance in [*m*]. This positioning method has diverse categories, including:

- *Acoustic array*: Similar to the concept of GPS satellites, and by using more than one beacon transponders, the system will be able to determine the position in any underwater location based on the phase-difference of the signals arriving at the transceivers. Short- as well as ultrashort-baseline [7] and long-baseline [166] techniques may be used in this category. In short- and ultrashort-baseline relying on ship-mounted transponders, the un-

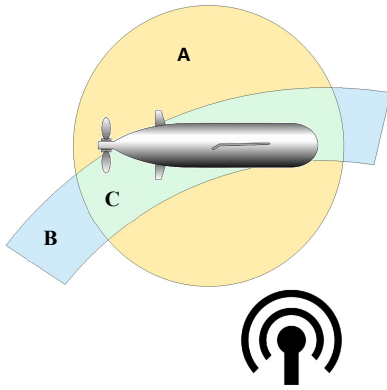


Fig. 9. Using a fixed beacon at known location to decrease the location-uncertainty of an AUV from 'A' (intersecting with 'B') to 'C'.

dersea system localizes itself relative to the ship floating at the surface; while in long-baseline that uses GPS-intelligence as well as dispersed buoys and beacons, the location can be determined.

- *Single fixed transponder:* The idea behind fixed beacon based positioning is shown in Fig. 9. The AUV in this picture has an uncertain prior knowledge about its position (region A). However, it knows its distance from a fixed beacon subject to a degree of uncertainty (region B). With the advent of combining these two pieces of information, the AUV finds its position subject to a reduced uncertainty (region C). Due to the fact that it uses only a single geo-referenced beacon, this positioning system is cheaper and easier to install than multi-beacon long-baseline techniques [167].
- *Cooperative navigation:* This method is also known as modem-based navigation, because in recent advanced positioning systems, modem transponders are also used to send beacons in support of navigation based on the ToF [168]. These modems do not have to be stationary, hence they can be installed on a moving vessel having a known geo-position or a swarm of underwater vehicles to communicate and to cooperatively localize each other.

Range sonars: Sonars are robust, well-established standalone acoustic transceivers that were originally used for imaging and ranging. The main ranging sonars used in underwater navigation and mapping include [29]:

- *Echo sounder:* A single narrow beam is used for determining the distance from obstacles [169] or from the seafloor.
- *Multi-beam:* Measures the ToF for each beam to assemble a bathymetric map [170].

Image-based positioning: This technique uses environmental images, taken either by monocular or stereoscopic cameras (or even by imaging sonars) for navigation. In this positioning method, substantial processing power is required for feature extraction and for processing in order to detect and identify key points, objects, and regions of interest.

The main idea behind the vision-based navigation, also known as visual odometry, is to capture images of the seabed and then to match subsequent images in order to navigate

[171], [172]. stereoscopic cameras have the advantage of capturing 3D transformations between consecutive image pairs.

It is plausible that errors can propagate and accumulate in the context of this technique. To address this problem and achieve a bounded positioning error, the SLAM tactic can be merged with image-based positioning for improved localization.

SLAM: Concurrent or Simultaneous Localization and Mapping is defined as the process in which an autonomous vehicle builds a map of a specific area and also localizes itself within it [29]. The mapping process of SLAM may rely on a variety of devices such as cameras, sonars, or inertial sensors, respectively, leading to vision-based, sonar-based, or sensor-based SLAM techniques [173], [174]. In all cases, the features of the sensed data or captured image are extracted. Then, based on those features, the position is detected and stored.

3) *Marine life tracking:* At first glance, underwater animal tracking seems to be nothing more than a memory-aided localization method in IoUT, which was the subject of the previous section. However, undersea animals may be quite small and they cannot carry relatively heavy inertial sensors, transponders, sonars, or cameras. Therefore, sophisticated new tracking methods have to be devised.

Additionally, as discussed in Section III-D2 and also bearing in mind the penetration depth formula of (1), the GPS signals having a frequency of 1.2 and 1.5 GHz cannot penetrate seawater beyond a few meters. Accordingly, alternative marine life tracking methods have to be implemented in IoUT applications, some of which are evaluated in this section. All these methods can make use of a data storage tag to archive data on a memory chip for future retrieval [175].

RFID tags: This tracking technology uses Radio Frequency Identification (RFID) patches and has a limited range of about 10 m in freshwater. However, it does not work well in seawater owing to its high salinity. RFID tags are equipped with a unique Identification code and need an external energy source in the form of a low frequency signal, in order to become activated and to retrieve data. These tags are also available in passive integrated transponder form, which are specific implantable RFID devices [4].

Acoustic tags: This tracking technology emits signals in the form of acoustic waves and has a reasonable transmission range both in fresh and seawater. In this method, a pinging sound with an embedded ID is periodically transmitted to an array of remote acoustic sensors (hydrophones). This ping is used to locate the animals and the ID is used to identify them [175].

Image-based tracking: Perhaps image-based tracking is one of the most advanced underwater tracking techniques reported to date. It uses images of a moving or fixed camera system to track, recognize, count, measure, and also study the animal's behaviors. In a fixed camera based system, usually a baited or trawled station is used for attracting intended species [140]. On the other hand, the tracking of animals in a moving camera based system uses cameras installed on a moving platform. The main challenge in these systems is to promptly process the large variety of images [21].

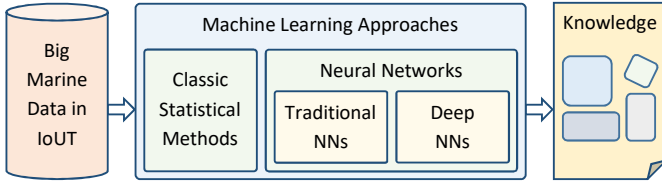


Fig. 10. Conceptual categorization of the main topics in Section IV.

E. Section Summary

In this section, we have studied the essential topics of big marine data. At the beginning, we introduced the five BMD system components, followed by further elaborations on the first component, i.e. on data acquisition. Thereat, we reviewed the underwater data gathering tools and techniques, along with the data aggregation protocols for more compactly representing the gathered data. We also studied data fusion methodologies, conceived for the fusion of data from distinct datasets into an integrated database, for the sake of having more generally representative information. Our discussions in this section were then continued by classifying BMD into 1D sensor signals, 2D image data, and 3D video streams. For all these categories, we surveyed the typical IoUT infrastructures and observatory systems, which freely offer their BMD for researchers. The visibility, contrast, color distortion, and light scattering in 2D underwater images as well as 3D video data gathering were also addressed. Then, different distributed as well as cloud-based BMD processing frameworks were introduced and a couple of essential BMD applications were studied. These applications covered the oceanic GIS, underwater localization, and marine life tracking, which all play essential roles in IoUT contexts.

All of the tools and methods that have been discussed in this section, including the sensory and imagery data sources, GIS-based ready-to-use data in IoUT, and underwater object tracking as well as geo-tagging methods will generate BMD, which has to be processed and analysed. In the following section, we will focus our attention on big data processing techniques. Thereby, we will review state-of-the-art technologies in data cleansing and data processing using ML techniques for underwater applications.

IV. MACHINE LEARNING FOR BMD ANALYTICS

As mentioned in Section III, *data processing* is one of the five critical system components in the chain of wealth-creation from data. To address the commercial and industrial demands, ML and its Deep Learning (DL) variant constitute promising solution. In this section, as visualized in Fig. 10, we focus our attention on the use of ML strategies including the classical ML approaches, as well as the more traditional and emerging NN-based learning methods for BMD processing.

A. Machine Learning Techniques

Considering the large amount of data generated both in IoT and IoUT, there is a growing need for powerful tools and techniques capable of distilling and visualizing knowledge

gleaned from data. These tools and techniques have evolved over the past century! As a benefit, ML is readily available for the analysis of BMD [33]. ML can be viewed as a collection of software algorithms that empower intelligent machines to improve their performance on an accomplishing pre-configured specific task. This particular task in turn, may be categorized under descriptive, predictive, and prescriptive models.

The ML approaches may be divided into supervised and unsupervised methods. The unsupervised methods are mainly used for data clustering, according to the features embedded in the data itself. These methods do not rely on an expert for labeling and for entering inputs. On the other hand, in supervised learning, data will be labeled, prior to its exploitation, by an expert. In this case, the resultant ML solution is referred to as a classifier, which is then involved for the ensuing data classification phase.

Classification methods are devised for categorizing data, representing for example a measurement from a sensor or a pixel in an image into one of the legitimate predefined output classes. When there are only two possible answers to a given question (i.e. yes or no), the classification problem is often termed as *detection*. In cases of three or more legitimate output classes associated with multiple choices, classification is also often referred to as recognition. Methodologically, classification algorithms can be divided into statistical methods (also known as conventional data analytics) and Neural Networks (NN). The neural-based methods may themselves be divided into traditional NNs and deep NNs [176].

All ML approaches are designed to assign a given input data (X) to a set of predefined classes (C_i). This is a well-known classification/clustering problem, which can be solved by either generative or discriminative models. The generative models are those which solve this problem by using the joint probability distribution function of [177],

$$i = \arg \max_C P(C)P(X|C). \quad (17)$$

By contrast, the family of discriminative models uses the conditional probability distribution function of [177],

$$i = \arg \max_C P(C|X). \quad (18)$$

According to the Bayes' theorem from a theoretical perspective, these methods are identical. In practice however, it is usually easier to calculate $P(C|X)$; especially when we have a large amount of data (i.e. X) for training our model. The conventional data analytic methods tend to use generative models, while most NNs rely on discriminative models [178].

In generative models, the behavior of both C and X should be known. However, for discriminative models, we directly deal with the unknown C , based on a given X . Discriminative models are generally simpler, faster, and have less parameters to adjust. Accordingly, while the family of conventional statistic approaches may be good enough for handling moderate-dimensional situations, traditional ML and modern DL based approaches (with discriminative models) are needed for processing big marine data problems (i.e. image,

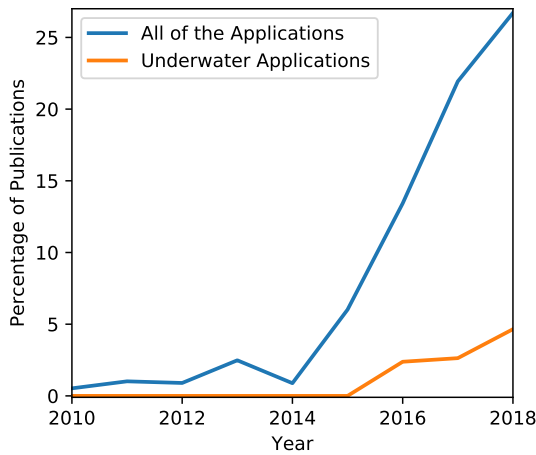


Fig. 11. Percentage of deep learning usage in all machine learning publications, as searched between IEEE Explore as well as Web of Science databases.

video, and other sensory information generated within the IoUT infrastructure).

The concept of DL is built around the idea that artificial neurons are capable of automatically extracting features and learn a pattern, provided that there are enough hidden layers and unweighted neurons in their networks. Based on this concept, deep networks have evolved from the traditional NNs by invoking more than one hidden layer (technically, dozens of layers). Using these extra layers, deep networks become capable of extracting features and reduce the learning dimension as it will be introduced later in Section IV-D.

The growth rate of DL usage in all ML publications in the post-2010 era is demonstrated in Fig. 11. This figure shows an astonishing factor 25 increase in the percentage of publications in as few as four years. These publications have substantially advanced the field by proposing new algorithms, networks, and strategies for improving the performance of deep networks. Table XI presents the most recognized supervised and unsupervised deep learning networks. All the tandem arrows and circular nodes in this table ($\rightarrow\bigcirc$) are representative of an artificial neuron processing the sum of weighted inputs and a subsequent activation function (e.g. Sigmoid, tanh, etc.). Please note that every NN has a bias input. These inputs are deliberately eliminated in the shown diagrams, for presentation simplicity.

Again, deep networks have several hidden layers. This is shown in Table XI by utilizing a dash-dot line ($-\cdot-\cdot-$), whenever the network is capable of incorporating multiple hidden layers. However, it should also be considered that, as these networks grow deeper, they also require more training data. Further details on various types of networks and on their advantages/disadvantages are provided in Table XI and will be discussed throughout this section. However, we study these networks mainly from an IoUT as well as BMD perspective and discuss what specific network types and algorithms would be particularly suitable both for BMD analysis and for IoUT applications. In doing so, we appraise many reported use cases of ML in underwater applications in Section IV-F, but before that, let us discuss the various software frameworks for

architectural deep network design, suitable for big marine data types.

B. Deep Learning Frameworks and Libraries

To facilitate the development of various DL architectures, such as those listed in Tables XI and XVI for different applications, numerous software frameworks and libraries have been developed for open access by the ever-growing deep learning community [184]. Table XII lists and compares a range of open-source DL tools and frameworks. Some of the abbreviations used in this table are Deep Learning for Java (DL4j), Microsoft Cognitive Toolkit (previously known as CNTK), and TensorFlow (TFlow).

All the frameworks seen in Table XII are capable of operating on NVidia® CUDA-supported GPUs relying on parallel processing. By contrast, OpenMP is another shared-memory based multiprocessing programming interface, which is only supported by some of the frameworks. As it will be discussed in Section IV-E, shared-memory based methods are critically important for speeding up the deep network's training process.

So far we have discussed various architectures capable of processing BMD and marine data. Once the data has been collected, some pre-processing steps have to be performed for ensuring that the data used to train the ML algorithm is clean and does not result in future learning problems. In the next section, this important issue is discussed in more details.

C. Marine Data Cleaning for Machine Learning

Data preparation and cleansing is a critical step in almost every ML project and it usually takes over half of the entire project duration to collect and clean the data. When dealing with the inaccessible harsh underwater environment of the IoUT, data cleansing becomes even more important. This is because the underwater environment is affected by many environmental factors, as shown in Fig. 4, which make it almost impossible to acquire clean data. Additionally, data sources tend to be quite vulnerable to the hostile underwater environment. Therefore, data preprocessing and cleaning, before training is crucial for the success of any ML aided BMD processing in IoUT. Below, various IoUT data types and their main preparatory considerations are discussed.

1) *Underwater sensor data cleaning*: Sensors are inevitable parts of every IoUT subsystem and are used in nearly all underwater exploration and surveillance applications [185]. They continuously measure physical, chemical, as well as biological parameters and generate a huge volume of data. Examples of these sensor and data are listed in Table II.

Data cleansing for sensory data sources in underwater BMD applications is typically concerned with missing values, contaminated measurements, and detecting outliers. There are a variety of techniques for these preparatory steps. To deal with a missing value in any sensor data, one can [186],

- Delete the entire record;
- Use a global constant;
- Use a statistical value (e.g. mean or median);
- Use an inter-class statistical value;
- Use the most likelihood (i.e. more probable) value;

TABLE XI
DEEP NETWORKS WITH INPUT, HIDDEN, MID-OUTPUT, AND OUTPUT LAYER NEURAL NODES TO EVALUATE STATIC, DYNAMIC, SEQUENTIAL, OR HIERARCHICAL INPUT DATA TYPES [179]–[183]

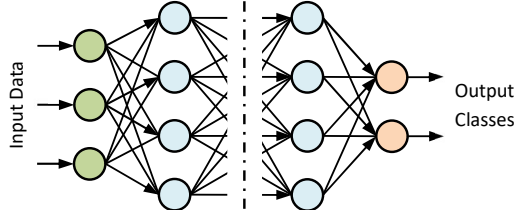
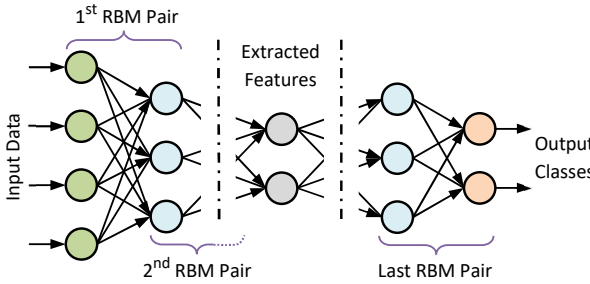
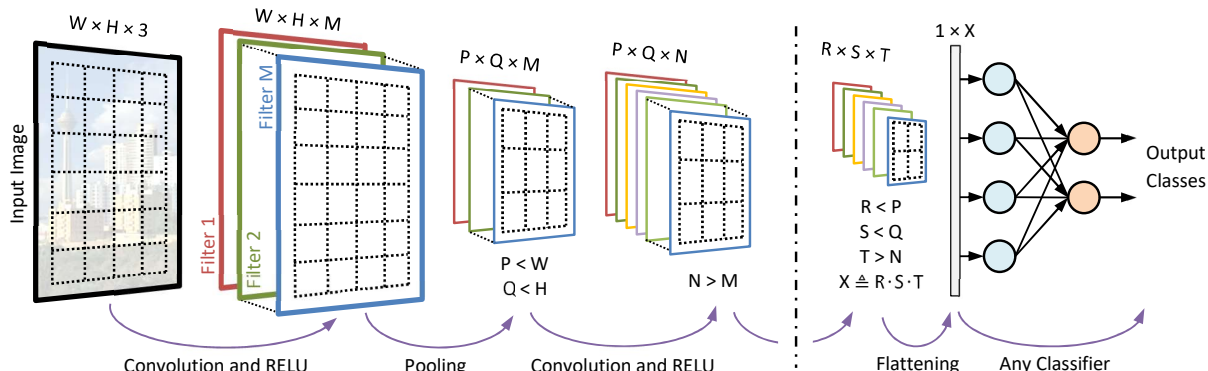
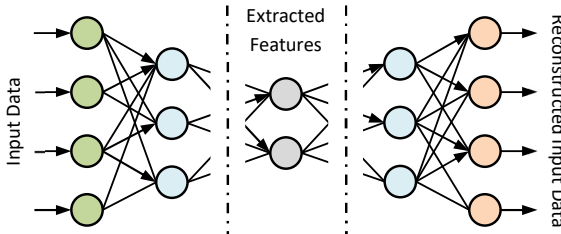
Fully-Connected (FC) or Dense Network		Description: This is a nonlinear multipurpose data classifier. The input data passes through the weighted vertexes and nonlinear activation functions, to the output layer, where its class will be decided. Pros: - It is widely used and well documented. Cons: - Gradient vanishing and gradient explosion (for unstable derivatives) and slow convergence during the BP training process are expected.
Deep Belief Network (DBN)		Description: DBN is structurally similar to FC networks, but they differ in their training. DBNs are trained by exploiting the concept of successive RBM pairs. Every layer in DBN is a visible RBM layer for the next neighbor and a hidden RBM layer for the previous one. This pairwise training is a beneficial substitution for the BP training scheme in FC networks. DBN as a supervised method can use labeled data to fine-tune the weights, after RBM initialization. Pros: - It overcomes the gradient vanishing and gradient explosion problems in BP method, by employing the RBM training scheme. - By using the Kullback-Leibler (KL) divergence metric, the information loss (or relative entropy) is considered as the cost function. - Can be used in both supervised and unsupervised learning. Cons: - It usually requires larger training time, especially when it undergoes a fine-tuning step.
Convolutional Neural Network (CNN)	 Description: CNN is probably the most popular data and image processing deep learning network by far. The input, for example an image of size $W \times H$ and 3 color channels, is partitioned with overlapping channels at first. Then a series of filters (or kernels) are convolved with each window to extract features. Despite the filters can be initialized by the kernels of Table XIV, but their weight might change during the train process. The filtered values are then guided to pass a nonlinear activation function (i.e. rectifier (RELU), $\tanh$, sigmoid, etc.). Afterward, the 1st and 2nd dimensions of the resultant matrix will be reduced in the pooling step. The number of consecutive convolution and pooling steps can be adjusted. Finally, the multi-dimensional matrix will be flattened to a 1-dimensional vector and will be passed to a classifier network (e.g. a FC network) to decide about the appropriate output class. Pros: - CNNs have a variety of architectures, many of which are widely used and well documented. - It has already been embedded in many software libraries and frameworks, making it available in high-level programming. - Less neural connections are required in CNN, because neurons are not fully connected. - Every neuron in the CNN learns a patch of the image. This contrasts with other methods (e.g. DBN) where every neuron has a contribution in learning the entire image. Cons: - CNN usually requires a lot of consecutive convolution and pooling layers to deal with a simple image. - It usually requires many labeled data to be appropriately trained.	
Autoencoder (AE) Network		Description: This network consists of two separate parts. The first part extracts the feature vector from the input data, while the second part reconstructs the same data, from the extracted features. After training the network, these 2 parts can be used concurrently or separately. Pros: - It offers a wide range of applications including data de-noising, dimensionality reduction, data compression and decompression, etc. Cons: - The network structure is robust and the feature vector size cannot be adjusted, after training process.

TABLE XI
(CONTINUED)

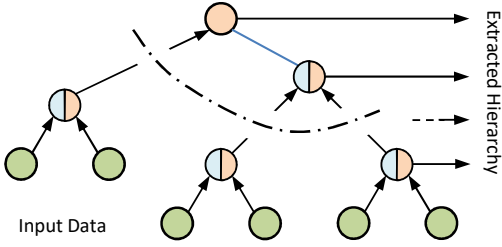
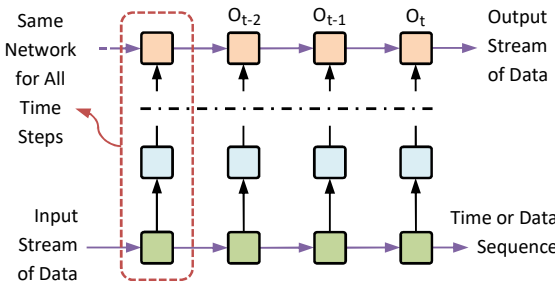
Recursive Neural Tensor Network (RNTN)	 Input Data Extracted Hierarchy	Description: The RNTN is a deep version of shallow autoregressive models. It is suitable to extract hierarchy in almost any sort of input data (i.e. syntactic parsing, image segmentation, etc.). It can also be used in sequential data analysis (e.g. sentiment analysis in text messages).	Pros: - The tree structure of RNTN is completely customizable and can be tailored to any application. - Any neural node of the network can also be used as an output. Cons: - RNTN is a memoryless network, which is not recommended in dynamic system evaluation.
Recurrent Neural Network (RNN)	 Same Network for All Time Steps Input Stream of Data Time or Data Sequence Output Stream of Data O_{t-2} O_{t-1} O_t	Description: RNN is a deep and deterministic version of the well-known Hidden Markov Model (HMM) of predicting any sequential data or time-series. In RNN, HMM layers are stacking on top of each other to create a deep network with intrinsic memory. The concept of point attractors in their state space acts as settling points to recall memories from the past.	Pros: - RNNs are very suitable in modeling complicated and nonlinear dynamic systems. - Their distributed hidden state (i.e. the intrinsic memory in hidden layers) helps them to effectively store information about the past. Cons: - They might show a degree of chaotic behavior. - RNNs are hard to train.

TABLE XII
COMPARING THE BEST OPEN-SOURCE SOFTWARE FRAMEWORKS AND LIBRARIES FOR DEEP LEARNING

Framework Name	Initial Author (License)	Initial Release	Operating Systems	Interface Languages	Parallelization Support		Deep Networks Support					Description
					GPU	OpenMP	FC/AE	DBN	CNN	RNN	RNTN	
TFflow	Google (Apache)	2015	Linux, Win, Mac, Android	Python, C++, Java, R	✓	✗	✓	✓	✓	✓	✗	- Most commonly used framework - Great documentation and community support
Keras	F. Chollet (MIT)	2015	Linux, Win, Mac, Android	Python, C++, Java, R	✓	✗	✓	✓	✓	✓	✗	- Operates on top of TFflow or Theano - Offers a high-level interface
CNTK	Microsoft (MIT)	2016	Linux, Win	Python, C++	✓	✓	✓	✗	✓	✓	✗	- High performance on big datasets - Easy training
DL4j	A. Gibson (Apache)	2016	Linux, Win, Mac, Android	Python, Java, Scala, Kotlin	✓	✓	✓	✓	✓	✓	✓	- Used in industrial big data applications - Has a rich set of deep networks support
MXNET	ASF (Apache)	2016	Linux, Win, Mac, iOS, Android	Python, C++, R, Scala	✓	✓	✓	✓	✓	✓	✗	- High efficiency and flexibility - Suitable for image classification
Neon	Nervana (Apache)	2016	Linux, iOS	Python	✓	✓	✓	✓	✓	✓	✗	- Optimized for Intel CPUs - One of the fastest frameworks
PyTorch	A. Paszke (BSD)	2016	Linux, Win, Mac	Python	✓	✓	✓	✓	✓	✓	✗	- Python version of the Torch framework - High-level object-oriented programming
Caffe2	Berkeley (BSD)	2017	Linux, Win, Mac, iOS, Android	Python	✓	✗	✓	✗	✓	✓	✗	- Efficient for CNN applications - Fast operation

- Use the most likelihood inter-class value.

Noise is another phenomenon affecting almost any sensory data. The process of de-noising a sensory measurement is termed as *noise cancellation*. Although no universally applicable de-noising methods exist, one should find the one meeting the SNR criteria. The SNR is a very common parameter of characterizing the performance of different noise cancellation

methods. Some of these methods include:

- Using a low pass filter [187];
- Using regression models [188];
- Applying a data binning method [189];
- Using wavelet methods in time-series [186].

Among these methods, low pass filter is the most common one and wavelet-based methods are the most complex ones

to implement. Additionally, data binning is not particularly popular for sensory data, but it is widely used for images as it will be discussed in the next section.

The final step in underwater sensory data cleansing is the outlier detection. This step can be readily carried out by considering three well-known criteria, named the IQR, T^2 , and Q criteria [186]. There are also some more advanced methodologies like, probabilistic models, clustering methods, distance-based detection, density-based detection, etc. [186].

2) *Underwater image and video data cleaning and quality enhancement*: As we have already discussed in Section II-B, underwater image acquisition suffers from strong absorption and scattering. Additionally, there are two other undersea signal degradation factors, namely chromatic aberrations [190]–[192] (i.e. color distortion or chromatic distortion) and noise imposed by both natural and artificial light sources [193]. These destructive factors significantly affect the quality of the captured video or image and should be mitigated by a data cleansing algorithm.

Any image and video preparation algorithm has to assess the image quality first. Automatic image quality evaluation algorithms assign an objective metric, which is a weighted sum of the image colorfulness, contrast, sharpness, etc. [194], [195]. This metric is then used as an objective function to be maximized by other image enhancement procedures. Therefore, underwater image and video quality assessment constitutes an essential step before image retrieval, image quality optimization, video compression, and other visual signal processing steps. It can also act as a guide for determining the data bandwidth required by the underwater communication algorithms, as well as by other decision-making processes [196].

After the automatic quality assessment of the underwater images and videos, software-based image preparation algorithms are used for enhancing the image quality. As a general rule, it is always cheaper to improve the image quality in software, instead of implementing bespoke high-cost imaging devices.

In a comprehensive review paper by Han *et al.* [135], the image preparation methods are divided into image dehazing and color enhancement. The authors then introduce and compare numerous methods for each category. However, they have missed the innovative method proposed by Ancuti *et al.* [197] that addresses both image dehazing and image enhancement at the same time. This method improves the global contrast that was degraded by light absorption, increases the edge sharpness impaired by light scattering, and exposes the dark regions with the aid of color balancing. The proposed method relies on a single-image camera-independent technique that can be applied to both photos and full-motion videos. Furthermore, it does not require any prior knowledge about the environmental conditions.

3) *Underwater image data binning*: In addition to image enhancement, image data binning is another data cleansing method that can be used in IoUT, which relies on grouping pixels into distinct partitions based on the similarity of their characteristics [189]. The goal of constructing these partitions is both to reduce the amount of noise and to resolve data

complexity. After binning, every pixel has a label to explicitly indicate its parent partition, based on its membership index. Although image binning is not an essential preparation step, it is recommended before invoking feature extraction for reducing the computational complexity and to speed up the ML process [189].

Image clustering methods, such as image binning, were developed much earlier than the state-of-the-art ML and DL algorithms and date back to the age of statistical mathematics. However, none of these methods are universally accepted and it is still a challenging task to select the most appropriate image partitioning method for a given application. A list of image clustering methods, that are beneficial in underwater applications is provided in Table XIII. These methods either tend to exploit image discontinuities (like edges) or similar regions to partition a given image.

In addition to the algorithms introduced in Table XIII, there are many others that have however been more rarely used in the literature. These methods are reviewed by Flake *et al.* [214] as well as Wang *et al.* [189] and might be worth investigating in future studies. Some of them are the Minimum Cut, Mean-Shift, Turbo-Pixels, Lattice Cut, Compact Super-Pixels, Constant Intensity Super-Pixels, Entropy Rate Super-Pixels, Homogeneous Super-Pixels, Topology Preserved Regular Super-Pixel, SEEDS, VCells, Depth-Adaptive Super-Pixels, Voxel Cloud Connectivity Segmentation, Structure Sensitive Super-Pixels, Saliency-based Super-Pixel, and Linear Spectral Clustering.

In addition to data cleansing, feature extraction is another essential step for any ML applications, which rely on statistical methods and traditional NNs. This substantial step in ML is discussed in the next section.

D. Feature Extraction for Marine Data Analytics

Feature extraction is a mathematical step, in which raw data is replaced by its numerical descriptors. This step is responsible for transforming large vectors of sensory data and large matrices of image data into their low-dimensional vector-based representatives. It is usually the most intricate part of almost any ML-aided computer vision problem and the solution should be tailored for the particular needs of the specific project at hand. Although all of the modern deep learning algorithms extract their own feature vectors automatically, these deep networks are hard to train, when relying on limited training datasets.

Finding a series of useful features is even harder in underwater image processing applications in high-dynamic environments subject to non-uniform light illumination, variable scene brightness, and degraded colors. Table XIV lists a number of salient descriptor routinely used in underwater applications. These features are categorized into four classes, including color, texture, shape (boundary), and other descriptors. The papers listed in Table XIV tend to use a series of descriptors, depending on their specific target applications. In one case, the number of descriptors used has been as high as 66 features in the Fish Recognition project carried out by Huang *et al.* [210].

After selecting a number of features, it is recommended to mathematically evaluate their correlation and then reduce the

TABLE XIII
IMAGE CLUSTERING METHODS AND ALGORITHMS IN UNDERWATER APPLICATIONS

Method (basis)	Algorithms	Reported Underwater Applications
Edge-detection (based on the 1 st and 2 nd order image derivatives)	• Watershed • Snakes (Active Contour Model) • Blob detection (i.e. Laplacian of the Gaussian, Determinant of Hessian, Difference of Gaussians, etc.)	• Jellyfish Detection by Watershed and Snakes [198] • Plankton Recognition by Snakes [199] • Deep Sea Megafauna Recognition by blob [200]
Thresholding (based on a threshold value (clip-level) in feature space)	• Otsu's Method (Maximum Variance) • Expectation-Maximization • Maximum Entropy Method • Color or Intensity Histogram Thresholding	• Fish Detection [201] • Jellyfish Detection by Expect.-Maximization [198] • Deep Sea Megafauna Recognition [202] • Plankton Recognition [25], [203]
Distance- or Region-based (based on the similarity in attributes)	• Region Growing (RG) • Simple Linear Iterative Clustering (SLIC) • Region Splitting and Merging • Morphological Spatial Pattern Analysis (MSPA) • Statistical Clustering Methods (K-Means, Subtractive Clustering, AGNES, DBSCAN) • Neural Clustering Methods (Self-Organizing Map (SOM), Hierarchical SOM, Hierarchically Growing Hyperbolic SOM) • Fuzzy Clustering Method (FCM) • Wavelet Transform • Compression (Texture and Boundary)	• Shrimp Detection by SLIC [23] • Plant Detection by Gabor Wavelet [20] • Plant Recognition by Gabor Wavelet [153] • Plankton Recognition by MSPA [24], [204] • Coral Detection by Clustering [205] • Object Detection by SLIC [18] • Object Detection by Gabor Wavelet [17] • Object Detection by Clustering [19] • General Underwater Clustering by FCM [206] • General Underwater Clustering by RG [207] • Coral Recognition by MSPA [134] • Coral Recognition by RG [208]
Math Calculus (based on solving a Differential or Integral Equation)	• Level Set Model • Mumford Shah Model • Chan-Vese Model	To the best of authors' knowledge, this method has never been used in underwater applications.
Graph Cut (based on the undirected graph partitioning (i.e. Markov Random Fields) to model the impact of pixel neighborhoods)	• Normalized Cuts • Lazy Random Walks • Isoperimetric Partitioning • Minimum Spanning Tree-based • GrabCut • Model Based	• Fish Recognition by GrabCut [209], [210]
Video Motion (based on the object movement)	• Subtracting a Pair of Images	• Fish Detection [211], [212] • Fish Recognition [139] • Lobster Detection [213]

number of features to the number of truly independent descriptors. This process is termed as *data dimensionality reduction* and can be performed by feature reduction techniques, feature selection, and feature aggregation [222].

Some of the popular dimensionality reduction techniques found in literature are Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), independent component analysis, non-negative matrix factorization, Self-Organizing Map (SOM), sequential forward search, sequential backward search, bag of words, etc. [222]–[226].

So far in this section, the software components suitable for ML and DL aided IoUT have been studied. However, these ML-aided solutions also rely on appropriate high-performance hardware platforms.

E. Hardware Platforms for ML in IoUT

The implementation of any ML solution, from its data cleansing and algorithm development to its final deployment, may rely on a variety of hardware platforms. These hardware platforms typically boost the overall throughput by parallel processing. These processing methods can be broadly divided into the following two main categories:

- *Shared-memory multi-processors*: They rely on multiple processors that all share a memory unit [227] and the

ML algorithm resides within this memory unit. Some of the more popular shared-memory methods are based on Application-Specific Integrated Circuits (ASIC) [228], Field-Programmable Gate Arrays (FPGA) [229], multi-purpose and multi-core CPUs, and Graphics Processing Units (GPU) [230].

- *System of networked processors*: As already introduced in Section III-C, a Distributed Computing System (DCS) is a system of networked processors, which coordinates the processors' actions by passing messages to each other [154], [231]. They can be used to parallelize ML algorithms in three different ways, including:

- *Data parallelization*: This is performed by running the same ML algorithm in all of the distributed computers and then dividing the data between them. Every computer estimates all parameters based on a separate dataset, before they exchange their estimates for formulating a final estimate. MapReduce is one of the most popular data parallelization methods [232], [233].
- *Model parallelization*: In this paradigm, the ML algorithm will have to be decomposed into different functions and operands. The algorithmic parts are then shared among multiple computers and every

TABLE XIV
DATA PROCESSING AND MACHINE LEARNING FEATURE-SETS IN
PUBLISHED UNDERWATER APPLICATIONS

Color Descriptors	Underwater Applications
• Image statistics (e.g. Hu's 7 invariant moments, affine moment invariants, Skewness, kurtosis, mean)	[23]–[25], [139], [150], [204], [208], [210], [215]
• Segment contrast	[23]
• Histogram descriptor	[19], [23], [134], [200], [209], [210], [216]
• Five MPEG7 color features	[200], [202]
• Transparency ratio	[203]
Texture Descriptors	
• Three MPEG7 texture features	[200], [202]
• Wavelet transform	[205]
• Gabor filter	[20], [134], [139], [153], [200], [210], [216]
• Filter banks (e.g. Schmid, maximum response, Leung and Malik, root filter set)	To the best of the authors' knowledge, these descriptors have never been used in underwater applications.
• Grey level co-occurrence matrices properties (contrast, correlation, energy, and homogeneity)	[22], [24], [25], [134], [139], [204], [210], [215], [217]
Shape Descriptors and Key Points	
• Hough transform	[27]
• HOG (Histograms of Oriented Gradients) based (e.g. SIFT [®] , SURF [®] , GLOH)	[17], [200], [209], [210], [218]
• Binary descriptors	[15], [16], [134], [219], [220]
• Weber local descriptor	[221]
• Convexity	[24], [150], [203]
• Fourier descriptors	[24], [25], [139], [204], [210]
• Frequency domain descriptors like phase congruency	[221]
• Curvature scale space transform	[139]
Other Descriptors	
• Granulometrics (size, area, and orientation) to recognize an already detected object	[22], [24], [25], [140], [203], [204], [210]
• Eigenvalues and covariance matrix to recognize an already detected object	[203]
• Motion related	[22]
• 3D shape, surfaces, and texture descriptors	[153]

part has its own specific set of parameters. The input data, however, will be simultaneously fed to all of them, where every computer is responsible for estimating the set of parameters assigned. During the parameter training process, those parallelized computers exchange their partial error vectors back and forth to meet the convergence criteria and to come up with the final parameter estimations [234], [235].

- *Pipelined parallelization*: In this method, the algorithm is shared among distributed computers, similar to the above-mentioned model parallelization method. These parallelized algorithm parts are chained together from one input block to one (or more) output blocks, termed as *pipelines*. In contrast

TABLE XV
COMPARING DIFFERENT PARALLEL PROCESSING HARDWARE PLATFORMS
TO CARRY OUT A TYPICAL MACHINE LEARNING ALGORITHM

Criteria	ASIC	FPGA	CPU	GPU	DCS
Inference Speed	++	++	--	+	–
Product Unit Cost	++	+	+	–	--
Design Complexity	--	–	++	+	–
Occupied Space	++	++	–	+ ^a	--
Research Friendly	--	–	+	++	+ ^b
Market Friendly	++	+	--	–	+ ^c

^a New GPU platforms such as NVIDIA Jetson occupy small spaces.

^{b, c} Cloud-based DCS platforms such as AWS EMR are readily available.

to model parallelization, data will also split into a series of records. At the beginning, the first record is passed to the first computer in the DCS, to carry out its own task (i.e. local parameter adjustment). The output of this computer is then relayed to the next computer to carry out its own task over the first data record. Meanwhile, the first computer starts processing the second data record. This pipe-lined process continues until the last record in the database exits the DCS and consequently, the local parameters of the last block in the pipeline are updated [236].

A rudimentary qualitative comparison of different parallel processing hardware platforms used for ML-aided and BMD processing is offered in Table XV. Here, the strength of each platform in terms of a specific criterion is indicated by the + and – signs, where more +s represents better performance in the context of that specific benchmark. The last two rows of this table summarize the entire table with respect to the associated research or industrial applications. For instance, as the Table shows, the CPUs require the lowest design time and impose the lowest design complexity, but they suffer from the lowest operational speed in all design and application phases, i.e. data cleansing, NN training, and final inference. Therefore, they may be used for research, but they are less suitable for industrial applications.

Perhaps the most prominent choice for typical scientific application would be the GPU [237]. The hardware and software resources required for GPUs are affordable and their speed is high enough to cover almost any application. Both the research-based as well as the industrial-scale employment of GPUs in underwater data processing and ML applications have been frequently reported in the literature. These applications range from obstacle detection and collision avoidance, to image-based SLAM localization, and even further to underwater object detection (e.g., coral detection) [230], [238].

However, the major problem with GPUs is their dependence on a bulky host computer. This has limited their implementation in low-power and lightweight IoUT platforms as well as underwater vehicles. To address this problem, new compact GPU designs have emerged to offer AI-ready computing resources. For example, [NVIDIA Jetson](#) is a standalone GPU-accelerated embedded system, which has a small volume. The

high performance, low power, and compact form-factor of the Jetson family makes them ideal for example for deep learning aided computer vision applications [238].

Since the Jetson embedded systems are empowered by the Linux Ubuntu operating system, they are eminently suitable for deep NN frameworks of Table XII, e.g., TensorFlow. Thus, the low-latency inference capability of machine learning algorithms can be readily realized in underwater vehicles and platforms [230].

Additionally, NVIDIA's Compute Unified Device Architecture (CUDA) library unleashes the GPU's parallel processing capabilities for applications other than machine vision. As previously mentioned in Section IV-B, CUDA is an API model that allows engineers to use a CUDA-enabled GPU for general purpose processing, this is also referred to in parlance as a *General Purpose GPU* (GPGPU). For instance, Pallayil *et al.* [239] have used a GPGPU for performing high-complexity real-time beamforming in their underwater acoustic phased array. They simply used the popular C Language in the Linux environment to harness the fast Fourier transform in the CUDA library, to implement their underwater frequency-domain beamformer.

ASICs and FPGAs are listed in Table XV, which have the lowest form-factor. They also offer a high throughput and high power efficiency, which makes them eminently suitable for industry-scale IoUT projects [240]. Given these compelling attributes, FPGAs are repeatedly featured in real-time and low-power underwater applications. For example, Karabchevsky *et al.* [240] have implemented a standalone FPGA architecture for noise suppression in underwater sonars. Their proposed signal processing implementation is claimed to overcome the sonar-based underwater visibility problems.

Following the above survey of various data processing methods, platforms, and their hardware realizations, diverse ML algorithms used for underwater applications will be discussed in the next section.

F. ML Techniques in BMD Applications

The advantages of ML techniques make them eminently suitable for most underwater applications. The categorical investigation of these techniques as well as their implementation in BMD applications will be carried out in this section.

1) *Deep NNs for static IoUT data:* Static data is exemplified by still images in contrast to full-motion video clips. In the context of Table XI, the fully-connected, Deep Belief Network (DBN) [179], CNN [180], Autoencoder [181], and Recursive Neural Tensor Network (RNTN) [182] networks are eminently suitable for static marine data processing. In the case of RNTNs, they are usually fed by a single static underwater image in every single data entry step. This image consists of multiple segments, and RNTN is supposed to determine the hierarchy of every segment inside the given image (e.g. background, coral, plant, fish, etc.). The double-colored nodes of RNTN in Table XI are hidden neurons that can also be designated by the network designer to be an output neuron, representing those expected hierarchical segments.

Among the static data processing networks seen in Table XI, Autoencoders are the only unsupervised clustering

NNs. Every Autoencoder consists of two parts, which can be used either separately or simultaneously. The first part processes a static input data such as an underwater image and extracts its features. The second part on the other hand, takes the feature vector and tries to reconstruct the input image again. If the inherent image features were adequately captured, the reconstructed input data appearing at the output will be similar to the input data itself. From an application-oriented perspective, one can use the first part of the Autoencoder for dimensionality reduction and data compression, while the second part is suitable for data decompression. Meanwhile, by using both parts simultaneously together, Autoencoder will act like a noise reducing NN.

Another network architecture in Table XI is the family of DBNs. These networks consist of consecutive shallow learning Restricted Boltzmann Machine (RBM) pairs, which gradually reduces the dimension of input data from the top-level of the entire search/classification towards the final unique classifier. Any mid-layer in DBN may act as output layer of the feature extraction. Again, by continuously decreasing the feature vector size, the procedure culminates by unambiguously classifying the input data. Therefore, DBN can act as an unsupervised clustering NN, provided that it is terminated somewhere at a mid-layer. By contrast, it can also act as a supervised classifier, if the number of nodes is reduced to the specific number of classes and if labeled data is used to train the network.

The final deep network for static data that will be discussed here is the convolutional neural network. This network relies on multiple stages of convolution and pooling, as seen in Table XI. It is considered to be the best deep classification method, especially when dealing with static images or previously recorded outputs of sensors and of hydrophones. Several architectures of this popular network have been designed for scientific use, some of which are also supported by Google and Microsoft (as exemplified by Inception [241] and ResNet [242]). Several of these CNN architectures are compared in Table XVI, based on the excellent review article by Canziani *et al.* [243]. The Top-1 accuracy² listed in this table is measured with the aid of the single central-crop sampling technique³ of [244] for all of the networks.

The number of network parameters in Table XVI is proportional both to their memory footprint and to their required training time. On the other hand, the number of operations required for a single forward pass, as shown in the table is capable of indicating the overall inference speed of the network. The lower the number of operations, the higher the inference speed. Here, ENet is not a CNN, but it is directly inspired by CNNs. By the same token, Network in Network (NIN) is not a CNN, but it relies on convolution operations. Some of these architectures are also included in the deep learning libraries of Section IV-B and can be readily involved

²Classifiers usually assign a probability value to all of their output classes. Thereafter, the class with the highest probability (top-1) will be considered as the final answer, which is not always true. Sometimes the correct answer is among the top-N classes. Using top-5 accuracy is common, when comparing different classifiers.

³A given image may have four corner crops and one central crop.

TABLE XVI
COMPARISON OF A SELECTED NUMBER OF CNN ARCHITECTURES
SUBMITTED TO THE ANNUAL IMAGENET CHALLENGE [243], [244]

Architecture Name	Top-1 Accuracy	No. of Parameters	No. of Operations
AlexNet [245]	> 56%	≈ 50 M	≈ 2 G
NIN [246]	> 62%	≈ 6 M	≈ 3 G
VGGNet-19 [247]	> 70%	≈ 150 M	≈ 39 G
Inception-v4 [241]	> 80%	≈ 35 M	≈ 18 G
ResNet-152 [242]	> 77%	≈ 50 M	≈ 23 G
ENet [248]	> 67%	≈ 5 M	≈ 2 G

for any project. Let us now extend our discussions to cover deep NNs in dynamic systems.

2) *Deep NNs for dynamic IoUT data:* Recurrent Neural Network (RNN) [183], [249] and its variants (i.e. long short-term memory, gated recurrent unit, etc.) constitute the only deep NN architecture in Table XI that can be used for nonlinear dynamic systems, as exemplified by continuous underwater sensor outputs. RNNs are capable of using both time-series and sequential data streams to construct supervised classifiers.

RNNs constitute a deep version of Hidden Markov Models (HMMs) [249], which represent a simple form of the broader family of dynamic Bayesian networks. Markov models (i.e. Markov chains) are stochastic models, in which the output of a NN in response to the current state, only depends on the output of certain selected neurons in the previous state. The values of all neurons in all states are visible and are considered as the outputs.

Hidden Markov models are similar to Markov models, but they rely on non-observable or hidden neurons. In these networks, the visible output is directly calculated by applying a nonlinear function to the output of the hidden neurons. As illustrated in Table XI, an RNN is constructed by stacking HMMs on top of each other. The rectangular shape of the nodes in this picture indicates that in contrast to the circles of the other networks, these nodes are not constituted by a single neuron, but rather they are a combination of neurons in the form of a HMM. After studying the DNNs suitable for static and dynamic data analysis, let us study their capabilities to solve real-life problems in marine environment.

3) *ML solutions in underwater image applications:* The concepts of image and video quality assessment as well as image restoration were studied previously in Section IV-C2. We have also stated the fact that underwater imaging applications heavily rely on image enhancement algorithms to cope with the destructive effects of visible-light environments. To address this problem, deep learning techniques have been employed for enhancing images suffering from noise, absorption, scattering, and color distortion effects [190], [250], [251]. Furthermore, the benefits of DNNs in underwater image quality assessment can be assessed in future studies with reference to their overwater counterparts [252]. These applications, while in their infancy, are rapidly expanding.

Other applications that can benefit from deep learning are

underwater image clustering and binning. We briefly reviewed these concepts in Section IV-C3, noting that some underwater clustering algorithms are also introduced in Table XIII. By the way, using novel deep learning techniques in underwater image clustering has not as yet attracted the attention it deserves. For instance, one can beneficially exploit the embedded image clustering technique of [253] as an unsupervised DNN methodology in underwater image segmentation.

Additionally, using ML techniques in underwater object recognition have been previously used for various underwater applications and have shown different performance. Here, we provide a summary of these techniques applied to some common underwater object of interest recognition and compare their performance. In order to have a fair comparison, the Average Precision (AP) parameter, which is a widely accepted performance measure, can be used. The average precision in any statistical or ML-based classifier can be calculated as [14],

$$AP \triangleq \frac{1}{N_C} \sum_{i=1}^{N_C} \frac{TP_i}{TP_i + FP_i}, \quad (19)$$

where the True Positive (TP) and False Positive (FP) values are calculated for N_C number of classes. The AP parameter of many ML techniques published in the underwater plant, fish, and coral recognition literature are compared in Table XVII. All the algorithms referenced in this table are based on the well-known Fish4Knowledge and EILAT datasets introduced in Table IX and are also based on the video footages recorded by AUV and ROV vehicles.

It is quite common in ML to cascade different algorithms into a single method to attain an improved performance (e.g. decision tree with SVM, and CNN with fully-connected). As an instance of algorithm merging in underwater applications, Faillettaz *et al.* [25] as well as Hu and Davis [204] cascaded a fully-connected classifier over shape-based feature-sets with an SVM classifier over texture-based feature-sets. By combining the results from these two classifiers, they claimed to have achieved an improved average precision, as defined in (19).

Another noteworthy cascaded solution was conceived by Schoening *et al.* [202], where they employ multiple cascaded SVM binary detectors to construct a deep sea megafauna recognizer. The binary SVM detectors have object-specific operation, which makes them more accurate. Accordingly, by combining these binary detectors, high precision multi-object recognition was achieved. However, combining those object-specific binary detectors requires more hardware resources than a single classifier.

Finally, it is worth mentioning that underwater imaging applications are not limited at all to the visible light domain. For example, undersea sonar imagery can provide high-resolution images of the seabed, even in turbid water with low visibility. Some of the reported applications of sonar images include ocean mapping, mine-countermeasures, oil prospecting, and underwater search and rescue (e.g. finding the drowned corpses, wrecks, and airplanes) [256]. In this regard, relying on human operators in sonar-based underwater object recognition applications is not recommended, since they will experience fatigue by staring at the display screen, and

TABLE XVII
COMPARING THE PRECISION OF MULTIPLE MACHINE LEARNING TECHNIQUES TO RECOGNIZE THREE DIFFERENT UNDERWATER SPECIES

Application	Fish Recognition											
Dataset	Fish4Knowledge Dataset [14], [15], [209], [210], [254]											
Methods	Regression Tree	LeNet	Linear Discriminant Analysis and SVM	Flat SVM	BEOTR	Decision Tree and SVM	Softmax	K-Nearest Neighbors	DeepFish	GoogLeNet	VGGNet	DeepFish and SVM
Average Precision (%)	75.4	76.8	80.1	82.9	84.5	84.6	87.6	89.8	90.1	93.9	96.6	98.2
Application	Plant Recognition						Coral Recognition					
Dataset	ROV and AUV Video Footages [215], [219]						EILAT Dataset [134], [255]					
Methods	Self-Organizing Map (SOM)	Learning Vector Quantization	Decision Tree	K-Nearest Neighbors	Random Forest	Fully-Connected	SVM	Decision Tree	Bayes	K-Nearest Neighbors	Random Forest	Fully-Connected
Average Precision (%)	93.8	94.1	95.1	95.3	95.8	96.1	96.1	69.6	82.1	83.3	84.6	85.2

TABLE XVIII
COMPARING THE PRECISION OF MULTIPLE MACHINE LEARNING TECHNIQUES TO RECOGNIZE OBJECTS IN SONAR IMAGERY

Application	Object Recognition				
Dataset	Echoscope Sonar Image Dataset [257]				
Methods	K-Nearest Neighbors	Multilayer Perceptron	SVM	CNN (AlexNet)	CNN (LeNet)
Average Precision (%)	72.0	89.3	92.7	94.1	97.0

they might consequently miss the object of interest. Therefore, intelligent image recognition methods can be trained to replace human operators, especially in long-duration search scenarios.

To highlight the accuracy of machine learning techniques in underwater sonar-based object recognition, the AP metric of (19) is used here in Table XVIII. This table compares the performance of deep NNs to that of several statistical methods as well as to that of traditional NNs. While the statistical and traditional networks seen in this table are trained using HOG features from Table XIV, CNN deep networks do not need any feature extraction based preprocessing. Additionally, as seen in this table, appropriately designed and well-trained DNNs may outperform traditional models.

4) *ML solutions in underwater video applications*: As already discussed in Section IV-E, the capability of hardware platforms to train and analyze DNNs was impressively improved recently. This improvement has attracted increasing attention to the subject area of real-time IoUT video applications. It was also pointed out in Section III-B3 that the rapid growth in underwater video data volumes will require the development of automatic video processing, which can be carried out by machine learning techniques. These automatic

solutions will be used in a variety of underwater video data processing applications such as:

- *Visible light video tracking*: This application is designed for scanning video sequences to follow a specific element of interest [152]. A pair of basic problems in a video object tracking solution is how to predict the location of a moving element in the next frame and how to detect the element within this predicted region. Both of these be handled by deep NNs [140]. This application was also discussed in more detail in Section III-D.
- *Sonar video tracking*: Sonar videos are recorded with the aid of acoustic waves, instead of visible light. Within turbid undersea environments, sonar systems offer a significant advantage by supporting long-range and low data-rate imaging [141]. Similar to visible light video object tracking, sonar-based video object tracking can also be carried out with the aid of deep NNs [258].
- *Underwater photo mosaicing*: This is the act of combining separate visible light or sonar video frames, for capturing a wider perspective of the region of interest. Machine learning based photo mosaicing of underwater images is now routinely performed by state-of-the-art underwater vehicles both for exploration and for navigation [20], [29], [259]. Automatic localization and positioning of submersible vehicles with the aid of photo mosaicing is termed as SLAM [174] and it is studied in Section III-D.
- *Marine life studies*: The analysis of underwater species is an indispensable part of any observatory video system. Thanks to the wide availability of machine learning based data processing toolsets, marine biologists are now capable of analyzing the high-volume video data captured for extracting the desired information. Scientific studies based on marine life video data applications have been published in different areas, including underwater species

TABLE XIX
COMPARING THE MEAN SQUARED ERROR OF ML TECHNIQUES IN SEA
SURFACE TEMPERATURE PREDICTION AT PHILIPPINE SEA

Application	Sensor Timeseries Forecasting				
Dataset	NOAA Dataset in Table VIII				
Methods	SVM [267]	CFCC- LSTM [267]	SVR [268]	FC- LSTM [268]	GRU-ED [268]
One Day Forecasting	0.434	0.166	0.095	0.061	0.063
One Week Forecasting	N/A	N/A	0.214	0.168	0.162
One Month Forecasting	1.478	1.145	0.212	0.343	0.207

behavior understanding (ethology) [139], [213], [260], abundance and counting [22], [150], [199], [261], size measurement [150], detection and recognition [14], [15], [199], [210], [212], [261], and tracking [21], [140], [198], [199]. We have covered some of these aspects of marine life analysis in Section III-D.

5) *ML solutions in underwater sensor applications:* Despite the unique benefits of applying RNNs to the processing of sensor outputs and in nonlinear dynamic systems control, there is a paucity of reported use cases in real-world IoUT applications. Their underwater applications are limited to a few scenarios, such as IoUT sensor data forecasting [262], underwater vehicle sensor read and fault diagnosis [263], and the dynamic control of underwater movements [264].

Among these applications, RNN-based predictive models conceived for IoUT sensory data forecasting as well as for missed sensory data implantation are better investigated [265]. More specifically, sea water temperature and salinity predictions are claimed to be important, because:

- Water temperature and salinity have a direct effect on the acoustic communications between IoUT nodes [266],
- Oceanic temperature has a substantial impact on both the land and the marine ecosystems by regulating the global climate [262].

Hence, the employment of machine learning techniques to design accurate predictive models is promising in BMD. For example, the influence of Long Short-Term Memory (LSTM) [267] and Gated Recurrent Unit (GRU) [268], as a pair of common variants of RNNs on highly accurate water temperature prediction is presented in Table XIX. Even though sea surface temperature forecasting is challenging due to the influence of numerous complex and nonlinear thermodynamic factors, data-driven DNNs are capable of learning these dynamic behaviors. By comparing RNN-powered networks to traditional machine learning solutions in Table XIX, the efficiency benefit of DNNs in IoUT sensor data prediction becomes explicit.

To recap, the current usage of RNN and its variants in IoUT applications are rare. Nonetheless, their versatile applications in IoT in smart homes [269], smart cities [270], weather forecasting [271], and other areas promise a similar growth for IoUT applications in the near future.

G. Section Summary

In this section, the employment of ML-based techniques in BMD processing was studied. We started this section by

the definition of machine learning, classifying it into: classic statistical methods, traditional neural networks, and modern (i.e. deep) neural networks. The associated chronological perspective was also presented. Then, we discussed NNs by surveying the major deep networks in the literature and their potential applications both in static and dynamic underwater data processing. Despite the rapid development of sophisticated, but complex deep NNs, light-weight traditional NNs might remain typical in limited-complexity underwater applications. Therefore, more detailed discussions were provided in this section to cover data cleaning and feature extraction techniques in BMD. We then surveyed the available software frameworks and hardware platforms, including a collection of freely available libraries and frameworks. We also provided a comparison of the hardware infrastructures suitable for the software products discussed. Finally, the average precision and accuracy of diverse machine learning approaches suitable for underwater applications was studied.

V. CHALLENGES AND FUTURE DIRECTIONS IN IOUT AND BMD

Having reviewed the state-of-the-art research in the areas of IoUT, BMD, and machine intelligence, the challenges and opportunities in these growing fields will be discussed. We also propose solutions and future research directions to address the challenges and to pursue the opportunities.

One of the main obstacles that has been hindering further advances in the IoUT domain is that well-known terrestrial technologies, which perform well in the IoT domain, tend to be unsuitable in underwater applications. Many issues in the oversea application domain can be readily solved, easily, while they pose a significant challenge in underwater scenarios. Below we will continue with a list of challenges, opportunities, and future trends in the IoUT, BMD, and ML fields. Some of these challenges may be mitigated with the aid of big data processing and analytics, while some others require research efforts from the broader engineering community, hardware vendors, and policy makers.

A. Underwater Network Management System

Due to the significant growth in the number of Internet-connected underwater devices, the IoUT infrastructure tends to exhibit increased complexity. Consequently, improved Network Management Systems (NMS) are required, which represent the process of monitoring and controlling every aspects of the underlying network, for ensuring its seamless operation [272]. The monitoring must be automatic and prompt in locating, measuring, and reporting faults. Additionally, their control should be capable of efficient and reliable resource allocation or troubleshooting [38].

While the concepts of network management in IoUT are somewhat similar to those in IoT, the methodologies are different, as discussed in Section II-E owing to the differences between overwater and underwater communication channels. Hence, the extension of NMS in IoT to Underwater NMS (U-NMS) in IoUT requires further research [273] for each of the six aspects of U-NMS in the FCAPSC model, namely

the Fault-, Configuration-, Account-, Performance-, Security-, and Constraint-managements [273]. It is worth mentioning that FCAPSC of IoUT was derived from the original five elements in FCAPS of IoT, which was introduced by the International Organization for Standardization (ISO) in the 1990s. The constraint-management element requires U-NMS to deliver continuous connectivity even in the face of the hostile underwater channel, node mobility, device fragility, environmental dynamics, and technological heterogeneity.

To facilitate the implementation of the FCAPSC model, U-NMS protocols tend to divide their influence domains into the family of network functionalities (e.g. routing management, protocol assignment, security checks, etc.) and of device operations (e.g. UWSN maintenance, energy conservation, device positioning, time synchronization, etc.) [38]. The relevant studies around these domains were comprehensively surveyed in Section II. To the best of our knowledge, the Underwater Simple Network Management Protocol (U-SNMP) is the only U-NMS protocol, which covers the first domain of influence (i.e. the network functionalities). U-SNMP in IoUT is again, an extension of SNMP in IoT, and it is a manager-agent-based protocol, which is used in communication between devices [273]. On the other hand, Lightweight Machine to Machine (LWM2M) is an IoT protocol that covers the second domain of influence (i.e. the device operations). LWM2M is applicable to IoUT as well, subject to some modest adjustments [38].

Despite its popularity, U-SNMP lacks facilities for network configuration, and LWM2M suffers from challenges owing to the associated heterogeneous network support [38]. By contrast, there are other network management protocols in IoT that offer better performance. Future studies might consider appropriately adopting those overwater protocols to underwater applications. For example, the Common Management Information Protocol (CMIP) offers better security features and it is suitable for wide area networks [274]. Another alternative might be the LoWPAN Network Management Protocol (LNMP) in IoT, which has low data rates, low power consumption, low cost, and supports flexible topologies [275].

While the above protocols are better suited for distributed network managements, the centralized SDN management techniques of Section II-I are also in need for underwater management systems. Here, the Open vSwitch Database (OVSDB) protocol at overwater SDNs constitutes a promising base for designing its underwater counterpart. However, any attempt to adopt OVSDB to underwater SDNs requires further research for addressing its current security issues [276].

Even after dividing the U-NMS responsibilities into two major domains of influence and limiting the U-NMS protocols correspondingly, the number of challenges in each domain will remain significant [38]. This is a direct consequence of the broad nature of the U-NMS topic itself. To better understand these challenges and study their future directions, we have discussed each challenge in a dedicated subsection. The following subsections will cover multiple aspects of U-NMS, ranging from energy conservation to device maintenance, to its security issues and communications. Some of these subsections discuss using BMD analytics as well as powerful deep learning techniques.

B. Energy Conservation and Harvesting in IoUT Devices

Energy conservation and harvesting in IoUT devices are of prime concern in almost all underwater applications, while they can be readily addressed in overwater scenarios, where energy can be harvested from the sun and preserved in the system [277]. Some innovative methods of gleaning energy in underwater environments include:

- *Overwater solar energy*: As already discussed in Section II-J, by performing computations in above-water edge-devices, such as surface-floated buoys and vehicles, solar energy can be harvested. Indeed, solar powered buoys are amongst the oldest methods of environmental energy harvesting techniques [278], [279].
- *Underwater solar energy*: It was stated in Section II-B that light is strongly absorbed in water and it additionally suffers from color distortion. However, as mentioned in Section III-B2, blue light will penetrate water deeper than other visible light frequencies. Some previous studies suggested the use of solar cells to harvest the blue frequency band of the solar energy underwater. These photovoltaics are claimed to provide useful power at the depth of more than 9 m [280]. However, many underwater applications are invoked at depths well beyond the light penetration domain.
- *Tidal-wave energy*: Harvesting kinetic energy from waves in the littoral tidal basin can be readily achieved by using today's technology. This can also offer a source of energy in the underwater benthic zone. These systems are based on piezoelectric elements and the energy generated is high enough to power UWSNs and their devices [281]. Similarly, some contributions report the employment of the same technique to harvest energy from fluid-flow in pipelines [282].
- *Wireless energy transfer*: This method was introduced for RFIDs and acoustic tags in Section III-D3. Additionally, a remotely powered acoustic UWSN was reported in [283] as another energy acquisition alternative for the IoUT. Thereby, sensor nodes harvest the mechanical wave power supplied by an external acoustic source. Another use case of wireless energy transfer may rely on in-situ magnetic charging stations, where underwater vehicles (e.g. AUV, ROV, etc.) can use these stations to recharge their batteries [284].
- *Wired energy transfer*: In a clear contrast to the previous items, every close-to-shore IoUT application may rely on energy transmission through a cabled network. In this method, the energy arrive from a solar, wind, or urban power network and it is directly transferred to the UWSN. These systems, however, are costly because of the cabled infrastructures on both land and at sea [9].

The above methods have the potential to be used in marine type projects. They can extend the lifetime of IoUT networks and boost their QoS. However, except for the *wired energy transfer* method, they tend to be unpredictable, hence none of them guarantees the uninterrupted delivery of energy. To address this issue, the following pair of solutions may assist:

- *Using rechargeable batteries*: All the aforementioned

methods can be accompanied by tandem batteries to store energy. These batteries are recharged during the instances of energy acquisition and deliver their stored energy afterwards. This solution obviously prolongs the sensor network's lifetime, but requires maintenance and increases the cost of the system [9].

- *Managing energy consumption:* Just like any other electrical grid worldwide, the IoUT has to manage its energy demands by optimizing the power allocation to guarantee uninterrupted data collection and transmission [285]. Indeed, all components of the IoUT should be energy-aware. In this context, an innovative energy-aware robot was proposed by Wu *et al.* [286], which had the shape of a killer-whale. This robot has had reduced energy consumption for its propulsion as a benefit of its excellent lift-to-drag ratio, which is important for effortless gliding in water. Compared to the Seaglidors introduced in Table VII, the controllable flukes of this robot offer substantial energy savings, better maneuverability as well as enhanced endurance.

C. Development of Low-cost and Affordable Sensors

The underwater sensors listed in Table II and multi-sensor buoys are usually very costly compared to their overwater counterparts. To address this multi-disciplinary challenge, the following approaches can be considered:

- *Quality vs. cost trade-off:* Low-cost sensing devices having lower precision measurements could be purchased to strike a quality vs. cost trade-off.
- *Transferring specimen for inland assessment:* The establishment and operation of an underwater in-situ sensing unit is generally more expensive than a laboratory-based experiment. Therefore, whenever possible, samples could be transferred to an inland lab, to avoid the need for costly in-situ processing and evaluation.
- *Inferential measurements:* It is a common technique in industrial instrumentation to estimate a parameter from the values of other parameters, which are easier to measure. For example, calculating the total amount of ions dissolved in water is always easier by measuring its electrical resistance instead of utilizing costly electrochemical sensors [287].

In addition to the above-mentioned general recommendations, designing specific cost-effective sensors can help the evolution of IoUT technology. In order to design such sensors, scientists from different research backgrounds have to cooperate. The result of this cooperation will be *ad hoc* solutions, which are tailored to the predefined need of any project. For instance, here we list several contributions involving low-cost underwater sensors.

- Islam *et al.* [288] proposed a low-profile and low-cost microstrip patch antenna to measure the salinity of water. They found that the antenna's reflection coefficient is proportional to the amount of salt or sugar dissolved in water.
- Vorathin *et al.* [289] constructed a high-resolution hydrostatic pressure and depth sensor by attaching a fiber

Bragg grating on a rubber diaphragm. Their sensor is claimed to enhance the sensitivity and to compensate the temperature effects.

- Wang *et al.* [290] conceived a low-cost turbidity sensor, based on the 90° scattered light detection principles. To elaborate, they used off-the-shelf infrared LEDs having controlled light emission to construct a low-cost, yet accurate product.
- Kirkey *et al.* [291] proposed an inexpensive fluorometer based on an optical backscatter transducer. Explicitly, their idea is to use low-frequency circuitry for modulating the light source. Using this technique, their product will be very cost-effective.
- By relying on the fundamental concepts of Time-Domain Reflectometry (TDR), Time-Domain Transmissometry (TDT), or Fiber Bragg Grating (FBG) in optical physics, optical fibers can act as a sensor to detect a wide range of underwater physical parameters. Examples include leak detection in pipelines and estimating its location [292], stress response of the offshore platforms (i.e. legs of the jacket structure) and detecting its deformation [293], bending moments of the flexible risers in a hang-off position to avoid exceeding its absolute maximum ratings [293], temperature and pressure measurement as well as eliminating the temperature-pressure cross-sensitivity [294], etc.

In addition to the above techniques of reducing the sensors costs, reducing the physical size of underwater sensors also tends to reduce their production cost. Examples of low-cost smaller sensors include miniature underwater robots [295] and low-power nano-sensors [128].

Furthermore, using the edge computing capabilities discussed in Section II-J, it is possible to establish a laboratory on board of an oceanic exploration platform [37]. This limited-capability on-board lab may conduct preliminary experiments on samples, before transmitting the numerical results through the web. Additionally, by using a video-empowered command and control system, there would be no need for an expert to be present in-situ. This laboratory on the edge will also eliminate the need for sending the specimen to an inland lab for evaluation, which would be both expensive and time-consuming.

D. Large-scale IoUT Underwater Communications

In terrestrial telecommunication, electromagnetic waves, copper cables, and optical fibers are the mainstream transmission media. However, as studied in Sections II-B and II-D5, fiber-optics are expensive to deploy and maintain under water, hence typically acoustic, electromagnetic, and optical technologies are deployed, which do not propagate well, hence making IoUT telecommunication challenging.

Electromagnetic and optical technologies only cover short communication distances and are therefore unsuitable for long-range IoUT communications. Acoustic technologies tend to be more amenable to long-range IoUT networks, but they have a narrow frequency bandwidth and are prone to cross-talk with other local acoustic applications [296]. Hence they are also unsuitable for large-scale networks.

A promising, but costly technique of addressing the communication challenge in IoUT is to combine heterogeneous communication technologies [110]. However, the design of multi-technology multi-mode gateways for undersea applications is a challenging task, especially when considering the energy harvesting difficulties. These gateways may be combined with the SDN and cognitive radio concepts for efficiently sharing the limited spectrum undersea [110]. Future research should conceive energy-aware software and low-power hardware solutions for these gateways to improve the quality of communication in IoUT networks.

Another promising solution to underwater communication problems, especially in the case of long-range inter-continental telecommunications, is the inter-connection of the IoUT and aeronautical technology. In this technique and according to Table IV, the data in the application layer of underwater networks that is produced by sensors and imagery equipment can be transferred to the overwater buoys and to the floating ships. Therefrom, data will be handed to the low-earth orbit or medium earth orbit communication satellites, or to the aerial vehicles. A similar example of this technique is provided by NCEI, Landsat, *Aquarius*, SARAL, *CryoSat*, *Jason*, *HY2-A*, and IRS satellites in Table VIII for remote sensing of ocean surface parameters. Another example is offered by [115] for the connectivity of unmanned aerial vehicles and IoUT.

Additionally, reducing the data volume to make it suitable for transmission using narrow-bandwidth acoustic technology could be used for mitigating the communication challenges in large-scale IoUT networks. This was discussed under the concept of MCC and MEC edge computing paradigms in Section II-J. According to this method, edge-processing is capable of reducing the volume of raw data [113]. This consequently reduces the bandwidth requirement.

In this regard, the combination of the MEC paradigm with smart unmanned vehicles can offer an alternative solution to the problem of long-distance underwater communications. In this solution, autonomous underwater vehicles [114], unmanned aerial vehicles [115], etc. can be used for IoUT data collection. Here, edge computing can undertake some essential computations, so that big data collection will be mitigated, while latency-sensitive situations are handled promptly [115]. However, the lack of energy in undersea environments, makes the employment of MEC a challenge. In closing we note that MEC has a similar architecture to the previously suggested inter-connection between the IoT and IoUT.

E. Dynamic IoUT Signal Routing and Traffic Control

The underwater propagation environment is quite hostile; hence it is of low channel capacity, which makes even point-to-point single-link data transmission challenging in the IoUT. This becomes even more challenging when a network of concurrently communicating nodes is considered. For example, in the presence of a network supporting multiple transmitters, the tele-traffic escalates, and traffic control becomes a challenging task, when aiming for a reasonable QoS [1].

To avoid any tele-traffic congestion, efficient routing management is crucial. To elaborate a little further, in *ad hoc*

networks the design-dilemma is whether to use more short hops at the cost of an increased delay or fewer longer hops. It is beneficial to use low-complexity non-coherent transceiver techniques and take into account the battery-charge during routing.

Additionally, intelligent traffic control systems, using both the deep learning approaches [297] and the SDNs [298], [299] may be devised in underwater applications. These systems are capable of efficiently handling concurrent data transfers to avoid congestion.

To employ deep learning in traffic control systems, we first have to define the action space. To do so, consider a heterogeneous IoUT network constructed of both wired and wireless connections. As detailed in Section II-C2, all the nodes of such a network may connect with one another, using multiple hops. As a consequent of both the limited underwater transmission range [300] or the large-scale infrastructural size of the network [110], the number of possible hops escalates, leading to numerous potential paths for a data packet to travel between a pair of nodes. In this context, every path can be considered as an action in the action space. After taking an action, the system's feedback (a.k.a. reward) can be quantified by the traffic load level of the nodes in the following timeslot, which is formatted as an award value matrix. The combination of the actions and their consequent rewards provides the required training data for semi-supervised deep reinforcement learning, as advocated in [297]. To the best of the authors' knowledge, such datasets for IoT, do not exist for IoUT. Once collected, these datasets of IoUT routing management and traffic control can then be used to train ML-based models. After successfully passing the training phase, this deep model can be used to infer the best overall path for a given data packet.

Although the above DNN can be conveniently used in most of the IoUT structures, it might falter as the IoUT architecture gets wider. By expanding the network scale, the number of possible paths will increase exponentially, and the deep NN can no longer learn the patterns in the data flow. To address this problem, Fadlullah *et al.* [297] have proposed a solution, which can also be employed in underwater applications. To elaborate a little further, they suggested to:

- 1) Change the action space from the entire set of all path combinations to simply the next hop destination;
- 2) Replace the semi-supervised reinforcement learning by a cascaded combination of supervised CNN and DBN;
- 3) 3) Predict the award value matrix (by CNN) before deciding on the best action (by DBN).

These recommendations for using DNNs for wide-scale IoUT traffic control will reduce the packet loss rate as well as improve the network throughput. As a result, the routing performance will improve compared to that of the conventional methods.

The other technique of improving the network throughput is that of adhering to the SDN methodology. This type of network is described in Section II-I, which relies on a centralized control system, for monitoring the network's traffic flow. According to Xie *et al.* [298], the centralized management of SDNs substantially benefits from using DNNs. For example,

the functions of routing optimization, traffic prediction, path load prediction, node deployment optimization, delay prediction, QoS prediction, content delivery optimization, resource allocation, SDN reconfiguration, optimized spectrum sharing, number of active nodes estimation, and intrusion detection would all benefit from using DNNs in overwater SDNs. Hence it is promising to critically appraise their synergies also in underwater applications.

F. Securing Underwater Networks

As discussed in Section II-H, network security has to be considered throughout the entire IoUT architectural model, from the physical layer to the application layer. However, the highly unreliable underwater communication channels having high propagation delays and low energy resources, make network security a challenging topic [111].

In this context, and according to a review article by Jiang *et al.* [97], two major concerns in underwater communication security, which require further investigations and therefore offer research opportunities are:

- According to both Section II-D and Table IV, underwater sensors in IoUT are considered as network endpoints, located at the highest TCP/IP layer (i.e., the application layer). These endpoints are vulnerable to potential intrusion attacks and therefore, have to be secured. But the main focus in UWSN security researches was on the lower physical, data link, and network layers, missing out the higher transport and application layers. It is difficult to guarantee a secure connection in IoUT by relying on the lower three layers alone. On the other hand, the corresponding application layer security protocols of the conventional IoT are often computationally complex and hence power-thirsty. Those protocols are not readily applicable to underwater endpoints. Therefore, upper-layer security should be further investigated to provide a reliable underwater network.
- Considering the resource-constrained nature of the IoUT, compared to the terrestrial IoT, a comprehensive framework is required for optimally distributing the security-related functionalities across all the layers. This optimal distribution of security tasks will minimize the overall resource consumption of the UWSN. To the best of our knowledge, no comprehensive framework is available for energy-effective security enforcement in IoUT at the time of writing.

To address these challenges, taking a similar approach to the IoT can be helpful. Earlier in Section II-H, we introduced several layer-based security protocols to be employed in the IoUT infrastructure. We also suggested ECC as a cryptographic primitive in the application layer of the IoUT. However, due to the large-scale nature of IoUT systems and their complex environment, traditional cyber security techniques cannot be readily adapted. Therefore, new research directions relying on DL may be adopted. To address this issue, large datasets of different IoUT security attacks have to be collected first. To the best of our knowledge, such datasets do not exist for underwater applications. However, datasets of various security

attacks are already available for download for the conventional IoT (e.g. CICIDS2017) [301]. Once collected, these datasets of IoUT security attacks can then be used to train ML-based models for protecting IoUT networks against a wide range of security threats.

Having said that, if a large and clean dataset is not collectable for a given IoUT attack, semi-supervised reinforcement learning might be considered. As already discussed in Section V-E, the deep reinforcement learning models can be trained by the combination of actions and their consequent rewards. As a result, these models can be trained during a practical experiment, leaving out the data gathering step. Such an experiment has been conducted by Xiao *et al.* [302] in a pool with underwater transducers. They have used this setup to train their proposed reinforcement learning model against sophisticated jamming attacks in UWSNs. With no prior data gathering, the node has inferred, where the heavily jammed locations are and avoided them. It has also learned to adjust its power-level for achieving the required bit error rate.

The IoUT architectural model of Table IV is based on the similar 5-layer TCP/IP model of IoT. As a result, the layer-wise security threats of IoUT are of similar nature to the common layer-wise attacks experienced in IoT. Some examples of these common IoT/IoUT security issues are related to the recognition and segregation of malicious requests (e.g. distributed denial-of-service protection), prevention of policy violations (e.g. anomaly as well as intrusion detection), provision of authentication (e.g. man-in-the-middle attack prevention and data-injection protection), etc. [97], [111].

When choosing an ML-based model for satisfying certain security requirements in IoT, CNN as well as RNN deep NNs are proven to have a better single-attack detection performance, than to other traditional NNs. Furthermore, if a sole ML model has to detect a collection of multiple attacks, DBNs might be considered as a better choice [301], [303]. However, the application of these ML models in IoUT requires further investigations. We anticipate that the security domain in IoUT can significantly benefit from the techniques already available in IoT. This offers a great research opportunity for the developers to adapt these methods to the challenging underwater environment.

G. Deleterious Effects of Imprecise Channel Modeling

Naturally, the channel plays an essential role in designing the underwater deployment of endpoint nodes, relay nodes, and sinks. The strategic deployment of nodes is capable of increasing the entire system's battery life and improving the QoS. Almost all of the underwater acoustic, electromagnetic, and optical channel models as described in Sections II-B and II-E and also the underwater magnetic induction channel model of [304] rely on approximations to simplify the overall model. For example, many channel models assume straight signal propagation undersea, which simplifies numerical calculations [6]. However, approximations and idealized simplifying assumptions will result in imprecise models and inaccurate communication.

Imprecise channel modeling can also lead to inaccurate simulation-based modeling of underwater communications.

Below, we provide a list of recent publications, which showed the impact of imperfect channel modelling in the simulations of underwater communication, and therefore used precise channel models for their simulations.

- Using a preamble alerts the receiver about the reception of an incoming data burst and switches it from its low-power dormant mode to its high-power active mode. Therefore, both missing the detection of a preamble and declaring its reception, when it is actually absent reduces the receiver's battery life. A beneficial preamble detection method was proposed by Li *et al.* [85], for an underwater digital communication system, which coexists with other deployed networks. Using an accurate channel model is critical in this identification method in order to prevent a receiver from being triggered by other systems and consequently extends the underwater battery life.
- IoUT communication is affected by numerous signal impairments, such as a high propagation delay and high signal attenuation, as shown in Fig. 4. These reduce the link reliability, which can be mitigated by sophisticated Forward Error Correction (FEC) and Automatic Repeat reQuest (ARQ) techniques. However, at low SNRs the throughput may be reduced by excessive data retransmission. Liang *et al.* [86] optimized the overall transmission redundancy to be used in UWSNs by relying on an accurate channel model.
- In the simulation-based investigation of adaptive transmission in time varying underwater acoustic channels conducted by Wang *et al.* [87], the transmitter's data queue length and the predicted channel conditions were relied upon for designing the adaptive transmitter parameter values. Although their method relies on reinforcement learning to yield considerable energy saving, it cannot correctly learn the accurate parameter values without a precise channel model.

To address the need for precise channel modeling, further research is required for devising precise yet computationally efficient models, for ensuring that the simulation of underwater communications in context of the emerging IoUT is as reliable as possible. This would mitigate the need for and the likelihood of future network upgrades.

H. Sparse and High-maintenance Sensing Devices in IoUT

In contrast to the IoT, the sensing devices of the IoUT are sparsely deployed and exposed to severe environmental effects [4]. The sparse configuration and the harsh environmental conditions make the maintenance of the IoUT, costly. Explicitly, maintenance should mitigate the effects of erosion, corrosion, sediments, pollutions, and other phenomena imposed by seawater.

To address these issues and to reduce the maintenance cost of sparse high-maintenance nodes in the IoUT, a compelling solution is to incorporate self-management capability [88], including self-evaluation, self-configuration, and automatic reports to human operators. Therefore, developing intelligent ML-based hardware nodes for the IoUT, which have a self-management and decision-making capability and conceiving

their required software are promising avenues for research in IoUT, which need further attention.

I. Poor Underwater Positioning and Navigation

As mentioned earlier, GPS signals do not penetrate the sea, hence other navigation techniques have to be used. A number of navigation methods such as blind positioning, acoustic transponders, ranging sonars, image-based positioning, and SLAM were introduced in Section III-D2. However, none of these stand-alone techniques offer a non-accumulating positioning error, therefore, none of them are adequate [161].

Hence, further research is required for improving underwater navigation systems. The challenges to address are:

- Selecting and combining the large amount of data from the aforementioned stand-alone navigation techniques is a challenging task, while considering the system's cost and complexity as well as precision. A research opportunity to address this issue is to study different combinations of navigation techniques, while striking a trade-off between cost and accuracy. For instance, Bonin-Font *et al.* [20] have combined image-based positioning with SLAM to achieve improved navigation. Another study has combined blind positioning with a long-baseline acoustic transponder to reduce the positioning errors caused by acoustic ray bending and variable sound velocities [162].
- Preliminary environmental survey for acquiring offline data is a challenging mandatory step, which has to be conducted before scene analysis based localization techniques (i.e. image-based positioning systems as well as acoustic transponders). This step is required for extracting positional fingerprints (i.e. features), which will be subsequently used for accurately training a localization ML algorithm. However, gathering these big datasets to train ML algorithms is not trivial in underwater environments. Furthermore, traditional ML models can be easily mislead by any variation in the underlying high-dynamic underwater scenes. To address this, using deep NNs having automatic feature extraction capability is highly recommended. The benefits of diverse statistical methods, traditional NNs, and deep NNs in scene-based indoor positioning systems are reported in [305]. These quick lines can be adapted for underwater applications.
- Finding and implementing new natural phenomena for improving underwater positioning as well as navigation is another challenge. In this context, a gravity-aided navigation system was conceived in [163], which is based on exploiting the difference between the observed and the predicted gravity. These methods however, suffer from biases and error accumulation, which has to be addressed in future research.
- Adapting the existing Low-Power Wide Area Network (LPWAN) technologies, which has been discussed in Section II-D5, to carry out IoUT localization is a potential opportunity. In a review article of Zafari *et al.* [305], a collection of these wireless technologies (i.e. SigFox, Lo-RaWAN, Weightless, etc.) was studied in IoT localization. Using the same approach would be beneficial for IoUT,

as they all consume extremely low energy and operate in a wide reception range.

J. Non-Destructive Testing in Underwater Applications

Non-destructive testing (NDT) is the process of inspecting a system, device, or component, without imposing any changes on its shape or material. NDT can be performed while the Device Under Test (DUT) continues its normal operation. For example, both the TDR and TDT methods discussed in Section V-C, are variants of NDT methodologies relying on optical fibers, which are undertaken without interrupting the normal operation of their DUT [292].

Apart from optical fibers, NDT can also be carried out by other equipment. Some of these NDT tools include visible light cameras, electromagnetic flux detectors, ultrasonic transceivers, and magnetic inductors (i.e. eddy current sensors). Using these equipment in underwater NDT are reported in many diverse applications, such as:

- Leakage detection [292], vibration recognition [306], and non-stationary disturbances as well as strain sensing [307] in pipelines, by fiber-optics;
- Bending and deformation inspection in flexible risers, by fiber-optics [293];
- Temperature and pressure monitoring in downhole tools, by fiber-optics [294];
- Determining water-level in unmanned water resource management systems, by visual cameras [308];
- Inspecting the outer surface of large ship hulls, by electromagnetic flux detectors [309];
- High-sensitivity hydrophones for opto-acoustic imaging, by ultrasonic [310];
- Welding inspection and defect characterization in offshore platforms, by magnetic induction [311].

Developing a classic NDT algorithm for a typical component is not an easy task, requiring knowledge about both the DUT and the underlying physics of the NDT itself (i.e. ultrasonic, electromagnetic, or optical wave scattering). Nevertheless, in the modern age of ML techniques, one can gather a big dataset from any DUT. These data samples can then be fed to a deep NN for training purposes. By implementing this deep learning approach, developing NDT algorithms can be carried out faster, with minimal knowledge about the underlying physical concepts [306]–[308].

Another opportunity in underwater NDT is to use the recent developments in the field of distributed and cloud-based BMD Processing tools, which has been discussed in Section III-C. Relying on the frameworks listed in Table X, one can readily glean data from multiple independent NDT equipment, and then employ a data fusion technique from Section III-A3, to combine and process the gathered big data. It is proven by Bayes' theorem that the uncertainty in the final test result will dramatically decrease by fusing data from independent data sources [311].

K. Lack of Strong Data Leveraging Tools

The processing of BMD requires powerful hardware and software tools that can automatically extract knowledge from

large databases. Some of these hardware tools were introduced throughout this paper, specially in Sections III-C and IV-E. However, more advanced software techniques are required for automatic long-term data-gathering and data-monitoring applications. For instance, even though unmanned auto-annotation based industrial software is in very high demand for classification and labeling underwater objects, plants, or creatures, the existing marine image-annotation software packages are only semi-automated, at best [159].

The shortages in automatic data leveraging techniques is partially due to the uncertainty in selecting feature-set. Ambiguity in feature selection and in the ensuing feature reduction is a consistent challenge in automated ML-aided projects. In a conventional neural network, there are a variety of features and descriptors, hence selecting the most useful ones is challenging. These diverse underwater feature-sets have previously been discussed in Section IV-D and summarized in Table XIV. Recall that relying on a single feature is usually inadequate for accurate underwater classification and clustering [210].

Although modern deep learning approaches (e.g. fully-connected, CNN, Autoencoder, etc.) are promising in terms of overcoming this challenge, the advantages of classical ML methods may nonetheless provide better results. These benefits of classical ML methods include having fewer parameters, more rapid convergence during their training, better insights into the tangible physical interpretation of their operation, and much easier debugging as well as tuning the network. Further research is required for developing techniques and algorithms to infer useful features from a dataset, or even to automate the feature selection process. This will be invaluable for the ML community, because feature selection has a direct impact on the performance of ML-based solutions.

L. Training Deep Networks

When using deep networks, the feature extraction will be automatically handled by the hidden layers of the network. Despite this automatic feature extraction, the deep networks have the disadvantage of requiring large amounts of data for tuning their weights and biases, during the training step. The deeper and wider these networks become, the more useful features can be extracted [312]. Ultimately, we have to strike a trade-off between the training data volume required, the network size/power and its overall performance.

To address this trade-off, a pair of general solutions could be offered. The first solution is to satisfy the deep network's hunger by more data. The extra data can be generated automatically (for example by various data augmentation techniques or by employing generative neural networks, such as the Generative Adversarial Network (GAN) [313], [314]) or manually (by using web-based technologies to enlist the assistance of international experts and to produce a large volume of user-generated contents).

The second solution is to enhance the deep network's efficiency by modifying their building blocks and the inter-connection of neurons (like the convolution, pooling, and activation operations of a typical CNN [315] or pruning a deep NN to ease their operation on mobile devices [316]), in

order to reduce the number of network parameters. The vibrant deep learning community will no doubt continue to improve the deep network training and tackle the challenges. This will definitely improve the IoUT applications and advance BMD analytics.

M. Degraded Underwater Images

Undersea photography is always affected by environmental factors. Although the contributions surveyed in Section IV-C2 have addressed some of these issues, there is a lot of room for improvement. For instance, developing an imaging system capable of both real-time forward- and back-scattering elimination is critically needed.

Furthermore, the underwater image quality could be significantly improved by applying some well-known hardware methods and techniques, such as light polarization [317], multi-spectral imaging [318], or stereoscopic imaging [319]. Additional improvements may also be attained by software methods, such as wavelength compensation and color reconstruction [135], [197] both in active and passive underwater photography. Finally, the new concept of image reconstruction with DL [190], [250], [251] has promise for future research.

VI. CONCLUSION

The recent advances in IoT technology and the extension of its influence both to coastal and open sea areas has led to the proliferation of the number of Internet-connected objects both in over- and underwater applications. This technological evolution inspired the new scientific concept of IoUT constituted by marine sensors, cameras, hydrophones, etc. This concept opens many new research directions for undersea data acquisition, data communication, BMD handling, and oceanic data processing.

In this article, we commenced by surveying the state-of-the-art in underwater communications. Given the harsh underwater propagation environment, data communication is quite a challenging task in the IoUT. The families of advanced underwater communication models of acoustic, electromagnetic, and optical technologies were introduced and innovative solutions were proposed for increasing the overall link reliability by topology and routing optimization, security improvement, and protocol enhancements. Furthermore, the underwater channel modeling was studied along with software tools to simulate both those channel models as well as the communication protocols. Both SDN and edge computing techniques were also reviewed as a promising technique of improving underwater communications. We also reviewed the IoUT network architecture, based on the well-known 5-layer TCP/IP standard model.

Naturally, the IoUT leads to BMD generation and the associated challenges include data storage, transportation, preparation, and analysis. Because manual and semi-automatic data processing methods are no longer appropriate in the new era of the IoUT, the five system components of BMD solutions were discussed and the most recent frontier-research and a range of practical solutions were discussed for each component. These solutions covered the areas of sensor, image, and video data

sources, marine geographic data, localization and tracking, open access databases, distributed data processing, and cloud-based services. A complete section was also dedicated to machine intelligence (i.e. ML and DL) and its applications to marine data processing. The most recent research articles in both the hardware and software aspects of the IoUT and BMD processing were also surveyed, along with the critical appraisal of these works. Finally, numerous open research issues and future study directions were presented to provide an insight into the prospective applications, trends, and challenges. Do join this vibrant interdisciplinary research community, valued colleagues.

REFERENCES

- [1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: a survey on enabling technologies, protocols, and applications," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2347–2376, Jun. 2015.
- [2] M. H. Eldefrawy, N. Pereira, and M. Gidlund, "Key distribution protocol for industrial internet of things without implicit certificates," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 906–917, Feb. 2019.
- [3] C.-C. Kao, Y.-S. Lin, G.-D. Wu, and C.-J. Huang, "A comprehensive study on the internet of underwater things: applications, challenges, and channel models," *Sensors*, vol. 17, no. 7, Jul. 2017.
- [4] M. C. Domingo, "An overview of the internet of underwater things," *J. Netw. Comput. Appl.*, vol. 35, no. 6, pp. 1879–1890, Nov. 2012.
- [5] WHOI, "Know your ocean," Feb. 2019. [Online]. Available: <http://www.whoi.edu/know-your-ocean/>
- [6] Y. Li, Y. Zhang, W. Li, and T. Jiang, "Marine wireless big data: efficient transmission, related applications, and challenges," *IEEE Wireless Commun.*, vol. 25, no. 1, pp. 19–25, Feb. 2018.
- [7] G. Xiao, B. Wang, Z. Deng, M. Fu, and Y. Ling, "An acoustic communication time delays compensation approach for master-slave AUV cooperative navigation," *IEEE Sensors J.*, vol. 17, no. 2, pp. 504–513, Jan. 2017.
- [8] Z. Zeng, S. Fu, H. Zhang, Y. Dong, and J. Cheng, "A survey of underwater optical wireless communications," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 1, pp. 204–238, Jan. 2017.
- [9] C. A. Pérez, F. S. Valles, R. T. Sánchez, M. J. Buendía, F. López-Castejón, and J. G. Cervera, "Design and deployment of a wireless sensor network for the Mar Menor coastal observation system," *IEEE J. Ocean. Eng.*, vol. 42, no. 4, pp. 966–976, Oct. 2017.
- [10] F. Javed, M. K. Afzal, M. Sharif, and B. Kim, "Internet of things (IoT) operating systems support, networking technologies, applications, and challenges: a comparative review," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 3, pp. 2062–2100, Mar. 2018.
- [11] X. Cao, L. Liu, Y. Cheng, and X. S. Shen, "Towards energy-efficient wireless networking in the big data era: a survey," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 1, pp. 303–332, Jan. 2018.
- [12] S. Verma, Y. Kawamoto, Z. M. Fadlullah, H. Nishiyama, and N. Kato, "A survey on network methodologies for real-time analytics of massive IoT data and open research issues," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1457–1477, Apr. 2017.
- [13] L. F. Maimo, A. L. P. Gomez, F. J. G. Clemente, M. G. Perez, and G. M. Perez, "A self-adaptive deep learning-based system for anomaly detection in 5G networks," *IEEE Access*, vol. 6, pp. 7700–7712, Feb. 2018.
- [14] L. Meng, T. Hirayama, and S. Oyanagi, "Underwater-drone with panoramic camera for automatic fish recognition based on deep learning," *IEEE Access*, vol. 6, pp. 17 880–17 886, Mar. 2018.
- [15] X. Sun, J. Shi, L. Liu, J. Dong, C. Plant, X. Wang, and H. Zhou, "Transferring deep knowledge for object recognition in low-quality underwater videos," *Neurocomp.*, vol. 275, pp. 897–908, Jan. 2018.
- [16] N. Ani Brown Mary and D. Dharma, "Coral reef image classification employing improved LDP for feature extraction," *J. Vis. Commun. Image Repres.*, vol. 49, pp. 225–242, Nov. 2017.
- [17] M. Reggiannini and O. Salvetti, "Seafloor analysis and understanding for underwater archeology," *J. Cultur. Heritage*, vol. 24, pp. 147–156, Apr. 2017.

- [18] M. Jian, Q. Qi, J. Dong, Y. Yin, and K.-M. Lam, "Integrating QDWD with pattern distinctness and local contrast for underwater saliency detection," *J. Vis. Commun. Image Repres.*, vol. 53, pp. 31–41, May 2018.
- [19] T. Schoening, T. Kuhn, D. O. B. Jones, E. Simon-Lledo, and T. W. Nattkemper, "Fully automated image segmentation for benthic resource assessment of poly-metallic nodules," *Methods Ocean.*, vol. 15–16, pp. 78–89, Jul. 2016.
- [20] F. Bonin-Font, A. Burguera, and J. L. Lisani, "Visual discrimination and large area mapping of Posidonia Oceanica using a lightweight AUV," *IEEE Access*, vol. 5, pp. 24 479–24 494, Oct. 2017.
- [21] M. C. Chuang, J. N. Hwang, J. H. Ye, S. C. Huang, and K. Williams, "Underwater fish tracking for moving cameras based on deformable multiple kernels," *IEEE Trans. Syst., Man, Cybern.: Syst.*, vol. 47, no. 9, pp. 2467–2477, Sep. 2017.
- [22] P. Y. Lau, P. L. Correia, P. Fonseca, and A. Campos, "Estimating Norway lobster abundance from deep-water videos: an automatic approach," *IET Image Process.*, vol. 6, no. 1, pp. 22–30, Feb. 2012.
- [23] J. Osterloff, I. Nilssen, and T. W. Nattkemper, "A computer vision approach for monitoring the spatial and temporal shrimp distribution at the LoVe observatory," *Methods Ocean.*, vol. 15–16, pp. 114–128, Jul. 2016.
- [24] M. S. Schmid, C. Aubry, J. Grigor, and L. Fortier, "The LOKI underwater imaging system and an automatic identification model for the detection of zooplankton taxa in the Arctic Ocean," *Methods Ocean.*, vol. 15–16, pp. 129–160, Jul. 2016.
- [25] R. Failletaz, M. Picheral, J. Y. Luo, C. Guigand, R. K. Cowen, and J.-O. Irissou, "Imperfect automatic image classification successfully describes plankton distribution patterns," *Methods Ocean.*, vol. 15–16, pp. 60–77, Jul. 2016.
- [26] J. Pearlman, S. Jirka, J. d. Rio, E. Delory, L. Frommhold, S. Martinez, and T. O. Reilly, "Oceans of Tomorrow sensor interoperability for in-situ ocean monitoring," in *Proc. Ocean. Monterey, USA: IEEE*, Sep. 2016, pp. 1–8.
- [27] G. L. Foresti, "Visual inspection of sea bottom structures by an autonomous underwater vehicle," *IEEE Trans. Syst., Man, Cybern., Part B (Cybern.)*, vol. 31, no. 5, pp. 691–705, Oct. 2001.
- [28] R. Khalifeh, M. S. Yasri, B. Lescop, F. Gallée, E. Diler, D. Thierry, and S. Rioual, "Development of wireless and passive corrosion sensors for material degradation monitoring in coastal zones and immersed environment," *IEEE J. Ocean. Eng.*, vol. 41, no. 4, pp. 776–782, Oct. 2016.
- [29] L. Paull, S. Saeedi, M. Seto, and H. Li, "AUV navigation and localization: a review," *IEEE J. Ocean. Eng.*, vol. 39, no. 1, pp. 131–149, Jan. 2014.
- [30] T. Miyoshi *et al.*, "“big data assimilation” toward post-petascale severe weather prediction: an overview and progress," *Proc. IEEE*, vol. 104, no. 11, pp. 2155–2179, Nov. 2016.
- [31] C. Kalyvas, A. Kokkos, and T. Tzouramanis, "A survey of official online sources of high-quality free-of-charge geospatial data for maritime geographic information systems applications," *Inf. Syst.*, vol. 65, pp. 36–51, Apr. 2017.
- [32] E. Ahmed, I. Yaqoob, I. A. T. Hashem, I. Khan, A. I. A. Ahmed, M. Imran, and A. V. Vasilakos, "The role of big data analytics in internet of things," *Comput. Netw.*, vol. 129, pp. 459–471, Dec. 2017.
- [33] M. Mohammadi, A. Al-Fuqaha, S. Sorour, and M. Guizani, "Deep learning for IoT big data and streaming analytics: a survey," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 2923–2960, Jun. 2018.
- [34] D. Huang, D. Zhao, L. Wei, Z. Wang, and Y. Du, "Modeling and analysis in marine big data: advances and challenges," *Math. Probl. Eng.*, vol. 2015, pp. 1–13, Sep. 2015.
- [35] G. Xu, Y. Shi, X. Sun, and W. Shen, "Internet of things in marine environment monitoring: a review," *Sensors*, vol. 19, no. 7, p. 1711, Apr. 2019.
- [36] M. Jouhari, K. Ibrahim, H. Tembine, and J. Ben-Othman, "Underwater wireless sensor networks: a survey on enabling technologies, localization protocols, and internet of underwater things," *IEEE Access*, vol. 7, pp. 96 879–96 899, Jul. 2019.
- [37] T. Qiu, Z. Zhao, T. Zhang, C. Chen, and C. L. P. Chen, "Underwater internet of things in smart ocean: system architecture and open issues," *IEEE Trans. Ind. Inform.*, vol. 16, no. 7, pp. 4297–4307, Jul. 2020.
- [38] D. Raj, J. Lee, E. Ko, S. Shin, J.-I. Namgung, S.-H. Yum, and S.-H. Park, "Underwater network management system in internet of underwater things: open challenges, benefits, and feasible solution," *Electron.*, vol. 9, p. 1142, Jul. 2020.
- [39] R. Khalil, M. Babar, T. Jan, and N. Saeed, "Towards the internet of underwater things: recent developments and future challenges," *IEEE Consum. Electron. Mag.*, p. 1, May 2020.
- [40] Z. Zhou, B. Yao, R. Xing, L. Shu, and S. Bu, "E-CARP: an energy efficient routing protocol for UWSNs in the internet of underwater things," *IEEE Sensors J.*, vol. 16, no. 11, pp. 4072–4082, Jun. 2016.
- [41] F. Adamo, F. Attivissimo, C. G. C. Carducci, and A. M. L. Lanzolla, "A smart sensor network for sea water quality monitoring," *IEEE Sensors J.*, vol. 15, no. 5, pp. 2514–2522, May 2015.
- [42] R. Al-Zaidi, J. C. Woods, M. Al-Khalidi, and H. Hu, "Building novel VHF-based wireless sensor networks for the internet of marine things," *IEEE Sensors J.*, vol. 18, no. 5, pp. 2131–2144, Mar. 2018.
- [43] T. P. Hughes *et al.*, "Global warming and recurrent mass bleaching of corals," *Nature*, vol. 543, pp. 373–377, Mar. 2017.
- [44] J. F. Fuentes-Pérez, C. Meurer, J. A. Tuhtan, and M. Kruusmaa, "Differential pressure sensors for underwater speedometry in variable velocity and acceleration conditions," *IEEE J. Ocean. Eng.*, vol. 43, no. 2, pp. 418–426, Apr. 2018.
- [45] C. A. Balanis, *Antenna theory: analysis and design*. John Wiley & Sons, 2016.
- [46] M. A. Ainslie and J. G. McCollm, "A simplified formula for viscous and chemical absorption in sea water," *J. Acoust. Soc. America*, vol. 103, no. 3, pp. 1671–1672, Jun. 1998.
- [47] C. Gabriel, M. A. Khalighi, S. Bourennane, P. Leon, and V. Rigaud, "Monte-Carlo-based channel characterization for underwater optical communication systems," *IEEE/OSA J. Opt. Commun. Netw.*, vol. 5, no. 1, pp. 1–12, Jan. 2013.
- [48] E. Illi, F. E. Bouanani, K. Park, F. Ayoub, and M. Alouini, "An improved accurate solver for the time-dependent RTE in underwater optical wireless communications," *IEEE Access*, vol. 7, pp. 96 478–96 494, Jul. 2019.
- [49] J. S. Jaffe, "Underwater optical imaging: the past, the present, and the prospects," *IEEE J. Ocean. Eng.*, vol. 40, no. 3, pp. 683–700, Jul. 2015.
- [50] J. Wang, H. Zhou, Y. Li, Q. Sun, Y. Wu, S. Jin, T. Q. S. Quek, and C. Xu, "Wireless channel models for maritime communications," *IEEE Access*, vol. 6, pp. 68 070–68 088, Dec. 2018.
- [51] W. Sun, X. Yuan, J. Wang, Q. Li, L. Chen, and D. Mu, "End-to-end data delivery reliability model for estimating and optimizing the link quality of industrial WSNs," *IEEE Trans. Autom. Sci. Eng.*, vol. 15, no. 3, pp. 1127–1137, Jul. 2018.
- [52] H. Zhang, X. Liu, C. Li, Y. Chen, X. Che, L. Y. Wang, F. Lin, and G. Yin, "Scheduling with predictable link reliability for wireless networked control," *IEEE Trans. Wireless Commun.*, vol. 16, no. 9, pp. 6135–6150, Sep. 2017.
- [53] R. D. Gomes, C. Benavente-Peces, I. E. Fonseca, and M. S. Alencar, "Adaptive and beacon-based multi-channel protocol for industrial wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 132, pp. 22–39, Apr. 2019.
- [54] S. Murali and A. Jamalipour, "Mobility-aware energy-efficient parent selection algorithm for low power and lossy networks," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2593–2601, Apr. 2019.
- [55] J. C. Fink, K. Fermanich, and T. Ehlinger, *The effects of urbanization on Baird Creek, Green Bay, Wisconsin*. University of Wisconsin-Green Bay, 2005.
- [56] H. Kaushal and G. Kaddoum, "Underwater optical wireless communication," *IEEE Access*, vol. 4, pp. 1518–1547, Apr. 2016.
- [57] Z. Li and Y. Wu, "Smooth mobility and link reliability-based optimized link state routing scheme for MANETs," *IEEE Commun. Lett.*, vol. 21, no. 7, pp. 1529–1532, Jul. 2017.
- [58] S. Rani, S. H. Ahmed, J. Malhotra, and R. Talwar, "Energy efficient chain based routing protocol for underwater wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 92, pp. 42–50, Aug. 2017.
- [59] H. Tran-Dang and D. Kim, "Channel-aware cooperative routing in underwater acoustic sensor networks," *J. Commun. Netw.*, vol. 21, no. 1, pp. 33–44, Feb. 2019.
- [60] R. Zhao, H. Long, O. A. Dobre, X. Shen, T. M. N. Ngatched, and H. Mei, "Time reversal based MAC for multi-hop underwater acoustic networks," *IEEE Syst. J.*, vol. 13, no. 3, pp. 2531–2542, Sep. 2019.
- [61] B. Fu, Y. Xiao, H. Deng, and H. Zeng, "A survey of cross-layer designs in wireless networks," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 110–126, Jan. 2014.
- [62] M. Dye, R. McDonald, and A. Rufi, *Network fundamentals, CCNA exploration companion guide*, 1st ed. Cisco press, Oct. 2007.
- [63] J.-H. Cui, J. Kong, M. Gerla, and S. Zhou, "The challenges of building mobile underwater wireless networks for aquatic applications," *IEEE Netw.*, vol. 20, no. 3, pp. 12–18, May 2006.

- [64] S. Musilová, "Profiling and detection of IoT attacks in Telnet traffic," Master's Thesis, Czech Technical University in Prague, Department of Computer Science, Jan. 2020.
- [65] M. Xu and L. Liu, "Sender-receiver role-based energy-aware scheduling for internet of underwater things," *IEEE Trans. Emerg. Topics Comput.*, p. 1, Nov. 2016.
- [66] M. Koseoglu, E. Karasan, and L. Chen, "Cross-layer energy minimization for underwater ALOHA networks," *IEEE Syst. J.*, vol. 11, no. 2, pp. 551–561, Jun. 2017.
- [67] F. A. Alfouzan, A. Shahrab, S. M. Ghoreyshi, and T. Boutaleb, "A collision-free graph coloring MAC protocol for underwater sensor networks," *IEEE Access*, vol. 7, pp. 39 862–39 878, Mar. 2019.
- [68] X. Feng, Z. Wang, G. Han, W. Qu, and A. Chen, "Distributed receiver-oriented adaptive multichannel MAC for underwater sensor networks," *IEEE Access*, vol. 6, pp. 11 666–11 675, Feb. 2018.
- [69] Y. Chen and H. Wang, "Ordered CSMA: a collision-free MAC protocol for underwater acoustic networks," in *Proc. Ocean. Vancouver*, Canada: IEEE, Oct. 2007, pp. 1–6.
- [70] S. Jiang, "State-of-the-art Medium Access Control (MAC) protocols for underwater acoustic networks: a survey based on a MAC reference model," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 1, pp. 96–131, Jan. 2018.
- [71] M. Chen, Y. Shen, J. Luis, and C. Chou, "Energy-efficient OR-based MAC protocol for underwater sensor networks," in *Proc. Sensors. Valencia, Spain: IEEE*, Nov. 2014, pp. 118–121.
- [72] C. Li, Y. Xu, C. Xu, Z. An, B. Diao, and X. Li, "DTMAC: a delay tolerant MAC protocol for underwater wireless sensor networks," *IEEE Sensors J.*, vol. 16, no. 11, pp. 4137–4146, Jun. 2016.
- [73] T. Fujihashi, S. Saruwatari, and T. Watanabe, "Multiview video transmission over underwater acoustic path," *IEEE Trans. Multimedia*, vol. 20, no. 8, pp. 2166–2181, Aug. 2018.
- [74] N. Mohamed, I. Jawhar, J. Al-Jaroodi, and L. Zhang, "Monitoring underwater pipelines using sensor networks," in *Proc. 12th Int. Conf. High Perform. Comput. Commun.* Melbourne, Australia: IEEE, Sep. 2010, pp. 346–353.
- [75] B. E. Carpenter, *Network geeks: how they built the internet*. Springer, Apr. 2013.
- [76] M. Y. Aalsalem, W. Z. Khan, W. Gharibi, M. K. Khan, and Q. Arshad, "Wireless sensor networks in oil and gas industry: recent advances, taxonomy, requirements, and open challenges," *J. Netw. Comput. Appl.*, vol. 113, pp. 87–97, Jul. 2018.
- [77] N. W. Bergmann, J. Juergens, L. Hou, Y. Wang, and J. Trevathan, "Wireless underwater power and data transfer," in *Proc. 38th Annual IEEE Conf. Local Comput. Netw.* Sydney, Australia: IEEE, Mar. 2014, pp. 104–107.
- [78] S. T. Lentz, "The NEPTUNE canada communications network," in *Proc. Ocean. Vancouver, Canada: IEEE*, Sep. 2007, pp. 1–5.
- [79] L. Marchetti and R. Reggiani, "An efficient receiver structure for sweep-spread-carrier underwater acoustic links," *IEEE J. Ocean. Eng.*, vol. 41, no. 2, pp. 440–449, Apr. 2016.
- [80] P. Qarabagi and M. Stojanovic, "Statistical characterization and computationally efficient modeling of a class of underwater acoustic communication channels," *IEEE J. Ocean. Eng.*, vol. 38, no. 4, pp. 701–717, Oct. 2013.
- [81] J. Zhou, H. Jiang, P. Wu, and Q. Chen, "Study of propagation channel characteristics for underwater acoustic communication environments," *IEEE Access*, vol. 7, pp. 79 438–79 445, Jun. 2019.
- [82] M. Naderi, A. G. Zajić, and M. Pätzold, "A nonisovelocity geometry-based underwater acoustic channel model," *IEEE Trans. Veh. Technol.*, vol. 67, no. 4, pp. 2864–2879, Apr. 2018.
- [83] M. Naderi, M. Pätzold, R. Hicheri, and N. Youssef, "A geometry-based underwater acoustic channel model allowing for sloped ocean bottom conditions," *IEEE Trans. Wireless Commun.*, vol. 16, no. 4, pp. 2394–2408, Apr. 2017.
- [84] W. Li, Z. Huang, M. Gong, and Y. Ji, "Land-to-underwater optical communication system based on underwater visible light communication network units and fiber links," in *Proc. Asia Commun. Photon. Conf.* Wuhan, China: IEEE, Nov. 2016, pp. 1–3.
- [85] W. Li, S. Zhou, P. Willett, and Q. Zhang, "Preamble detection for underwater acoustic communications based on sparse channel identification," *IEEE J. Ocean. Eng.*, vol. 44, no. 1, pp. 256–268, Jan. 2019.
- [86] M. Liang, J. Duan, and D. Zhao, "Optimal redundancy control strategy for fountain code-based underwater acoustic communication," *IEEE Access*, vol. 6, pp. 69 321–69 334, Nov. 2018.
- [87] C. Wang, Z. Wang, W. Sun, and D. R. Fuhrmann, "Reinforcement learning-based adaptive transmission in time-varying underwater acoustic channels," *IEEE Access*, vol. 6, pp. 2541–2558, Dec. 2017.
- [88] A. Atyabi, S. MahmoudZadeh, and S. Nefti-Meziani, "Current advancements on autonomous mission planning and management systems: an AUV and UAV perspective," *Annual Rev. Control*, vol. 46, pp. 196–215, Aug. 2018.
- [89] R. Naumann, S. Dietzel, and B. Scheuermann, "Push the barrier: discrete event protocol emulation," *IEEE/ACM Trans. Netw.*, vol. 27, no. 2, pp. 635–648, Apr. 2019.
- [90] C. Petrioli, R. Petrocchia, J. R. Potter, and D. Spaccini, "The SUNSET framework for simulation, emulation and at-sea testing of underwater wireless sensor networks," *Ad Hoc Netw.*, vol. 34, pp. 224–238, Nov. 2015.
- [91] I. Dorothy and M. Chandrasekaran, "Simulation tools for mobile ad hoc networks: a survey," *J. Appl. Research Technol.*, vol. 16, no. 5, pp. 437–445, Oct. 2018.
- [92] L. Veltri, L. Davoli, R. Pecori, A. Vannucci, and F. Zanichelli, "NEMO: a flexible and highly scalable network emulator," *Software*, vol. 10, p. 100248, Dec. 2019.
- [93] M. Ghaleb, E. Felemban, S. Subramaniam, A. A. Sheikh, and S. B. Qaisar, "A performance simulation tool for the analysis of data gathering in both terrestrial and underwater sensor networks," *IEEE Access*, vol. 5, pp. 4190–4208, Mar. 2017.
- [94] R. Kisacik et al., "Distance and power based experimental verification of channel model in visible light communication," in *Proc. 27th Signal Process. Commun. Appl. Conf.* Sivas, Turkey: IEEE, Apr. 2019, pp. 1–4.
- [95] G. Yang, L. Dai, G. Si, S. Wang, and S. Wang, "Challenges and security issues in underwater wireless sensor networks," *Proc. Comput. Sci.*, vol. 147, pp. 210–216, Nov. 2019.
- [96] Editorial, "Underwater acoustic communications: where we stand and what is next?" *IEEE J. Ocean. Eng.*, vol. 44, no. 1, pp. 1–6, Jan. 2019.
- [97] S. Jiang, "On securing underwater acoustic networks: a survey," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 729–752, Jan. 2019.
- [98] M. Xu, Y. Fan, and L. Liu, "Multi-party secret key generation over underwater acoustic channels," *IEEE Wireless Commun. Lett.*, vol. 9, no. 7, pp. 1075–1079, Jul. 2020.
- [99] T. C. Yang and W. Yang, "Low probability of detection underwater acoustic communications for mobile platforms," in *Proc. Ocean. Quebec*, Canada: IEEE, Sep. 2008, pp. 1–6.
- [100] J. Jiajia, W. Xianquan, D. Fajie, F. Xiao, L. Chunyue, and S. Zhongbo, "A basic bio-inspired camouflage communication frame design and applications for secure underwater communication among military underwater platforms," *IEEE Access*, vol. 8, pp. 24 927–24 940, Jan. 2020.
- [101] S. Huang, X. Hou, W. Liu, G. Liu, Y. Dai, and W. Tian, "Mimicking ship-radiated noise with chaos signal for covert underwater acoustic communication," *IEEE Access*, vol. 8, pp. 180 341–180 351, Sep. 2020.
- [102] G. Dini and A. L. Duca, "SeFLOOD: a secure network discovery protocol for underwater acoustic networks," in *Proc. IEEE Symp. Comput. Commun.* Kerkyra, Greece: IEEE, Jun. 2011, pp. 636–638.
- [103] M. Zuba, M. Fagan, Z. Shi, and J. Cui, "A resilient pressure routing scheme for underwater acoustic networks," in *Proc. IEEE Glob. Commun. Conf.* Austin, USA: IEEE, Dec. 2014, pp. 637–642.
- [104] A. Caposelle, G. De Cicco, and C. Petrioli, "R-CARP: a reputation based channel aware routing protocol for underwater acoustic sensor networks," in *Proc. 10th Int. Conf. Underwater Netw. Syst.* New York, USA: ACM, Oct. 2015, pp. 1–6.
- [105] H. Luo, K. Wu, R. Ruby, Y. Liang, Z. Guo, and L. M. Ni, "Software-defined architectures and technologies for underwater wireless sensor networks: a survey," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 2855–2888, May 2018.
- [106] H. S. Dol, P. Casari, T. van der Zwan, and R. Otnes, "Software-defined underwater acoustic modems: historical review and the NILUS approach," *IEEE J. Ocean. Eng.*, vol. 42, no. 3, pp. 722–737, Jul. 2017.
- [107] Y. Luo, L. Pu, M. Zuba, Z. Peng, and J. Cui, "Challenges and opportunities of underwater cognitive acoustic networks," *IEEE Trans. Emerg. Topics Comput.*, vol. 2, no. 2, pp. 198–211, Jun. 2014.
- [108] I. F. Akyildiz, P. Wang, and S.-C. Lin, "SoftWater: software-defined networking for next-generation underwater communication systems," *Ad Hoc Netw.*, vol. 46, pp. 1–11, Aug. 2016.
- [109] C. Srimathi, S.-H. Park, and N. Rajesh, "Proposed framework for underwater sensor cloud for environmental monitoring," in *Proc. 5th Int. Conf. Ubiquitous Future Netw.* Da Nang, Vietnam: IEEE, Jul. 2013, pp. 104–109.
- [110] A. Celik, N. Saeed, B. Shihada, T. Y. Al-Naffouri, and M. Alouini, "A software-defined opto-acoustic network architecture for internet of underwater things," *IEEE Commun. Mag.*, vol. 58, no. 4, pp. 88–94, Apr. 2020.

- [111] C. Lal, R. Petroccia, K. Pelekanakis, M. Conti, and J. Alves, "Toward the development of secure underwater acoustic networks," *IEEE J. Ocean. Eng.*, vol. 42, no. 4, pp. 1075–1087, Oct. 2017.
- [112] J. Liu, G. Shou, Y. Liu, Y. Hu, and Z. Guo, "Performance evaluation of integrated multi-access edge computing and fiber-wireless access networks," *IEEE Access*, vol. 6, pp. 30 269–30 279, May 2018.
- [113] T. Wang, D. Zhao, S. Cai, W. Jia, and A. Liu, "Bidirectional prediction-based underwater data collection protocol for end-edge-cloud orchestrated system," *IEEE Trans. Ind. Inform.*, vol. 16, no. 7, pp. 4791–4799, Jul. 2020.
- [114] S. Cai, Y. Zhu, T. Wang, G. Xu, A. Liu, and X. Liu, "Data collection in underwater sensor networks based on mobile edge computing," *IEEE Access*, vol. 7, pp. 65 357–65 367, May 2019.
- [115] Q. Wang, H. Dai, Q. Wang, M. K. Shukla, W. Zhang, and C. G. Soares, "On connectivity of UAV-assisted data acquisition for underwater internet of things," *IEEE Internet Things J.*, vol. 7, no. 6, pp. 5371–5385, Mar. 2020.
- [116] P. Porambage, J. Okwuibe, M. Liyanage, M. Ylianttila, and T. Taleb, "Survey on multi-access edge computing for internet of things realization," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 2961–2991, Jun. 2018.
- [117] X. Di, Y. Zhang, T. Liu, S. Kang, and Y. Zhao, "Mobile fog computing-assisted resource allocation for two-hop SWIPT OFDM networks," *Wireless Commun. Mob. Comput.*, vol. 2018, Sep. 2018.
- [118] S. Safavat, N. N. Sapavath, and D. B. Rawat, "Recent advances in mobile edge computing and content caching," *Digit. Commun. Netw.*, Sep. 2019.
- [119] M. Chiang and T. Zhang, "Fog and IoT: an overview of research opportunities," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 854–864, Dec. 2016.
- [120] M. S. Brown, *Data mining for dummies*, 1st ed. John Wiley & Sons, Sep. 2014.
- [121] S. Chhabra and D. Singh, "Data fusion and data aggregation/summarization techniques in WSNs: a review," *Int. J. Comput. Appl.*, vol. 121, no. 19, Jul. 2015.
- [122] S. Misra and R. Kumar, "An analytical study of LEACH and PEGASIS protocol in wireless sensor network," in *Proc. Int. Conf. Innov. Inf., Embed. Commun. Syst.* Coimbatore, India: IEEE, Mar. 2017, pp. 1–5.
- [123] S. Chand, S. Singh, and B. Kumar, "Heterogeneous HEED protocol for wireless sensor networks," *Wireless Personal Commun.*, vol. 77, no. 3, pp. 2117–2139, Feb. 2014.
- [124] J. Ma, S. Wang, C. Meng, Y. Ge, and J. Du, "Hybrid energy-efficient APTEEN protocol based on ant colony algorithm in wireless sensor network," *Wireless Commun. Netw.*, vol. 2018, no. 1, p. 102, Dec. 2018.
- [125] J. Qiu, Z. Xing, C. Zhu, K. Lu, J. He, Y. Sun, and L. Yin, "Centralized fusion based on interacting multiple model and adaptive Kalman filter for target tracking in underwater acoustic sensor networks," *IEEE Access*, vol. 7, pp. 25 948–25 958, Feb. 2019.
- [126] F. Castanedo, "A review of data fusion techniques," *Sci. World J.*, vol. 2013, Sep. 2013.
- [127] K. Nomura, D. Sugimura, and T. Hamamoto, "Underwater image color correction using exposure-bracketing imaging," *IEEE Signal Process. Lett.*, vol. 25, no. 6, pp. 893–897, Jun. 2018.
- [128] S. N. Zulkifli, H. A. Rahim, and W.-J. Lau, "Detection of contaminants in water supply: a review on state-of-the-art monitoring technologies and their applications," *Sensors Actuat. B: Chemic.*, vol. 255, pp. 2657–2689, Feb. 2018.
- [129] P. Kassal, M. D. Steinberg, and I. M. Steinberg, "Wireless chemical sensors and biosensors: a review," *Sensors Actuat. B: Chemic.*, vol. 266, pp. 228–245, Aug. 2018.
- [130] J. E. Hewitt and S. F. Thrush, "Monitoring for tipping points in the marine environment," *J. Env. Manag.*, vol. 234, pp. 131–137, Mar. 2019.
- [131] N. G. Wright and H. K. Chan, "Low-cost internet of things ocean observation," in *Proc. Ocean.* Monterey, USA: IEEE, Sep. 2016, pp. 1–5.
- [132] M. Palaniswami, A. S. Rao, and S. Bainbridge, "Real-time monitoring of the Great Barrier Reef using internet of things with big data analytics," *ITU J.: ICT Discov.*, vol. 1, no. 13, pp. 1–10, Oct. 2017.
- [133] O. Godø, S. Johnsen, and T. Torkelsen, "The LoVe ocean observatory is in operation," *Marine Technol. Soc. J.*, vol. 48, Mar. 2014.
- [134] A. S. M. Shihavuddin, N. Gracias, R. Garcia, C. A. Gleason, and B. Gintert, "Image-based coral reef classification and thematic mapping," *Remote Sens.*, vol. 5, no. 4, Apr. 2013.
- [135] M. Han, Z. Lyu, T. Qiu, and M. Xu, "A review on intelligence dehazing and color restoration for underwater images," *IEEE Trans. Syst., Man, Cybern.: Syst.*, vol. PP, no. 99, pp. 1–13, Jan. 2018.
- [136] S. Q. Duntley, "Light in the sea," *J. Opt. Soc. America*, vol. 53, no. 2, pp. 214–233, 1963.
- [137] J. R. V. Zaneveld and W. S. Pegau, "Robust underwater visibility parameter," *Opt. Exp.*, vol. 11, no. 23, pp. 2997–3009, Nov. 2003.
- [138] A. A. M. Salih, K. Hasikin, and N. A. M. Isa, "Adaptive fuzzy exposure local contrast enhancement," *IEEE Access*, vol. 6, pp. 58 794–58 806, Sep. 2018.
- [139] C. Spampinato, D. Giordano, R. Di Salvo, Y.-H. J. Chen-Burger, R. B. Fisher, and G. Nadarajan, "Automatic fish classification for underwater species behavior understanding," in *Proc. 1st Int. Worksh. Anal. Retrieval Tracked Events Motion Imagery Streams*. Firenze, Italy: ACM, Oct. 2010, pp. 45–50.
- [140] M. C. Chuang, J. N. Hwang, K. Williams, and R. Towler, "Tracking live fish from low-contrast and low-frame-rate stereo videos," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 1, pp. 167–179, Jan. 2015.
- [141] S. Negahdaripour, "On 3-D motion estimation from feature tracks in 2-D FS sonar video," *IEEE Trans. Robot.*, vol. 29, no. 4, pp. 1016–1030, Aug. 2013.
- [142] C. Cheng and H. Yang, "Realization of 3D stereo imaging for underwater applications," in *Proc. Int. Conf. Intell. Robot. Control Eng.* Lanzhou, China: IEEE, Aug. 2018, pp. 242–246.
- [143] L. Mazzei, L. Corgnati, S. Marini, E. Ottaviani, and B. Isoppo, "Low cost stereo system for imaging and 3D reconstruction of underwater organisms," in *Proc. Ocean.* Genoa, Italy: IEEE, May 2015, pp. 1–4.
- [144] H. Assali, Y. Petillot, and J. Bell, "Acoustic stereo imaging (ASI) system," in *Proc. Ocean.* Bremen, Germany: IEEE, May 2009, pp. 1–7.
- [145] A. Anwer, S. S. A. Ali, A. Khan, and F. Mériaudeau, "Real-time underwater 3D scene reconstruction using commercial depth sensor," in *Proc. Int. Conf. Underwater Syst. Technol.: Theory Appl.* Penang, Malaysia: IEEE, Dec. 2016, pp. 67–70.
- [146] C.-C. Wang, S.-W. Shyue, and S.-H. Cheng, "Underwater structure inspection with laser light stripes," in *Proc. Int. Symp. Underwater Technol.* Tokyo, Japan: IEEE, May 2000, pp. 201–205.
- [147] S. Ishibashi, "The stereo vision system for an underwater vehicle," in *Proc. Ocean.* Bremen, Germany: IEEE, May 2009, pp. 1–6.
- [148] S. Negahdaripour, H. Sekkati, and H. Pirsiavash, "Opti-acoustic stereo imaging: on system calibration and 3-D target reconstruction," *IEEE Trans. Image Process.*, vol. 18, no. 6, pp. 1203–1214, Jun. 2009.
- [149] A. Marouchos, M. Sherlock, A. Filisetti, and A. Williams, "Underwater imaging on self-contained tethered systems," in *Proc. Ocean.* Anchorage, USA: IEEE, Sep. 2017, pp. 1–5.
- [150] F. Westling, C. Sun, and D. Wang, "A modular learning approach for fish counting and measurement using stereo baited remote underwater video," in *Proc. Int. Conf. Digit. Image Comput.: Techn. Appl.* Wollongong, Australia: IEEE, Nov. 2014, pp. 1–7.
- [151] C. Mulrow and H. Maas, "A universal approach for geometric modelling in underwater stereo image processing," in *Proc. ICPR Worksh. Comput. Vis. Anal. Underwater Imagery*. Stockholm, Sweden: IEEE, Aug. 2014, pp. 49–56.
- [152] E. Trucco and K. Plakas, "Video tracking: a concise survey," *IEEE J. Ocean. Eng.*, vol. 31, no. 2, pp. 520–529, Apr. 2006.
- [153] M. Johnson-Roberson, S. Kumar, O. Pizarro, and S. Williams, "Stereoscopic imaging for coral segmentation and classification," in *Proc. Ocean.* Boston, USA: IEEE, Sep. 2006, pp. 1–6.
- [154] W. Xingang, "A research review of distributed computing system," in *Recent Develop. Intell. Comput., Commun. Devices*. Springer, Aug. 2018, pp. 357–368.
- [155] ASF, "Apache project directory," Feb. 2019. [Online]. Available: <https://projects.apache.org/>
- [156] B. Power and J. Weinman, "Revenue growth is the primary benefit of the cloud," *IEEE Cloud Comput.*, vol. 5, no. 4, pp. 89–94, Jul. 2018.
- [157] Qubole, "Big data trends and challenges," Oct. 2018. [Online]. Available: <https://go.qubole.com/CA---RP---2018-Big-Data-Survey-LP.html>
- [158] S. Zhelev and A. Rozeva, "Big data processing in the cloud - challenges and platforms," in *Proc. 43rd Int. Conf. Appl. Math. Eng. Econo.*, vol. 1910, no. 1. Sozopol, Bulgaria: AIP, Dec. 2017.
- [159] J. N. Gomes-Pereira et al., "Current and future trends in marine image annotation software," *Prog. Ocean.*, vol. 149, pp. 106–120, Dec. 2016.
- [160] V. Barale, "A supporting marine information system for maritime spatial planning: the European atlas of the seas," *Ocean Coast. Manag.*, Dec. 2018.
- [161] P. Liu, B. Wang, Z. Deng, and M. Fu, "INS/DVL/PS tightly coupled underwater navigation method with limited DVL measurements," *IEEE Sensors J.*, vol. 18, no. 7, pp. 2994–3002, Apr. 2018.

- [162] T. Zhang, L. Chen, and Y. Yan, "Underwater positioning algorithm based on SINS/LBL integrated system," *IEEE Access*, vol. 6, pp. 7157–7163, Jan. 2018.
- [163] B. Wang, J. Zhu, Z. Deng, and M. Fu, "A characteristic parameter matching algorithm for gravity-aided navigation of underwater vehicles," *IEEE Trans. Ind. Electron.*, vol. 66, no. 2, pp. 1203–1212, Feb. 2019.
- [164] Y. Chen, D. Zheng, P. A. Miller, and J. A. Farrell, "Underwater inertial navigation with long baseline transceivers: a near-real-time approach," *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 1, pp. 240–251, Jan. 2016.
- [165] Z. Song and K. Mohseni, "Long-term inertial navigation aided by dynamics of flow field features," *IEEE J. Ocean. Eng.*, vol. 43, no. 4, pp. 940–954, Oct. 2018.
- [166] P. Batista, "GES long baseline navigation with unknown sound velocity and discrete-time range measurements," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 1, pp. 219–230, Jan. 2015.
- [167] Y. Han, C. Shi, D. Sun, and J. Zhang, "Research on integrated navigation algorithm based on ranging information of single beacon," *Appl. Acoust.*, vol. 131, pp. 203–209, Feb. 2018.
- [168] C. Renner, "Packet-based ranging with a low-power, low-cost acoustic modem for micro AUVs," in *Proc. 11th ITG Int. Conf. Syst., Commun. Coding*. Hamburg, Germany: VDE, Feb. 2017, pp. 1–6.
- [169] T. A. Sarmiento and R. R. Murphy, "Insights on obstacle avoidance for small unmanned aerial systems from a study of flying animal behavior," *Robot. Auton. Syst.*, vol. 99, pp. 17–29, Jan. 2018.
- [170] F. Menna, P. Agraftiotis, and A. Georgopoulos, "State of the art and applications in archaeological underwater 3D recording and mapping," *J. Cultur. Heritage*, Oct. 2018.
- [171] A. Al-Kaff, D. Martín, F. García, A. d. I. Escalera, and J. María Armingol, "Survey of computer vision algorithms and applications for unmanned aerial vehicles," *Expert Syst. Appl.*, vol. 92, pp. 447–463, Feb. 2018.
- [172] J. Gao, A. A. Proctor, Y. Shi, and C. Bradley, "Hierarchical model predictive image-based visual servoing of underwater vehicles with adaptive neural network dynamic control," *IEEE Trans. Cybern.*, vol. 46, no. 10, pp. 2323–2334, Oct. 2016.
- [173] N. Palomeras, N. Hurtós, M. Carreras, and P. Ridao, "Autonomous mapping of underwater 3-D structures: from view planning to execution," *IEEE Robot. Autom. Lett.*, vol. 3, no. 3, pp. 1965–1971, Jul. 2018.
- [174] P. Norgren and R. Skjetne, "A multibeam-based SLAM algorithm for iceberg mapping using AUVs," *IEEE Access*, vol. 6, pp. 26 318–26 337, Apr. 2018.
- [175] M. Føre, K. Frank, T. Dempster, J. A. Alfreidsen, and E. Høy, "Biomonitoring using tagged sentinel fish and acoustic telemetry in commercial salmon aquaculture: a feasibility study," *Aquacult. Eng.*, vol. 78, pp. 163–172, Aug. 2017.
- [176] J. Schmidhuber, "Deep learning in neural networks: an overview," *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2015.
- [177] Y. Noh, B. Zhang, and D. D. Lee, "Generative local metric learning for nearest neighbor classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 1, pp. 106–118, Jan. 2018.
- [178] E. Adeli, K. Thung, L. An, G. Wu, F. Shi, T. Wang, and D. Shen, "Semi-supervised discriminative classification robust to sample-outliers and feature-noises," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 2, pp. 515–522, Feb. 2019.
- [179] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006.
- [180] Y. LeCun, Y. Bengio *et al.*, "Convolutional networks for images, speech, and time series," *Handbook Brain Theory Neural Netw.*, vol. 3361, no. 10, pp. 255–258, Apr. 1995.
- [181] K. Kim and H. Myung, "Autoencoder-combined generative adversarial networks for synthetic image data generation and detection of jellyfish swarm," *IEEE Access*, vol. 6, pp. 54 207–54 214, Sep. 2018.
- [182] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proc. Empirical Methods Natural Language Process*. Seattle, USA: Association for Computational Linguistics, Oct. 2013, pp. 1631–1642.
- [183] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.
- [184] D. Ravi, C. Wong, F. Deligianni, M. Berthelot, J. Andreu-Perez, B. Lo, and G. Yang, "Deep learning for health informatics," *IEEE J. Biomed. Health Inform.*, vol. 21, no. 1, pp. 4–21, Jan. 2017.
- [185] L. Xiao, G. Sheng, X. Wan, W. Su, and P. Cheng, "Learning-based PHY-layer authentication for underwater sensor networks," *IEEE Commun. Lett.*, vol. 23, no. 1, pp. 60–63, Jan. 2019.
- [186] C. C. Aggarwal, *Data mining: the textbook*. Springer, 2015.
- [187] S. Gupta and S. Roy, "Medav filter—filter for removal of image noise with the combination of median and average filters," in *Proc. Recent Trends Signal Image Process*. Singapore: Springer, May 2018, pp. 11–19.
- [188] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An introduction to statistical learning*, 7th ed. Springer, Jun. 2017.
- [189] M. Wang, X. Liu, Y. Gao, X. Ma, and N. Q. Soomro, "Superpixel segmentation: a benchmark," *Signal Process.: Image Commun.*, vol. 56, pp. 28–39, Aug. 2017.
- [190] C. Li, J. Guo, and C. Guo, "Emerging from water: underwater image color correction based on weakly supervised color transfer," *IEEE Signal Process. Lett.*, vol. 25, no. 3, pp. 323–327, Mar. 2018.
- [191] Y. Peng and P. C. Cosman, "Underwater image restoration based on image blurriness and light absorption," *IEEE Trans. Image Process.*, vol. 26, no. 4, pp. 1579–1594, Apr. 2017.
- [192] N. Wang, H. Zheng, and B. Zheng, "Underwater image restoration via maximum attenuation identification," *IEEE Access*, vol. 5, pp. 18 941–18 952, Sep. 2017.
- [193] M. Li, J. Liu, W. Yang, X. Sun, and Z. Guo, "Structure-revealing low-light image enhancement via robust retinex model," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 2828–2841, Jun. 2018.
- [194] Y. Wang, N. Li, Z. Li, Z. Gu, H. Zheng, B. Zheng, and M. Sun, "An imaging-inspired no-reference underwater color image quality assessment metric," *Comput. Electr. Eng.*, vol. 70, pp. 904–913, Aug. 2018.
- [195] K. Panetta, C. Gao, and S. Agaian, "Human-visual-system-inspired underwater image quality measures," *IEEE J. Ocean. Eng.*, vol. 41, no. 3, pp. 541–551, Jul. 2016.
- [196] M. Yang and A. Sowmya, "An underwater color image quality evaluation metric," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 6062–6071, Dec. 2015.
- [197] C. O. Ancuti, C. Ancuti, C. D. Vleeschouwer, and P. Bekaert, "Color balance and fusion for underwater image enhancement," *IEEE Trans. Image Process.*, vol. 27, no. 1, pp. 379–393, Jan. 2018.
- [198] J. Rife and S. M. Rock, "Segmentation methods for visual tracking of deep-ocean jellyfish using a conventional camera," *IEEE J. Ocean. Eng.*, vol. 28, no. 4, pp. 595–608, Oct. 2003.
- [199] D. M. Kocak, N. d. V. Lobo, and E. A. Widder, "Computer vision techniques for quantifying, tracking, and identifying bioluminescent plankton," *IEEE J. Ocean. Eng.*, vol. 24, no. 1, pp. 81–95, Jan. 1999.
- [200] T. Schoening, "Automated detection in benthic images for megafauna classification and marine resource exploration: supervised and unsupervised methods for classification and regression tasks in benthic images with efficient integration of expert knowledge," Ph.D. dissertation, Der Technischen Fakultät der Universität Bielefeld, 2015.
- [201] M. C. Chuang, J. N. Hwang, K. Williams, and R. Towler, "Automatic fish segmentation via double local thresholding for trawl-based underwater camera systems," in *Proc. 18th Int. Conf. Image Process.* Brussels, Belgium: IEEE, Sep. 2011, pp. 3145–3148.
- [202] T. Schoening, M. Bergmann, J. Ontrup, J. Taylor, J. Dannheim, J. Gutt, A. Purser, and T. W. Nattkemper, "Semi-automated image analysis for the assessment of megafaunal densities at the arctic deep-sea observatory HAUSGARTEN," *PLoS One*, vol. 7, no. 6, p. 38179, Jun. 2012.
- [203] L. Tong, K. Kramer, D. B. Goldgof, L. O. Hall, S. Samson, A. Remsen, and T. Hopkins, "Recognizing plankton images from the shadow image particle profiling evaluation recorder," *IEEE Trans. Syst., Man, Cybern., Part B (Cybern.)*, vol. 34, no. 4, pp. 1753–1762, Aug. 2004.
- [204] Q. Hu and C. Davis, "Accurate automatic quantification of taxa-specific plankton abundance using dual classification with correction," *Marine Ecol. Prog. Series*, vol. 306, pp. 51–61, Jan. 2006.
- [205] A. Purser, M. Bergmann, T. Lundälv, and T. Nattkemper, "Use of machine-learning algorithms for the automated detection of cold-water coral habitats: a pilot study," *Marine Ecol. Prog. Series*, vol. 397, p. 241, Dec. 2009.
- [206] X. Li, J. Song, F. Zhang, X. Ouyang, and S. U. Khan, "MapReduce-based fast fuzzy c-means algorithm for large-scale underwater image segmentation," *Future Generat. Comput. Syst.*, vol. 65, pp. 90–101, Dec. 2016.
- [207] Z. Peizhe and L. Canping, "Region-based color image segmentation of fishes with complex background in water," in *Proc. Int. Conf. Comput. Sci. Autom. Eng.*, vol. 1. Shanghai, China: IEEE, Jun. 2011, pp. 596–600.

- [208] S. L. Benfield, H. M. Guzman, J. M. Mair, and J. A. T. Young, "Mapping the distribution of coral reefs and associated sublittoral habitats in Pacific Panama: a comparison of optical satellite sensors and classification methodologies," *Int. J. Remote Sens.*, vol. 28, no. 22, pp. 5047–5070, Oct. 2007.
- [209] M. C. Chuang, J. N. Hwang, and K. Williams, "A feature learning and object recognition framework for underwater fish images," *IEEE Trans. Image Process.*, vol. 25, no. 4, pp. 1862–1872, Apr. 2016.
- [210] P. X. Huang, B. J. Boom, and R. B. Fisher, "Hierarchical classification with reject option for live fish recognition," *Mach. Vis. Appl.*, vol. 26, no. 1, pp. 89–102, Jan. 2015.
- [211] M. Chambah, D. Semani, A. Renouf, P. Coutellemont, and A. Rizzi, "Underwater color constancy: enhancement of automatic live fish recognition," in *Proc. 16th Annual Symp. Electron. Imaging*. San Jose, USA: International Society for Optics and Photonics, Dec. 2003, pp. 157–168.
- [212] D. Zhang, G. Kopanas, C. Desai, S. Chai, and M. Piacentino, "Unsupervised underwater fish detection fusing flow and objectiveness," in *Proc. IEEE Winter Appl. Comput. Vis. Worksh.* Lake Placid, USA: IEEE, Mar. 2016, pp. 1–7.
- [213] S. Pons, J. Piera, and J. Aguzzi, "Video-image processing applied to the analysis of the behaviour of deep-water lobsters (*Nephrops Norvegicus*)," in *Proc. Ocean*. Sydney, Australia: IEEE, May 2010, pp. 1–4.
- [214] G. W. Flake, R. E. Tarjan, and K. Tsioutsoulouklis, "Graph clustering and minimum cut trees," *Internet Math.*, vol. 1, no. 4, pp. 385–408, 2004.
- [215] A. Lüdtkke, K. Jerosch, O. Herzog, and M. Schlüter, "Development of a machine learning technique for automatic analysis of seafloor image data: case example, Pogonophora coverage at mud volcanoes," *Comput. Geosci.*, vol. 39, pp. 120–128, Feb. 2012.
- [216] M. Pugh, B. Tiddeman, H. Dee, and P. Hughes, "Towards automated classification of seabed substrates in underwater video," in *Proc. ICPR Worksh. Comput. Vis. Anal. Underwater Imagery*. Stockholm, Sweden: IEEE, Aug. 2014, pp. 9–16.
- [217] C. Spampinato, Y.-H. Chen-Burger, G. Nadarajan, and R. Fisher, "Detecting, tracking and counting fish in low quality unconstrained underwater videos," in *Proc. 3rd Int. Conf. Comput. Vis. Theory Appl.*, vol. 2, Madeira, Portugal, 2008, pp. 514–519.
- [218] M. M. M. Fouad, H. M. Zawbaa, N. El-Bendary, and A. E. Hassanien, "Automatic Nile Tilapia fish classification approach using machine learning techniques," in *Proc. 13th Int. Conf. Hybrid Intell. Syst. Gammarth, Tunisia*: IEEE, Dec. 2013, pp. 173–178.
- [219] F. Bonin-Font, M. M. Campos, and G. O. Codina, "Towards visual detection, mapping and quantification of *Posidonia oceanica* using a lightweight auv," *IFAC-PapersOnLine*, vol. 49, no. 23, pp. 500–505, 2016.
- [220] M. C. Chuang, J. N. Hwang, F. F. Kuo, M. K. Shan, and K. Williams, "Recognizing live fish species by hierarchical partial classification based on the exponential benefit," in *Proc. Int. Conf. Image Process.* Paris, France: IEEE, Oct. 2014, pp. 5232–5236.
- [221] M. Elawady, "Sparse coral classification using deep convolutional neural networks," *Comput. Research Repos.*, Nov. 2015.
- [222] P. R. Ambika and A. B. Malakreddy, "Review of existing research contribution toward dimensional reduction methods in high-dimensional data," in *Proc. Int. Conf. Comput. Netw. Commun. Technol.*, Seoul, South Korea, Jun. 2018, pp. 21–23.
- [223] S. Gepshtein and Y. Keller, "Iterative spectral independent component analysis," *Signal Process.*, vol. 155, pp. 368–376, Feb. 2019.
- [224] M. Mahdianpari, B. Salehi, F. Mohammadimanesh, B. Brisco, S. Mahdavi, M. Amani, and J. E. Granger, "Fisher linear discriminant analysis of coherency matrix for wetland classification using PolSAR imagery," *Remote Sens. Env.*, vol. 206, pp. 300–317, Mar. 2018.
- [225] X. Tang, Y. Dai, and Y. Xiang, "Feature selection based on feature interactions with application to text categorization," *Expert Syst. Appl.*, vol. 120, pp. 207–216, Apr. 2019.
- [226] R. Zhao and K. Mao, "Fuzzy bag-of-words model for document representation," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 2, pp. 794–804, Apr. 2018.
- [227] L. H. Ceze, *Encyclopedia of parallel computing*. Springer, 2011, ch. Shared-Memory Multiprocessors, pp. 1810–1812.
- [228] J. Lee, C. Kim, S. Kang, D. Shin, S. Kim, and H. Yoo, "UNPU: an energy-efficient deep neural network accelerator with fully variable weight bit precision," *IEEE J. Solid-State Circuits*, vol. 54, no. 1, pp. 173–185, Jan. 2019.
- [229] S. I. Venieris and C. Bouganis, "fpgaConvNet: mapping regular and irregular convolutional neural networks on FPGAs," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 2, pp. 326–342, Feb. 2019.
- [230] T. Manderson and G. Dudek, "GPU-assisted learning on an autonomous marine robot for vision-based navigation and image understanding," in *Proc. Ocean*. Charleston, USA: IEEE, Oct. 2018, pp. 1–6.
- [231] G. F. Coulouris, J. Dollimore, and T. Kindberg, *Distributed systems: concepts and design*, 5th ed. Pearson Education, 2011.
- [232] D. Xia, H. Li, B. Wang, Y. Li, and Z. Zhang, "A map reduce-based nearest neighbor approach for big-data-driven traffic flow prediction," *IEEE Access*, vol. 4, pp. 2920–2934, Jun. 2016.
- [233] Z. Xu, B. Liu, S. Zhe, H. Bai, Z. Wang, and J. Neville, "Variational random function model for network modeling," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 1, pp. 318–324, Jan. 2019.
- [234] Y. Ming, E. Zhu, M. Wang, Y. Ye, X. Liu, and J. Yin, "DMP-ELMs: data and model parallel extreme learning machines for large-scale learning tasks," *Neurocomp.*, vol. 320, pp. 85–97, Dec. 2018.
- [235] S. Garcia, D. Jeon, C. Louie, and M. B. Taylor, "The Kremlin oracle for sequential code parallelization," *IEEE Micro*, vol. 32, no. 4, pp. 42–53, Jul. 2012.
- [236] E. Schwartz, R. Giryes, and A. M. Bronstein, "DeepISP: toward learning an end-to-end image processing pipeline," *IEEE Trans. Image Process.*, vol. 28, no. 2, pp. 912–923, Feb. 2019.
- [237] M. Modasshir, A. Q. Li, and I. Rekleitis, "Deep neural networks: a comparison on different computing platforms," in *Proc. 15th Conf. Comput. Robot Vis.* Toronto, Canada: IEEE, May 2018, pp. 383–389.
- [238] H. Kim, J. Koo, D. Kim, B. Park, Y. Jo, H. Myung, and D. Lee, "Vision-based real-time obstacle segmentation algorithm for autonomous surface vehicle," *IEEE Access*, vol. 7, pp. 179 420–179 428, Dec. 2019.
- [239] V. Pallayil, M. Chitre, S. Kuselan, A. Raichur, M. Ignatius, and J. R. Potter, "Development of a second-generation underwater acoustic ambient noise imaging camera," *IEEE J. Ocean. Eng.*, vol. 41, no. 1, pp. 175–189, Jan. 2016.
- [240] S. Karabchevsky, D. Kahana, O. Ben-Harush, and H. Guterman, "FPGA-based adaptive speckle suppression filter for underwater imaging sonar," *IEEE J. Ocean. Eng.*, vol. 36, no. 4, pp. 646–657, Oct. 2011.
- [241] C. Szegedy, S. Loffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proc. 31st AAAI Conf. Artif. Intell.* San Francisco, USA: AAAI, Feb. 2017.
- [242] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.* Las Vegas, USA: IEEE, 2016, pp. 770–778.
- [243] A. Canziani, A. Paszke, and E. Culurciello, "An analysis of deep neural network models for practical applications," *Comput. Research Repos.*, Apr. 2017.
- [244] S. Zagoruyko, "ImageNet-validation.torch," Jun. 2016. [Online]. Available: <https://github.com/szagoruyko/imagenet-validation.torch>
- [245] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Adv. Neural Inf. Process. Syst.* Curran Associates, Inc., 2012, pp. 1097–1105.
- [246] M. Lin, Q. Chen, and S. Yan, "Network in network," *ArXiv*, Dec. 2013.
- [247] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *ArXiv*, Apr. 2015.
- [248] A. Paszke and A. Jackson, "Torch-opCounter," 2016. [Online]. Available: <https://github.com/apaszke/torch-opCounter>
- [249] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, p. 533, Oct. 1986.
- [250] J. Adler and O. Öktem, "Solving ill-posed inverse problems using iterative deep neural networks," *Inver. Probl.*, vol. 33, no. 12, p. 124007, May 2017.
- [251] Y. Li, H. Lu, J. Li, X. Li, Y. Li, and S. Serikawa, "Underwater image de-scattering and classification by deep neural network," *Comput. Electr. Eng.*, vol. 54, pp. 68–77, Aug. 2016.
- [252] W. Zhang, K. Ma, J. Yan, D. Deng, and Z. Wang, "Blind image quality assessment using a deep bilinear convolutional neural network," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 1, pp. 36–47, Jan. 2020.
- [253] Y. Yan, H. Hao, B. Xu, J. Zhao, and F. Shen, "Image clustering via deep embedded dimensionality reduction and probability-based triplet loss," *IEEE Trans. Image Process.*, vol. 29, pp. 5652–5661, Apr. 2020.
- [254] H. Qin, X. Li, J. Liang, Y. Peng, and C. Zhang, "DeepFish: accurate underwater live fish recognition with a deep architecture," *Neurocomp.*, vol. 187, pp. 49–58, Apr. 2016.

- [255] N. Wahidin, V. P. Siregar, B. Nababan, I. Jaya, and S. Wouthuyzen, "Object-based image analysis for coral reef benthic habitat mapping with several classification algorithms," *Proc. Env. Sci.*, vol. 24, pp. 222–227, Dec. 2015.
- [256] G. Huo, Z. Wu, and J. Li, "Underwater object classification in sidescan sonar images using deep transfer learning and semisynthetic training data," *IEEE Access*, vol. 8, pp. 47 407–47 418, Mar. 2020.
- [257] L. Jin, H. Liang, and C. Yang, "Accurate underwater ATR in forward-looking sonar imagery using deep convolutional neural networks," *IEEE Access*, vol. 7, pp. 125 522–125 531, Sep. 2019.
- [258] S. W. Perry and L. Guan, "A recurrent neural network for detecting objects in sequences of sector-scan sonar images," *IEEE J. Ocean. Eng.*, vol. 29, no. 3, pp. 857–871, Jul. 2004.
- [259] H. Singh, J. Howland, and O. Pizarro, "Advances in large-area photomosaicking underwater," *IEEE J. Ocean. Eng.*, vol. 29, no. 3, pp. 872–886, Jul. 2004.
- [260] R. Nian, B. He, B. Zheng, M. van Heeswijk, Q. Yu, Y. Miche, and A. Lendasse, "Extreme learning machine towards dynamic model hypothesis in fish ethology research," *Neurocomp.*, vol. 128, pp. 273–284, Mar. 2014.
- [261] S. Luo, X. Li, D. Wang, J. Li, and C. Sun, "Automatic fish recognition and counting in video footage of fishery operations," in *Proc. Int. Conf. Comput. Intell. Commun. Netw.* Jabalpur, India: IEEE, Dec. 2015, pp. 296–299.
- [262] K. Zhang, X. Geng, and X. Yan, "Prediction of 3-D ocean temperature by multilayer convolutional LSTM," *IEEE Geosci. Remote Sens. Lett.*, pp. 1–5, Jan. 2020.
- [263] J. Wang, G. Wu, L. Wan, Y. Sun, and D. Jiang, "Recurrent neural network applied to fault diagnosis of underwater robots," in *Proc. Int. Conf. Intell. Comput. Intell. Syst.*, vol. 1. Shanghai, China: IEEE, Nov. 2009, pp. 593–598.
- [264] Y. Shan, Z. Yan, and J. Wang, "Model predictive control of underwater gliders based on a one-layer recurrent neural network," in *Proc. 6th Int. Conf. Adv. Comput. Intell.* Hangzhou, China: IEEE, Oct. 2013, pp. 328–333.
- [265] Y. Zhang, P. J. Thorburn, W. Xiang, and P. Fitch, "SSIM—a deep learning approach for recovering missing time series sensor data," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6618–6628, Aug. 2019.
- [266] Y. Gou, T. Zhang, J. Liu, L. Wei, and J. Cui, "DeepOcean: a general deep learning framework for spatio-temporal ocean sensing data prediction," *IEEE Access*, vol. 8, pp. 79 192–79 202, Apr. 2020.
- [267] Y. Yang, J. Dong, X. Sun, E. Lima, Q. Mu, and X. Wang, "A CFCC-LSTM model for sea surface temperature prediction," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 2, pp. 207–211, Dec. 2018.
- [268] J. Xie, J. Zhang, J. Yu, and L. Xu, "An adaptive scale sea surface temperature predicting method based on deep learning with attention mechanism," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 5, pp. 740–744, Aug. 2020.
- [269] H. Kim and J. Y. Kim, "Environmental sound event detection in wireless acoustic sensor networks for home telemonitoring," *China Commun.*, vol. 14, no. 9, pp. 1–10, Sep. 2017.
- [270] A. Zyner, S. Worrall, and E. Nebot, "A recurrent neural network solution for predicting driver intention at unsignalized intersections," *IEEE Robot. Autom. Lett.*, vol. 3, no. 3, pp. 1759–1764, Jul. 2018.
- [271] F. S. Marzano, G. Rivolta, E. Coppola, B. Tomassetti, and M. Verdecchia, "Rainfall nowcasting from multisatellite passive-sensor images using a recurrent neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 11, pp. 3800–3812, Nov. 2007.
- [272] A. Pras, J. Schonwalder, M. Burgess, O. Festor, G. M. Perez, R. Stadler, and B. Stiller, "Key research challenges in network management," *IEEE Commun. Mag.*, vol. 45, no. 10, pp. 104–110, Oct. 2007.
- [273] H. Urunov, S. Y. Shin, S. H. Park, and K. Yi, "U-SNMP for the internet of underwater things," *Int. J. Control Autom.*, vol. 10, pp. 199–216, Jan. 2017.
- [274] A. K. Y. Wong, An Chi Chen, N. Paramesh, and P. Rav, "Ontology mapping for network management systems," in *Proc. Netw. Operat. Manag. Symp.*, vol. 1. Seoul, South Korea: IEEE/IFIP, Apr. 2004, pp. 885–886.
- [275] H. Mukhtar, K. Kang-Myo, S. A. Chaudhry, A. H. Akbar, K. Ki-Hyung, and S.-W. Yoo, "LNMP: management architecture for IPv6 based low-power wireless personal area networks (6LoWPAN)," in *Proc. Netw. Operat. Manag. Symp.* Salvador, Brazil: IEEE, Aug. 2008, pp. 417–424.
- [276] S. Sharma, D. Staessens, D. Colle, D. Palma, J. Gonçalves, R. Figueiredo, D. Morris, M. Pickavet, and P. Demeester, "Implementing quality of service for the software defined networking enabled future internet," in *Proc. 3rd Europ. Worksh. Softw. Defin. Netw.* London, UK: IEEE, Sep. 2014, pp. 49–54.
- [277] J. Bitto, R. Bahr, J. G. Hester, S. A. Nauroze, A. Georgiadis, and M. M. Tentzeris, "A novel solar and electromagnetic energy harvesting system with a 3-D printed package for energy efficient internet-of-things wireless sensors," *IEEE Trans. Microw. Theory Techn.*, vol. 65, no. 5, pp. 1831–1842, May 2017.
- [278] C. Alippi, R. Camplani, C. Galperti, and M. Roveri, "A robust, adaptive, solar-powered WSN framework for aquatic environmental monitoring," *IEEE Sensors J.*, vol. 11, no. 1, pp. 45–55, Jan. 2011.
- [279] K. B. Joshi, J. H. Costello, and S. Priya, "Estimation of solar energy harvested for autonomous jellyfish vehicles (AJVs)," *IEEE J. Ocean. Eng.*, vol. 36, no. 4, pp. 539–551, Oct. 2011.
- [280] P. P. Jenkins, S. Messenger, K. M. Trautz, S. I. Maximenko, D. Goldstein, D. Scheiman, R. Hoheisel, and R. J. Walters, "High-bandgap solar cells for underwater photovoltaic applications," *IEEE J. Photovolt.*, vol. 4, no. 1, pp. 202–207, Jan. 2014.
- [281] D. M. Toma, J. del Rio, M. Carbonell-Ventura, and J. M. Masalles, "Underwater energy harvesting system based on plucked-driven piezoelectrics," in *Proc. Ocean.* Genova, Italy: IEEE, May 2015, pp. 1–5.
- [282] F. U. Qureshi, A. Muhtaroglu, and K. Tuncay, "Near-optimal design of scalable energy harvester for underwater pipeline monitoring applications with consideration of impact to pipeline performance," *IEEE Sensors J.*, vol. 17, no. 7, pp. 1981–1991, Apr. 2017.
- [283] A. Bereketli and S. Bilgen, "Remotely powered underwater acoustic sensor networks," *IEEE Sensors J.*, vol. 12, no. 12, pp. 3467–3472, Dec. 2012.
- [284] T. Kan, Y. Zhang, Z. Yan, P. P. Mercier, and C. C. Mi, "A rotation-resilient wireless charging system for lightweight autonomous underwater vehicles," *IEEE Trans. Veh. Technol.*, vol. 67, no. 8, pp. 6935–6942, Aug. 2018.
- [285] L. Jing, C. He, J. Huang, and Z. Ding, "Energy management and power allocation for underwater acoustic sensor network," *IEEE Sensors J.*, vol. 17, no. 19, pp. 6451–6462, Oct. 2017.
- [286] Z. Wu, J. Yu, J. Yuan, M. Tan, and S. Qi, "Gliding motion regulation of a robotic dolphin based on a controllable fluke," *IEEE Trans. Ind. Electron.*, vol. 67, no. 4, pp. 2945–2953, Apr. 2020.
- [287] J. D. Hem, *Water analysis. inorganic species.* Academic Press, 2012, vol. 1, no. 1, ch. Conductance: a collective measure of dissolved ions, pp. 137–161.
- [288] M. T. Islam, M. N. Rahman, M. S. J. Singh, and M. Samsuzzaman, "Detection of salt and sugar contents in water on the basis of dielectric properties using microstrip antenna-based sensor," *IEEE Access*, vol. 6, pp. 4118–4126, Jan. 2018.
- [289] E. Vorathin, Z. M. Hafizi, A. M. Aizzuddin, M. K. A. Zaini, and K. S. Lim, "A novel temperature-insensitive hydrostatic liquid-level sensor using chirped FBG," *IEEE Sensors J.*, vol. 19, no. 1, pp. 157–162, Jan. 2019.
- [290] Y. Wang, S. M. S. M. Rajib, C. Collins, and B. Grieve, "Low-cost turbidity sensor for low-power wireless monitoring of fresh-water courses," *IEEE Sensors J.*, vol. 18, no. 11, pp. 4689–4696, Jun. 2018.
- [291] W. D. Kirkey, J. S. Bonner, and C. B. Fuller, "Low-cost submersible turbidity sensors using low-frequency source light modulation," *IEEE Sensors J.*, vol. 18, no. 22, pp. 9151–9162, Nov. 2018.
- [292] C. Liu, Y. Li, and M. Xu, "An integrated detection and location model for leakages in liquid pipelines," *J. Petrol. Sci. Eng.*, vol. 175, pp. 852–867, Apr. 2019.
- [293] P. Wang, X. Tian, T. Peng, and Y. Luo, "A review of the state-of-the-art developments in the field monitoring of offshore structures," *Ocean Eng.*, vol. 147, pp. 148–164, Jan. 2018.
- [294] X. Zhou, Q. Yu, and W. Peng, "Fiber-optic Fabry–Perot pressure sensor for down-hole application," *Opt. Lasers Eng.*, vol. 121, pp. 289–299, Oct. 2019.
- [295] F. Berlinger, J. Dusek, M. Gauci, and R. Nagpal, "Robust maneuverability of a miniature, low-cost underwater robot using multiple fin actuation," *IEEE Robot. Autom. Lett.*, vol. 3, no. 1, pp. 140–147, Jan. 2018.
- [296] E. A. Makled, A. Yadav, O. A. Dobre, and R. D. Haynes, "Hierarchical full-duplex underwater acoustic network: a NOMA approach," in *Proc. Ocean.* Charleston, USA: IEEE, Oct. 2018, pp. 1–6.
- [297] Z. M. Fadlullah, F. Tang, B. Mao, J. Liu, and N. Kato, "On intelligent traffic control for large-scale heterogeneous networks: a value matrix-based deep learning approach," *IEEE Commun. Lett.*, vol. 22, no. 12, pp. 2479–2482, Dec. 2018.
- [298] J. Xie, F. R. Yu, T. Huang, R. Xie, J. Liu, C. Wang, and Y. Liu, "A survey of machine learning techniques applied to software defined

- networking (SDN): research issues and challenges,” *IEEE Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 393–430, Jan. 2019.
- [299] B. Mao, F. Tang, Z. M. Fadlullah, N. Kato, O. Akashi, T. Inoue, and K. Mizutani, “A novel non-supervised deep-learning-based network traffic control method for software defined wireless networks,” *IEEE Wireless Commun.*, vol. 25, no. 4, pp. 74–81, Aug. 2018.
- [300] Y. Su, L. Guo, Z. Jin, and X. Fu, “A mobile-beacon based iterative localization mechanism in large-scale underwater acoustic sensor networks,” *IEEE Internet Things J.*, Sep. 2020.
- [301] M. Roopak, G. Yun Tian, and J. Chambers, “Deep learning models for cyber security in IoT networks,” in *Proc. 9th Annual Comput. Commun. Worksh. Conf.* Las Vegas, USA: IEEE, Jan. 2019, pp. 0452–0457.
- [302] L. Xiao, D. Jiang, Y. Chen, W. Su, and Y. Tang, “Reinforcement-learning-based relay mobility and power allocation for underwater sensor networks against jamming,” *IEEE J. Ocean. Eng.*, vol. 45, no. 3, pp. 1148–1156, Jul. 2020.
- [303] Y. Zhang, P. Li, and X. Wang, “Intrusion detection for IoT based on improved genetic algorithm and deep belief network,” *IEEE Access*, vol. 7, pp. 31 711–31 722, Mar. 2019.
- [304] Y. Li, S. Wang, C. Jin, Y. Zhang, and T. Jiang, “A survey of underwater magnetic induction communications: fundamental issues, recent advances, and challenges,” *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2466–2487, Feb. 2019.
- [305] F. Zafari, A. Gkelias, and K. K. Leung, “A survey of indoor localization systems and technologies,” *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2568–2599, Apr. 2019.
- [306] H. Wu, J. Chen, X. Liu, Y. Xiao, M. Wang, Y. Zheng, and Y. Rao, “One-dimensional CNN-based intelligent recognition of vibrations in pipeline monitoring with DAS,” *J. Lightw. Technol.*, vol. 37, no. 17, pp. 4359–4366, Sep. 2019.
- [307] K. Ma, H. Leung, E. Jalilian, and D. Huang, “Fiber-optic acoustic-based disturbance prediction in pipelines using deep learning,” *IEEE Sensors Lett.*, vol. 1, no. 6, pp. 1–4, Dec. 2017.
- [308] J. Pan, Y. Yin, J. Xiong, W. Luo, G. Gui, and H. Sari, “Deep learning-based unmanned surveillance systems for observing water levels,” *IEEE Access*, vol. 6, pp. 73 561–73 571, Nov. 2018.
- [309] M. R. Hedayati, A. A. Amidian, S. A. Sadr, and A. Razazan, “Intelligent ship hull inspection and NDT using ROV based flux leakage expert system,” in *Proc. 2nd Int. Conf. Comput. Intell., Model. Simul.* Tuban, Indonesia: IEEE, Sep. 2010, pp. 412–415.
- [310] C. Li, X. Peng, J. Liu, C. Wang, S. Fan, and S. Cao, “D-shaped fiber bragg grating ultrasonic hydrophone with enhanced sensitivity and bandwidth,” *J. Lightw. Technol.*, vol. 37, no. 9, pp. 2100–2108, May 2019.
- [311] X. E. Gros, P. Strachan, and D. Lowden, “Fusion of multiprobe NDT data for ROV inspection,” in *Proc. Ocean. Chall. Our Chang. Glob. Env.*, vol. 3. San Diego, USA: IEEE, Oct. 1995, pp. 2046–2050.
- [312] Y. Pang, M. Sun, X. Jiang, and X. Li, “Convolution in convolution for network in network,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 5, pp. 1587–1597, May 2018.
- [313] N. Li, Z. Zheng, S. Zhang, Z. Yu, H. Zheng, and B. Zheng, “The synthesis of unpaired underwater images using a multistyle generative adversarial network,” *IEEE Access*, vol. 6, pp. 54 241–54 257, Sep. 2018.
- [314] J. Li, K. A. Skinner, R. M. Eustice, and M. Johnson-Roberson, “WaterGAN: unsupervised generative network to enable real-time color correction of monocular underwater images,” *IEEE Robot. Autom. Lett.*, vol. 3, no. 1, pp. 387–394, Jan. 2018.
- [315] A. Jafari, A. Ganesan, C. S. K. Thalisetty, V. Sivasubramanian, T. Oates, and T. Mohsenin, “SensorNet: a scalable and low-power deep convolutional neural network for multimodal data classification,” *IEEE Trans. Circuits Syst. I: Reg. Papers*, vol. 66, no. 1, pp. 274–287, Jan. 2019.
- [316] J. Chang and J. Sha, “Prune deep neural networks with the modified $L_{1/2}$ penalty,” *IEEE Access*, vol. 7, pp. 2273–2280, Dec. 2018.
- [317] Y. Tian, B. Liu, X. Su, L. Wang, and K. Li, “Underwater imaging based on LF and polarization,” *IEEE Photon. J.*, vol. 11, no. 1, pp. 1–9, Feb. 2019.
- [318] D. G. Zawada, “Image processing of underwater multispectral imagery,” *IEEE J. Ocean. Eng.*, vol. 28, no. 4, pp. 583–594, Oct. 2003.
- [319] S. Suresh, E. Westman, and M. Kaess, “Through-water stereo SLAM with refraction correction for AUV localization,” *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 692–699, Apr. 2019.



and Engineering, James Cook University, Australia. His research interests are data science, machine learning, RF and microwave engineering, numerical methods, and signal processing.



of IoT Engineering at James Cook University, Cairns, Australia. Due to his instrumental leadership in establishing Australia’s first accredited Internet of Things Engineering degree program, he was inducted into Percy Foundation’s Hall of Fame in October 2018. He is an elected Fellow of the IET in UK and Engineers Australia. He received the TNQ Innovation Award in 2016, and Pearcey Entrepreneurship Award in 2017, and Engineers Australia Cairns Engineer of the Year in 2017. He was a co-recipient of four Best Paper Awards at WiSATS’2019, WCSP’2015, IEEE WCNC’2011, and ICWMC’2009. He has been awarded several prestigious fellowship titles. He was named a Queensland International Fellow (2010–2011) by the Queensland Government of Australia, an Endeavour Research Fellow (2012–2013) by the Commonwealth Government of Australia, a Smart Futures Fellow (2012–2015) by the Queensland Government of Australia, and a JSPS Invitational Fellow jointly by the Australian Academy of Science and Japanese Society for Promotion of Science (2014–2015). He is the Vice Chair of the IEEE Northern Australia Section. He was an Editor for IEEE Communications Letters (2015–2017), and is currently an Associate Editor for IEEE Internet of Things Journal and IEEE Access. He has published over 250 peer-reviewed papers including 3 books and 180 journal articles. He has served in a large number of international conferences in the capacity of General Co-Chair, TPC Co-Chair, Symposium Chair, etc. His research interest includes the Internet of Things, wireless communications, machine learning for IoT data analytics, and computer vision.



Mohammad Jahanbakht (S’18) received the B.Eng., M.Eng., and Ph.D. degrees, all in telecommunication engineering, from the Islamic Azad University of Iran, in 2003, 2005, 2009, respectively. He currently holds an adjunct faculty position at the Department of Electronic Engineering, Shahr-e-Qods Branch, Islamic Azad University, Tehran, Iran. In March 2018, he was awarded a Research Training Program (RTP) scholarship by the Australian Government to start a new project on machine intelligence for marine data at the College of Science

Wei Xiang (S’00–M’04–SM’10) received the B.Eng. and M.Eng. degrees, both in electronic engineering, from the University of Electronic Science and Technology of China, Chengdu, China, in 1997 and 2000, respectively, and the Ph.D. degree in telecommunications engineering from the University of South Australia, Adelaide, Australia, in 2004.

Professor Wei Xiang is Cisco Research Chair of AI and IoT and Director Cisco-La Trobe Centre for AI and IoT at La Trobe University. Previously, he was Foundation Chair and Head of Discipline of IoT Engineering at James Cook University, Cairns, Australia. Due to his instrumental leadership in establishing Australia’s first accredited Internet of Things Engineering degree program, he was inducted into Percy Foundation’s Hall of Fame in October 2018. He is an elected Fellow of the IET in UK and Engineers Australia. He received the TNQ Innovation Award in 2016, and Pearcey Entrepreneurship Award in 2017, and Engineers Australia Cairns Engineer of the Year in 2017. He was a co-recipient of four Best Paper Awards at WiSATS’2019, WCSP’2015, IEEE WCNC’2011, and ICWMC’2009. He has been awarded several prestigious fellowship titles. He was named a Queensland International Fellow (2010–2011) by the Queensland Government of Australia, an Endeavour Research Fellow (2012–2013) by the Commonwealth Government of Australia, a Smart Futures Fellow (2012–2015) by the Queensland Government of Australia, and a JSPS Invitational Fellow jointly by the Australian Academy of Science and Japanese Society for Promotion of Science (2014–2015). He is the Vice Chair of the IEEE Northern Australia Section. He was an Editor for IEEE Communications Letters (2015–2017), and is currently an Associate Editor for IEEE Internet of Things Journal and IEEE Access. He has published over 250 peer-reviewed papers including 3 books and 180 journal articles. He has served in a large number of international conferences in the capacity of General Co-Chair, TPC Co-Chair, Symposium Chair, etc. His research interest includes the Internet of Things, wireless communications, machine learning for IoT data analytics, and computer vision.

Lajos Hanzo (<http://www-mobile.ecs.soton.ac.uk>, https://en.wikipedia.org/wiki/Lajos_Hanzo) (FIEEE’04) received his Master degree and Doctorate in 1976 and 1983, respectively from the Technical University (TU) of Budapest. He was also awarded the Doctor of Sciences (DSc) degree by the University of Southampton (2004) and Honorary Doctorates by the TU of Budapest (2009) and by the University of Edinburgh (2015). He is a Foreign Member of the Hungarian Academy of Sciences and a former Editor-in-Chief of the IEEE

Press. He has served several terms as Governor of both IEEE ComSoc and of VTS. He has published 2000+ contributions at IEEE Xplore, 19 Wiley-IEEE Press books and has helped the fast-track career of 123 PhD students. Over 40 of them are Professors at various stages of their careers in academia and many of them are leading scientists in the wireless industry. He is also a Fellow of the Royal Academy of Engineering (FREng), of the IET and of EURASIP.



Mostafa Rahimi Azghadi (S'07–M'14–SM'19) completed his PhD in Electrical & Electronic Engineering at The University of Adelaide, Australia, earning the Doctoral Research Medal, and the 2015 Adelaide University Alumni Medal. He is currently a senior lecturer at the College of Science and Engineering, James Cook University, Australia, where he researches Machine Learning software and hardware design for a variety of applications including automation, precision agriculture, aquaculture, marine sciences, mining, and medical imaging. His research

has attracted over \$0.7 Million in funding from national and international resources.

Dr. Rahimi Azghadi was the recipient of several accolades including a 2015 South Australia Science Excellence award, a 2016 Endeavour Research Fellowship, a 2017 Queensland Young Tall Poppy Science Award, a 2018 Rising Star ECR Leader Fellowship, a 2019 Fresh Science Queensland finalist, and a 2020 Award for Excellence in Innovation and Change.

TITLE/ AUTHOR	Privacy-Preserving Blockchain-Based Federated Learning for Marine Internet of Things / Z. Qin, J. Ye, J. Meng, B. Lu and L. Wang
SOURCE	IEEE Transactions on Computational Social Systems Volume 9, No. 1, Pages 159-173 (Feb. 2022) https://doi.org/10.1109/TCSS.2021.3100258 (Database: IEEE)

Privacy-Preserving Blockchain-Based Federated Learning for Marine Internet of Things

Zhenquan Qin¹, Member, IEEE, Jin Ye, Jie Meng, Bingxian Lu², Member, IEEE,
and Lei Wang³, Member, IEEE

Abstract—The marine Internet of things (MIoT) is the application of the Internet of things technology in the marine field. Nowadays, with the arrival of the era of big data, the MIoT architecture has been transformed from cloud computing architecture to edge computing architecture. However, due to the lack of trust among edge computing participants, new solutions with higher security need to be proposed. In the current solutions, some use blockchain technology to solve data security problems while some use federated learning technology to solve privacy problems, but these methods neither combine with the special environment of the ocean nor consider the security of task publishers. In this article, we propose a secure sharing method of MIoT data under an edge computing framework based on federated learning and blockchain technology. Combining its special distributed architecture with the MIoT edge computing architecture, federated learning ensures the privacy of nodes. The blockchain serves as a decentralized way, which stores federated learning workers to achieve nontampering and security. We propose a concept of quality and reputation as the metrics of selection for federated learning workers. Meanwhile, we design a quality proof mechanism [proof of quality (PoQ)] and apply it to the blockchain, making the edge nodes recorded in the blockchain more high-quality. In addition, a marine environment model is built in this article, and the analysis based on this model makes the method proposed in this article more applicable to the marine environment. The numerical results obtained from the simulation experiments clearly show that the proposed scheme can significantly improve the learning accuracy under the premise of ensuring the safety and reliability of the marine environment.

Index Terms—Blockchain, edge computing, federated learning, marine Internet of things (MIoT), privacy.

I. INTRODUCTION

WITH the development of Internet of things technology, the concept of marine Internet of things (MIoT)

gradually appears in people's vision to effectively utilize marine resources [1]. The MIoT uses Internet technology to interconnect various marine sensing devices, so as to integrate maritime information and realize the monitoring and systematic management of marine complex data. MIoT is the application of IoT technology in the marine field, and its core technology is IoT technology. Generally, the MIoT collects various parameters related to the ocean through water or underwater sensing equipment, and then sends the data to the ocean observation system, data center, or cloud platform through various communication means [2]. Big data, machine learning, and other related technologies are used on the cloud platform to manage, analyze and utilize ocean data through customized software. However, with the advent of the era of big data, cloud computing mode is gradually facing many challenges such as communication load, network delay, and communication rate. The birth of moving edge computing solves these problems [3]. Mobile edge computing relieves the pressure on computing and communication by sinking computing and storage to the edge of the network. Therefore, the field of mobile edge computing has become a research hotspot. Some research studies have proposed a task scheduling algorithm based on vehicle edge computing (VEC) to improve efficiency and save resources [4], [5]. However, data sharing in MIoT faces two key challenges due to the lack of trust among edge computing participants and users' continuous concern over privacy [6]. First, edge nodes need to share data effectively, even if communication between nodes is unreliable. Further research is needed on how to improve the efficiency and reliability of data sharing. Second, because users are increasingly concerned about privacy, they want to be anonymous or not reveal their privacy when sharing data. This is still the case even in the MIoT. Therefore, in MIoT, more secure, reliable, and intelligent solutions are needed on how to effectively and securely share data [7].

Some researchers proposed an information-aware heuristic algorithm to optimize the age of information (AoI) index in mobile edge network [8]. While, in recent years, blockchain, with its characteristics of anonymity, tamper-free, and distribution, provides a safe and reliable solution for edge computing between multiple untrusted parties, and has become a new research hotspot in the field of edge computing [9], [10]. The essence of blockchain is a distributed ledger. Its biggest feature is that it transforms from a traditional centralized scheme to distributed network structure, and guarantees the security of data on the chain through encryption technologies such as asymmetric encryption [11].

Manuscript received March 12, 2021; revised May 31, 2021 and July 14, 2021; accepted July 21, 2021. Date of publication August 5, 2021; date of current version January 31, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 61902052; in part by the National Key Research and Development Plan under Grant 2017YFC0821003-2; in part by the Science and Technology Major Industrial Project of Liaoning Province under Grant 2020JH1/10100013; in part by the Dalian Science and Technology Innovation Fund under Grant 2019J11CY004 and Grant 2020JJ26GX037; and in part by the Fundamental Research Funds for the Central Universities under Grant DUT20ZD210 and Grant DUT20TD107. (Corresponding author: Bingxian Lu.)

Zhenquan Qin is with the Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province and the School of Software, Dalian University of Technology, Dalian 116620, China (e-mail: qzq@dlut.edu.cn).

Jin Ye, Jie Meng, Bingxian Lu, and Lei Wang are with the School of Software, Dalian University of Technology, Dalian 116620, China (e-mail: yejin@mail.dlut.edu.cn; mengjie812201@mail.dlut.edu.cn; bingxian.lu@dlut.edu.cn; lei.wang@dlut.edu.cn).

Digital Object Identifier 10.1109/TCSS.2021.3100258

Some researchers are concerned about the privacy and security of blockchain. Ning *et al.* [12] proposed a deep reinforcement learning (DRL) method to solve the privacy and security of blockchain. However, blockchain technology faces some challenges. First, the computing and authentication efficiency of blockchain is low [10], [13]. Second, further research is still needed on how to realize network edge intelligence based on blockchain. As an emerging technology with powerful functions in data processing in recent years, federated learning is more suitable for integrated application with blockchain in the field of edge computing due to its distributed and local computing characteristics [14]. In addition, federated learning can expand the entire training dataset by introducing more user participation, thus improving the quality of the overall model [15]. Therefore, federated learning is more suitable for integration with blockchain to improve the security and efficiency of the frontier intelligence field. As a result, federated learning improves computational efficiency and blockchain improves data reliability. In this architecture, for the MIoT scenario, the combination of federated learning and blockchain is clearly more suitable for the marine environment because its edge nodes have lower power consumption and greater mobility.

While the combination of blockchain and the federated learning framework can bring together the best of both technologies, many practical issues need to be considered when they are applied to specific scenarios. For example, Lu *et al.* [16] proposed a secure data sharing architecture for blockchain authorization, which combines data sharing issues with machine learning issues, maintains data privacy well through sharing data model rather than revealing actual data, and integrates federated learning in the licensing blockchain. In this scheme, federated learning model and parameter transfer processes are stored and shared through blockchain to ensure the security of the shared process. However, without considering the selection of federated learning workers, low-quality or malicious nodes may be involved in the process of federated learning, which wastes the integration of resources and affects the overall aggregation of the model. Kang *et al.* [17] proposed a scheme based on reputation as a reliable measure to select trusted workers for federated learning. Using a multiweight subjective logic model (MSL), the scheme proposes an efficient reputation calculation scheme according to the interaction history of task publishers and recommended reputation opinions, and uses the blockchain deployed on edge nodes for reputation management. However, this method only considers the reputation of the edge nodes and does not consider the parameter quality of nodes, so high-quality nodes cannot be selected more efficiently. To sum up, data security sharing technology based on federated learning and blockchain technology currently has the following defects.

- 1) The energy of edge devices is limited, so it is not suitable to run proof of work (PoW) [18] mechanism, which consumes a lot of computing energy in the blockchain.
- 2) There may be malicious nodes sabotaging the learning of worker nodes of federated learning.

- 3) For the marine environment, its special complexity and randomness need to be considered, and it needs to fit the reality.

Aiming at the above problems, this article proposes a data security sharing method under the edge computing framework based on federated learning and blockchain technology for the MIoT environment. The method combines the characteristics of federated learning and blockchain to improve the efficiency and security of edge computing. The method uses federated learning to process the original data and only stores the calculation results, thus reducing the cost of storage resources and solving the problem of limited energy of edge devices. We also introduce the reputation and parameter quality measures to store the reputation and parameter quality of workers in the reputation blockchain to ensure that they cannot be tampered with, so that task publishers can trust them, and then select workers to train the model, thus solving the problem that malicious nodes affect the overall model. At the same time, we also consider the number of workers. The task publisher chooses a certain number of reputable and qualified workers according to the quality measure, so as to save more resources and alleviate the problem of resource shortage of MIoT nodes. We also designed a parameter quality proof consensus mechanism [proof of quality (PoQ)], which adds workers with better parameter quality to the blockchain, improves the effect of the final model and training efficiency, and solves the problem of variable data quality of edge nodes in the ocean Internet of things. The overall setting of this article is to select workers who meet the conditions of task publishers. What we want is workers who make the task better, so workers who have bad effects on the task will be excluded. In addition, we also build a marine environment model for the marine environment and simulate the marine parameters for specific experiments.

The main contributions of this article are as follows.

- 1) We propose an edge computing framework that combines blockchain and federated learning, mainly for marine systems in the MIoT.
- 2) To measure the reputation and quality of federated learning workers, we use the triple subjective logic model to calculate the node reputation value from interaction effect, interaction time, and geographical location, so as to improve the efficiency and security of edge computing. At the same time, it also limits the number of nodes to avoid resource waste.
- 3) We design a consensus mechanism for PoQ. Task publishers use PoQ mechanism when adding blocks, and it is more suitable for task publishers to select workers with better parameters to add to the blockchain according to quality.
- 4) We build an environmental model for the marine environment, which is more suitable to solve the data security problems in the marine environment.

The rest of this article is organized as follows. Section II introduces the related work. The system model is described in Section III. We explain the reputation and quality measurement calculations in Section IV. Section V describes the PoQ mechanism in detail. Section VI explains the experimental results. Section VII is the conclusion of this article.

II. RELATED WORK

In recent years, blockchain technology has ushered in the development opportunity, and the state has also issued relevant policies and calls to increase the research efforts on blockchain technology. One of its famous achievements, Bitcoin [19], is popular all over the world and its price is getting higher and higher. Blockchain is more and more in line with the current era of big data by virtue of its decentralized characteristics. The structure of each block in the blockchain is roughly as follows: 1) block version number (i.e., validation rules); 2) the hash of the parent block; 3) time stamp; 4) random number increasing from 0; 5) number of transactions; and 6) Merkle-Root [20] (the hash value of MerkleRoot which concatenates the hash values of all transactions in the block) [21]. As a result, each block in the blockchain (except the first) points to its immediate predecessor (called the parent block) through a backreference, which guarantees its tamper-proof nature. At the same time, blockchain also ensures the reliability of data on the chain through the consensus mechanism among multiple untrusted distributed participants, intelligent contracts, etc. So, more and more scholars begin to pay attention to the application of blockchain in various research fields, in which the use of its distributed and security characteristics to solve problems in the field of edge computing accounts for a large part.

Some studies integrate blockchain into edge computing architecture. Ning *et al.* [22] constructed a crowd perception framework supporting blockchain for its, and formulated a multiobjective optimization problem. It used DRL algorithm and multiplier distributed alternating direction method (DIADEM) algorithm to improve the security and efficiency of the system. Shen *et al.* [23] proposed a blockchain-based privacy-protected medical image retrieval system. Using blockchain technology, they also proposed a layered architecture and threat model of the system. Wan *et al.* proposed a security and privacy architecture for smart factories, introduced security and privacy models, decomposed and reorganized the original industrial Internet of things architecture, and formed a new multicenter partially distributed architecture [11]. Zhang *et al.* [24] proposed a framework combining edge intelligence and blockchain authorization in literature, which can be used for flexible and safe resource allocation and management in the industrial Internet of things, and realized the merger and scheduling of distributed and heterogeneous edge resources. They also designed a credit differentiated marginal transaction approval mechanism. The mechanism uses a credit network to ensure the reliability of anonymous notification and encourages edge nodes to join the approval process. Each approval node has a credit account that reflects the credit history of the node during the transaction approval process. Huang *et al.* [25] proposed a credit-based consensus mechanism B-IoT, a credit-based work proof mechanism for Internet of things devices. In order to protect the confidentiality of sensitive Internet of things data, they also designed a data rights management method to regulate the access of sensor data and improve the security and transaction efficiency of the system. However, there are still some problems in the research of edge architecture. First, edge devices have limited power and are

not suited to run the consensus mechanisms in the blockchain that consumes a lot of computing power. Second, there is no consideration for solving the inefficiency of blockchain. This is because blockchain requires a large amount of additional computing resources to achieve consistent records among distributed participants, and how to improve the efficiency of blockchain transaction authentication is the key to improve the computational efficiency of blockchain. At the same time, how to better integrate blockchain and intelligent edges in distributed scenarios to improve the overall security and intelligence of the system is a problem that needs to be considered in future research.

The birth of federated learning technology provides a solution to the problems of this architecture. Different from traditional centralized learning, federated learning only delegates the model parameter results obtained by training to the client and sends the results of training tasks to the server, so as to maintain the locality of data to users and ensure the privacy of user data [26]. Google first introduced federated learning technology and used it to design a virtual keyboard application for smartphones called Gboard [27]. By combining blockchain with federated learning, some studies have endowed the system with artificial intelligence optimization decision-making ability on the basis of retaining the security and credibility of blockchain. Meanwhile, federated learning can reduce the cost of storage resources by processing the original data and storing only the calculation results. In addition, the optimization of the authentication calculation and transmission communication of blockchain transactions based on federated learning can improve the operation efficiency of blockchain [28]. The specific application of blockchain combined with federated learning technology in edge computing is as follows. First, federated learning utilizes the distributed personalized dataset of the edge nodes of the Internet of things (such as mobile devices) to perform local training, while providing privacy protection for the edge nodes. The edge server updates the global model by using the local training data parameters collected from the edge nodes. Then, blockchain is used to manage the interactive data in the whole process without denying or tampering, so as to ensure the security of the interactive process [19]. Kang *et al.* [29] proposed a federated learning scheme based on reputation as a reliability measure to select reliable workers, using the interaction history of multiple weighted task publishers based on the subjective logic model and the opinions of recommendation reputation, and designed a kind of efficient reputation calculation scheme. This is deployed on the edge of the node reputation using blockchain management. Lu *et al.* [16] proposed a secure data sharing architecture with blockchain authorization, which stored and shared federated learning model and parameter transfer process through blockchain to ensure the security of the shared process. Literature [30] is based on the federated learning scenario of blockchain, and considers two types of weight to select the customer subset to update the global model. This scheme has high stability and can improve the convergence speed. Lu *et al.* [31] proposed a new federated learning mechanism for privacy protection and designed a two-stage mitigation scheme consisting of intelligent data

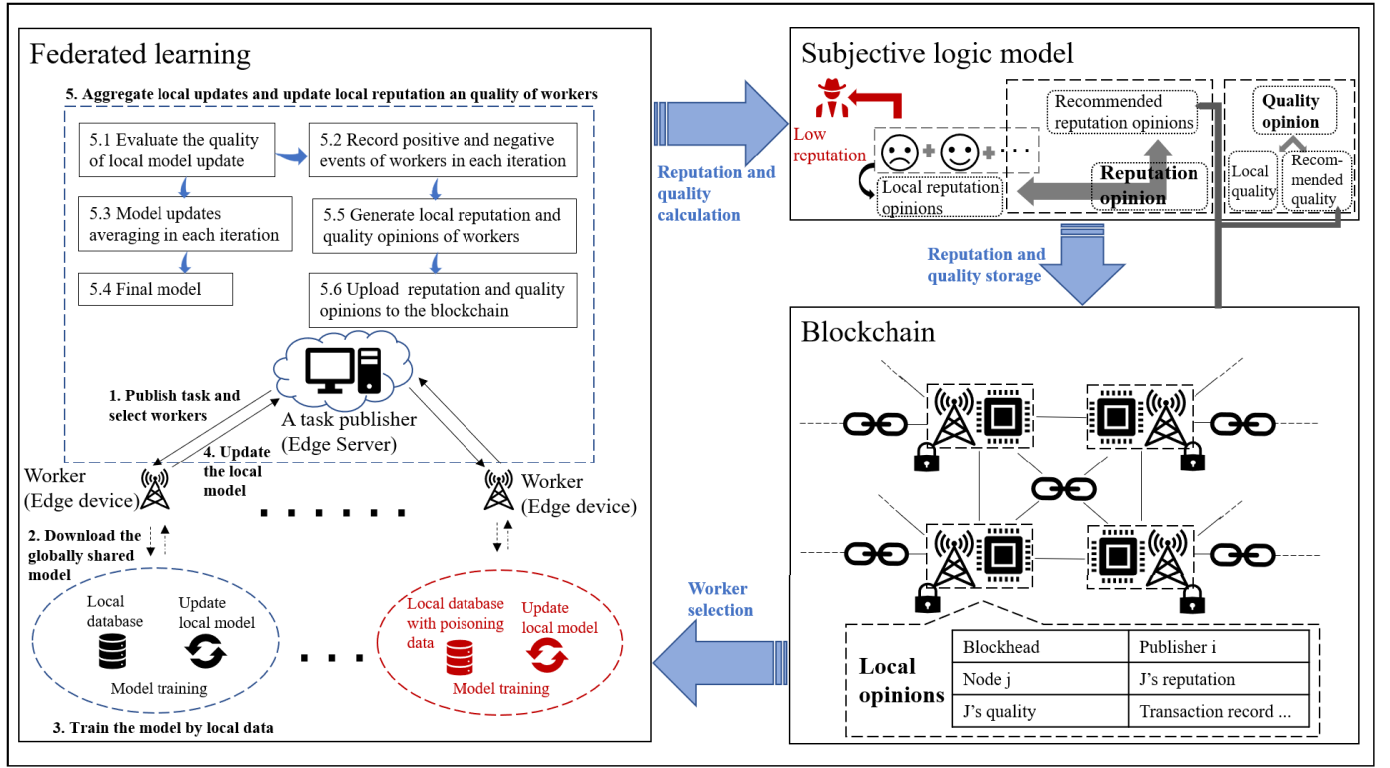


Fig. 1. Overview of our system model.

transformation and collaborative data leak detection, which has good accuracy, efficiency and high security.

Most of the above work is specifically focused on optimizing the performance of learning algorithms, such as optimizing the learning time or security of federated learning. However, most work assumes that all mobile devices are normal nodes, without considering the data quality of the nodes. Kang *et al.* [29] proposed a measurement scheme for node selection, reputation measurement, but they did not consider an appropriate blockchain consensus mechanism to suit the characteristics of edge nodes. At the same time, existing researches do not analyze specific scenarios, and the credibility measurement does not consider the distance factor. Therefore, in this article, we propose a more comprehensive measurement scheme, which comprehensively considers the interaction effect, time and distance factors, and propose the quality proof consensus mechanism (PoQ) to select higher quality workers and achieve higher accuracy. At the same time, we designed the marine environment model and tested the scheme against the marine environment. The results show that the scheme is more sensitive to the distance factor and can achieve higher accuracy faster.

III. SYSTEM OVERVIEW

This section describes the models used in the proposed approach, including the system model and environment model. Some of the main notations used in this article are shown in Table I.

A. System Model

In this article, we consider a distributed MIoT scenario involving federated learning data interaction between multiple

TABLE I
MAIN SYMBOLS

Notation	Definition
T_j	The set of opinion values for all workers in the opinion blockchain.
W	The set of workers requesting to participate in the federated learning task, which part of the nodes apply to join will get this part of the nodes as a set.
T_{min}	The threshold of opinion to choose workers on the federated learning task.
T_n	The direct values of workers' opinions in the federated learning.
$x_i(t), y_i(t), z_i(t)$	The current position of node i at time t .
$d_{ij}(t+1)$	The distance between node i and task publisher j at time $t+1$.
$SNR(d_{ij}, f)$	The average SNR at the receiver in MIoT.
P_e	The probability of sign error in the transmission process.
$b_{i \rightarrow j}, d_{i \rightarrow j}, u_{i \rightarrow j}$	Each reputation opinion is expressed as an opinion vector, which is trust $b_{i \rightarrow j}$, distrust $d_{i \rightarrow j}$ and uncertainty $u_{i \rightarrow j}$.
$T_{i \rightarrow j}$	A reputation value from the opinion vector to represent the worker's reputation.
ϑ_y	A freshness fading function to explain the freshness of interactive events.
$Sim(i, x)$	The similarity factor between task publisher i and task publisher x .
$q_{i \rightarrow j}$	The quality opinion vector.
B	The block to add to the opinion blockchain.
T	The time threshold.
Q	The quality threshold.

edge nodes and edge servers. Each edge node has its own data and serves the edge server. In this environment, due to the variability of the ocean, it is difficult for the edge nodes to control the quality of training parameters, and the nodes are generally mobile. Therefore, when the edge server releases the

federated learning task, it is necessary to select reliable and high-quality edge nodes for training each time to achieve better training effect. In view of this scenario, we design a method of secure data sharing for the MIIOT based on federated learning and blockchain technology.

The data flow in this article is from edge device to edge server to cloud server. The edge device must first transfer the model update from local data training to the edge server because the edge server is located at the edge of the network, which can greatly reduce the delay. In the MIIOT environment in this article, the edge servers are generally located in the shore base station, an edge server services a sea area. In the system model of this article, the edge servers are set as the task publisher of federated learning, and the edge devices are set as the workers of federated learning. The system model of this scheme is shown in Fig. 1, which integrates the federated learning system of blockchain, including the federated learning layer and the blockchain layer. The federated learning layer consists of an edge server with edge computing capability and some mobile edge nodes. In the MIIOT network, multiple task publishers with federated learning tasks (i.e., edge servers) perform model training based on data parameters from edge nodes instead of collecting data to protect the privacy of edge nodes. Each task publisher publishes the global model, and each worker (that is, the edge node) iteratively trains a shared global model, generating local model updates based on its local data. All workers then upload local model updates to the task publisher to update the global model. The training process is iterated until the accuracy of the global model reaches a predefined value. This predefined value is the effect set by the task publisher when the task is published, and it can be a certain accuracy or a certain number of iterations. The task publisher evaluates the local model update quality based on the behavior of model updates and training, and generates reputation and parameter quality opinions for interaction workers. More details on the evaluation and calculation of opinions will be given in Section IV. These opinions and interaction histories are stored in the opinion blockchain, which is tamper-proof, distributed, and shared among task publishers. In the blockchain layer, the opinion blockchain stores the worker's reputation and quality opinions, and the task publisher adds blocks using a parametric PoQ consensus algorithm. More details on the parameter quality proof mechanism will be given in Section V. Interaction history is also stored in the blockchain for verification by participants. Due to the decentralized and tamper-proof nature of the blockchain, opinions and histories in the block are enduring and transparent evidence. Subsequent task issuers will select workers based on this evidence.

As shown in Fig. 1, the general process of the scheme is as follows. First, the federated learning task publisher (the edge server) calculates the quality and reputation opinions of the edge nodes that meet the data conditions and selects the nodes. Then, the task publisher publishes the initial model to the selected nodes, which use the local dataset for local training. The task publisher then uses the local training data parameters collected from the node to update the global model and update reputation and quality opinions. Finally, blockchain

Algorithm 1 Framework of Federated Learning for Our System

Input:

- The set of opinion values for all workers in the opinion blockchain, T_j ;
- The set of workers requesting to participate in the federated learning task, W ;
- The threshold of opinion to choose workers on the federated learning task, T_{min} ;
- The number of iterations of the federated learning tasks, N ;

Output:

- The final accuracy of the federated learning tasks, A ;
- The direct values of workers' opinions in the federated learning, T_n ;

- 1: The federated learning task publisher publishes the task and gets W ;
 - 2: The task publisher selects a set of worker candidates W_j from W based on certain data requirements, as data types;
 - 3: The task publisher as the node of the blockchain, get the set of direct opinions value T_w of the workers in W_j from T_j ;
 - 4: Calculate the set of workers' combined opinions T_w^{final} according to T_w ;
 - 5: Select a set of workers W_{final}^{final} whose combined opinion is greater than T_{min} from T_w^{final} ;
 - 6: **for** $i = 0$; $i < N$; $i++$ **do**
 - 7: The task publisher publishes the global model to the workers in W_{final} ;
 - 8: The workers in W_{final} train the model using local datasets;
 - 9: The workers in W_{final} upload new model parameters;
 - 10: The task publisher update the global model;
 - 11: **end for**
 - 12: Calculate the workers' direct opinion value in W_{final} for this task;
 - 13: **return** A and T_n ;
-

technology is used to realize the effective management of nodes' opinions without denying or tampering. The detailed steps of the program are as follows.

Step 1 (Conduct a federated learning task): Algorithm 1 shows the federated learning module. Among them, federated learning tasks can use different optimization algorithms to train different models, such as Stochastic Gradient Descent (SGD) algorithm [32].

Step 2 (Update the blockchain): It mainly includes the following two steps.

- 1) *Build the opinion blockchain:* Because edge servers are easily connected by mobile devices and federated learning task publishers, there are enough computing and storage resources to work as preselected miners for the blockchain. Blockchain systems can be built with reliable PoQ algorithms as consensus algorithms

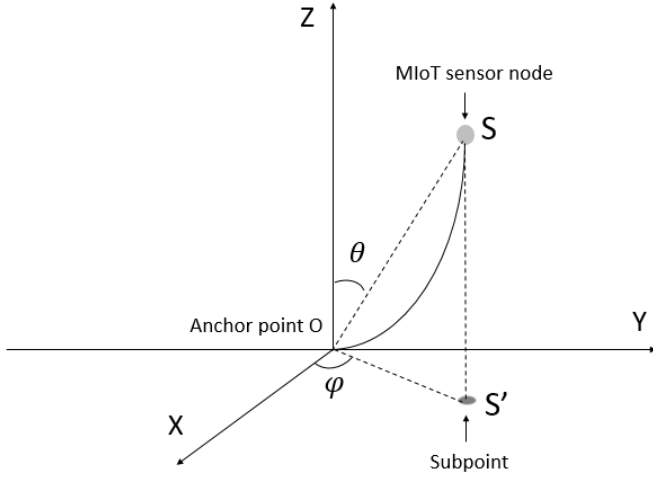


Fig. 2. Overview of marine environment model.

to guarantee the quality of workers recorded in the blockchain.

- 2) *Update comments to the blockchain:* Upon completion of the federated learning task, the task publisher updates its direct reputation and parameter quality opinions based on the interaction history with the worker. These digitally signed workers' comments are recorded as transactions and uploaded to the opinion blockchain. The miner puts the opinion record into the data block, performs block validation and after successful execution of the consensus algorithm (PoQ), adds the block to the blockchain. Finally, all task publishers can select reliable workers with high reputation and high parameter quality for their joint learning tasks with the help of the opinion blockchain.

B. Environment Model

Aiming at the MloT environment and considering the characteristics of the marine environment, this article quantifies the impacts of the environmental model into two categories: the mobility of ocean current flow and instability of acoustic communication. We build the MloT environment model by imitating the actual parameters of the marine environment.

1) *Mobility of Ocean Current Flow:* Under the influence of ocean currents, tides and other factors, sensor nodes show dynamic displacement in their relative positions. As shown in Fig. 2, consider anchoring the ocean sensor node i to a position O on the seabed, and the positions of the node and its projection on the lower surface are respectively represented as S and S' , and the included angle between S and the Z axis is θ , and the included angle between S' and the X axis is φ . Assuming that the task publisher j is taken as the origin, the current position of node i at time t is $(x_i(t), y_i(t), z_i(t))$, then the position of node i in the next time slot can be expressed as

$$\begin{cases} x_i(t+1) = x_i(t) + v_i(t) \sin d\theta_i(t) \cos d\varphi_i(t) \\ y_i(t+1) = y_i(t) + v_i(t) \sin d\theta_i(t) \sin d\varphi_i(t) \\ z_i(t+1) = z_i(t) + v_i(t) \cos d\theta_i(t) \end{cases} \quad (1)$$

where $v_i(t)$ is the velocity of sensor node i in time slot t , and it obeys the normal distribution $N(\mu_1, \sigma_1^2)$. $d\theta_i(t)$ and $d\varphi_i(t)$ are the motion directions of node i in time slot t , which obey the uniform distribution $U(0, \pi)$ and $U(0, 2\pi)$, respectively.

Then, the distance between node i and task publisher j is

$$d_{ij}(t+1) = \sqrt{x_i(t+1)^2 + y_i(t+1)^2 + z_i(t+1)^2}. \quad (2)$$

2) *Instability of Acoustic Communication:* The instability of underwater acoustic communication is mainly caused by channel fading and environmental noise [33]. Underwater acoustic channels show both range and frequency selective fading, which is caused by propagation loss and signal-heat transfer related to both transmission distance and transmission frequency. Based on the signal frequency f (in kHz), the attenuation occurring in the underwater acoustic channel can be estimated by the following relation:

$$A(d_{ij}, f) = A_0 d_{ij}^k a(f)^{d_{ij}} \quad (3)$$

where d_{ij} is the distance between transmitter i and receiver j , A_0 is the unit normalized constant, k is the expansion factor, and $a(f)$ (the unit is dB/km) is the absorption coefficient, which can be estimated by the following formula:

$$10 \log a(f) = 0.11 \frac{f^2}{1 + f^2} + 44 \frac{f^2}{4100 + f^2} + 2.75 \times 10^{-4} f^2 + 0.003. \quad (4)$$

Another factor affecting the quality of underwater acoustic channels is the existence of multiple noise sources in the underwater environment, usually including: turbulent noise $N_t(f)$, ship noise caused by surface wind $N_s(f)$, wave motion noise $N_w(f)$ and thermal noise $N_{th}(f)$. These noises are generally described by Gaussian statistics and are estimated by the following formula:

$$10 \log N_t(f) = 17 - 30 \log f \quad (5)$$

$$10 \log N_s(f) = 40 + 20(s - 0.5) + 26 \log f - 60 \log(f + 0.03) \quad (6)$$

$$10 \log N_w(f) = 50 + 7.5\sqrt{w} + 20 \log f - 40 \log(f + 0.4) \quad (7)$$

$$10 \log N_{th}(f) = -15 + 20 \log f \quad (8)$$

where s is a factor of the activity frequency of the surface container, and the value range of s is $[0, 1]$, and w is the wind speed (in m/s).

The effective noise level at frequency f is the sum of the contributions of the above factors is

$$N(f) = N_t(f) + N_s(f) + N_w(f) + N_{th}(f). \quad (9)$$

Then, for medium signal bandwidth B and transmitting power P , the average SNR at the receiver is calculated as

$$\text{SNR}(d_{ij}, f) = \frac{P/A(d_{ij}, f)}{N(f)B}. \quad (10)$$

Then, the probability of sign error in the transmission process can be approximated as

$$P_e = \frac{1}{4\text{SNR}}. \quad (11)$$

Finally, the bit error rate is used as the impact index of the environmental model.

IV. CALCULATE OPINION USING MULTIWEIGHT SUBJECTIVE LOGIC

Since the scheme is characterized by openness and immutability, an effective management mechanism is needed to effectively assess and manage the trust between nodes, so as to eliminate the possible existence of malicious nodes and other problems. In this section, we apply a subjective logic model to evaluate the opinion value of worker candidates. Subjective logic model is a widely used framework of probability theory, which is generally used to evaluate the level of credibility or reliability of different entities. The subjective logic model uses the term “opinion” to express subjective beliefs through positive and negative representations and uncertain statements [34]. In this article, in order to obtain more accurate workers’ opinions, we adopt multiple subjective logic model. Multiweight subjective logic is an extension of subjective logic, which considers different attributes of interaction events, such as interaction effect, time, location, etc. And each task publisher combines its direct and indirect opinions to generate a comprehensive opinion value for workers, resulting in a more comprehensive and reliable opinion.

In this article, two indexes are used to evaluate the worker candidates, one is the reputation opinion, the other is the parameter quality opinion. Reputation opinion evaluates the credibility of workers by integrating interaction effect, time, and location factors, and focuses on evaluating the overall reliability of workers. Based on reputation opinion, task publishers can set a certain threshold to exclude malicious workers. The parameter quality opinion evaluates the data quality of workers and focuses on the evaluation of the quality of workers by integrating the influencing factors of the environmental model. The task publisher sets a threshold based on the parameter quality opinion, which can exclude low-quality workers.

A. Calculate Reputation Opinion

Direct reputation opinion: Because reputation opinion is affected by many factors, the traditional subjective logic (TSL) is sometimes not enough for comprehensive consideration, and has gradually developed to multiple subjective logic. Therefore, this article adopts multiple subjective logic models to calculate direct reputation opinions and evaluates reputation opinions from three aspects: interaction effect, interaction location, and interaction time. This not only considers the direct effect, but also considers the influence of distance and time freshness, evaluates the workers objectively to the greatest extent, and ensures the reliability and adaptability of reputation opinions.

1) *Interaction Effect:* The task publisher verifies whether the local model update uploaded by the worker is reliable through specific verification means. If the local model update from the worker j is considered to be reliable, the task publisher i considers the learning of the worker j as a positive interaction event, and vice versa. Thus, the task publisher

records the number of positive and negative interaction events for all workers during the federal learning task, recorded as α_j and β_j , and then generates direct reputational opinions for the workers. Each reputation opinion is expressed as an opinion vector, which is composed of trust $b_{i \rightarrow j}$, distrust $d_{i \rightarrow j}$ and uncertainty $u_{i \rightarrow j}$. The uncertainty refers to the quality of the communication link between worker j and task publisher i , namely, the probability of failure of packet transmission. Thus, the direct impact of the interaction between the worker and the task publisher can be quantified as the percentage of positive (negative) interaction events, as follows:

$$\begin{cases} b_{i \rightarrow j} = (1 - u_{i \rightarrow j}) \frac{\alpha_j}{\alpha_j + \beta_j} \\ d_{i \rightarrow j} = (1 - u_{i \rightarrow j}) \frac{\beta_j}{\alpha_j + \beta_j} \\ u_{i \rightarrow j} = 1 - q_{i \rightarrow j} \end{cases} \quad (12)$$

where $q_{i \rightarrow j}$ represents the probability of successful packet transmission.

Generate a reputation value from the opinion vector to represent the worker’s reputation, as

$$T_{i \rightarrow j} = b_{i \rightarrow j} + \gamma u_{i \rightarrow j} \quad (13)$$

where γ is a given constant, representing the influence level of reputation uncertainty.

The positive interaction will increase the credibility opinion of the workers, while the negative interaction will decrease the credibility opinion of the workers. In addition, since negative interaction events have worse effects, our purpose is to suppress the occurrence of negative interaction events, so negative interaction events have a higher weight on the calculation of reputation opinions compared with positive interaction events. The weights of positive and negative interactions are respectively expressed as k and η ($k \leq \eta$ and $k + \eta = 1$). Therefore, the reputation opinion vector expression can be rewritten as

$$\begin{cases} b_{i \rightarrow j} = (1 - u_{i \rightarrow j}) \frac{k\alpha_j}{k\alpha_j + \eta\beta_j} \\ d_{i \rightarrow j} = (1 - u_{i \rightarrow j}) \frac{\eta\beta_j}{k\alpha_j + \eta\beta_j} \\ u_{i \rightarrow j} = 1 - q_{i \rightarrow j} \end{cases} \quad (14)$$

2) *Interaction Location:* The credibility of workers tends to change with location. The closer the worker is to the task publisher, the higher the credibility of the worker is, because it will be less affected by communication quality and other factors. At the same time, workers in closer proximity can also reduce more energy consumption associated with communication and save more resources. Here, the environmental model designed according to Section III mainly considers the mobility of ocean current flow and the instability of acoustic communication in the MIIOT and then takes them into account in reputation calculation. According to the environmental model, the variable P_e obtained from its final quantization is

added to the reputation opinion vector, which is

$$\begin{cases} b_{i \rightarrow j} = (1 - P_e)b_{i \rightarrow j} \\ d_{i \rightarrow j} = (1 - P_e)d_{i \rightarrow j} \\ u_{i \rightarrow j} = (1 - P_e)u_{i \rightarrow j}. \end{cases} \quad (15)$$

3) *Interactive Freshness*: Workers' credibility also changes over time, and workers are not always trustworthy and reliable in the interaction between task publishers and workers. More recent interactions have a greater impact on credibility than past interactions, and therefore carry more weight. In order to consider the influence of time on reputation, this article defines a freshness fading function to explain the freshness of interactive events: $\vartheta_y = z^{Y-y}$, where $z \in (0, 1)$ is the fading parameter given about the freshness of the event, and $y \in [1, Y]$ is the time slot that determines the fading degree of the freshness of the event. Therefore, the reputation opinion of task publisher i on worker j in a time window can be expressed as

$$\begin{cases} b_{i \rightarrow j} = \frac{\sum_{y=1}^Y \vartheta_y b_{i \rightarrow j}}{\sum_{y=1}^Y \vartheta_y} \\ d_{i \rightarrow j} = \frac{\sum_{y=1}^Y \vartheta_y d_{i \rightarrow j}}{\sum_{y=1}^Y \vartheta_y} \\ u_{i \rightarrow j} = \frac{\sum_{y=1}^Y \vartheta_y u_{i \rightarrow j}}{\sum_{y=1}^Y \vartheta_y}. \end{cases} \quad (16)$$

The final direct credit opinion is expressed as

$$T_{i \rightarrow j} = \frac{\sum_{y=1}^Y \vartheta_y T_{i \rightarrow j}}{\sum_{y=1}^Y \vartheta_y}. \quad (17)$$

Indirect reputation opinion: Indirect reputation opinion is the reputation opinion of other task publishers relative to the current task publisher, which is still expressed as the reputation vector. Indirect reputation opinions are considered in order to get suggestions from other task publishers, and evaluations from other task publishers are considered to make the comprehensive reputation opinions more authentic and reliable. First, consider the similarity of the workers when each task publisher publishes the task in order to evaluate the same workers. The similarity of workers' reputation opinions among task issuers can be measured by the similarity of the evaluation vector by modifying the cosine function. Consider the similarity factor between task publisher i and task publisher x , which is

$$\text{Sim}(i, x) = \frac{\sum_{j \in C} (D_{i \rightarrow j} - \bar{D}_i)(D_{x \rightarrow j} - \bar{D}_x)}{\sqrt{\sum_{j \in \Pi} (D_{i \rightarrow j} - \bar{D}_i)^2} \sqrt{\sum_{j \in X} (D_{x \rightarrow j} - \bar{D}_x)^2}} \quad (18)$$

where Π and X are set of workers with task publishers i and x , respectively. $C = \Pi \cap X$ is the set of workers that interact with task publisher i and task publisher x . $\bar{D}_i$ and $\bar{D}_x$ are the mean values of their direct reputation opinions of interacting workers in C , respectively. $D_{i \rightarrow j}$ and $D_{x \rightarrow j}$ are the reputational opinions of worker j from task publisher i and x , respectively.

Therefore, the overall weight of recommender x 's indirect reputation opinion is expressed as $\varpi_{i \rightarrow x} = \sigma_{i \rightarrow x} \times \text{Sim}(i, x)$, where $0 \leq \sigma_{i \rightarrow x} \leq 1$ is a predefined parameter representing the weight of the recommendation x of task publisher i during the calculation period.

All indirect reputation opinions of the recommender on worker J can be integrated into an overall recommendation reputation opinion in the form of vector, that is

$$\begin{cases} b_{x \rightarrow j}^{\text{rec}} = \frac{\sum_{x \in X} \varpi_{i \rightarrow x} b_{x \rightarrow j}}{\sum_{x \in X} \varpi_{i \rightarrow x}} \\ d_{x \rightarrow j}^{\text{rec}} = \frac{\sum_{x \in X} \varpi_{i \rightarrow x} d_{x \rightarrow j}}{\sum_{x \in X} \varpi_{i \rightarrow x}} \\ u_{x \rightarrow j}^{\text{rec}} = \frac{\sum_{x \in X} \varpi_{i \rightarrow x} u_{x \rightarrow j}}{\sum_{x \in X} \varpi_{i \rightarrow x}} \end{cases} \quad (19)$$

where X is a recommended set of interactions with worker j . The indirect reputational opinions of different recommenders are combined into an overall recommendation reputation according to the corresponding weight of task publisher i .

Comprehensive reputation opinion: Direct reputation opinion and indirect reputation opinion are combined to form a comprehensive reputation opinion as the final reputation opinion of worker candidate j . This takes into account not only recommended reputation opinions, but also direct reputation opinions to avoid cheating by other task publishers. The final reputational comment is expressed as

$$\begin{cases} b_{i \rightarrow j}^{\text{final}} = \frac{b_{i \rightarrow j} u_{x \rightarrow j}^{\text{rec}} + b_{x \rightarrow j}^{\text{rec}} u_{i \rightarrow j}}{u_{i \rightarrow j} + u_{x \rightarrow j}^{\text{rec}} - u_{x \rightarrow j}^{\text{rec}} u_{i \rightarrow j}} \\ d_{i \rightarrow j}^{\text{final}} = \frac{d_{i \rightarrow j} u_{x \rightarrow j}^{\text{rec}} + d_{x \rightarrow j}^{\text{rec}} u_{i \rightarrow j}}{u_{i \rightarrow j} + u_{x \rightarrow j}^{\text{rec}} - u_{x \rightarrow j}^{\text{rec}} u_{i \rightarrow j}} \\ u_{i \rightarrow j}^{\text{final}} = \frac{u_{x \rightarrow j}^{\text{rec}} u_{i \rightarrow j}}{u_{i \rightarrow j} + u_{x \rightarrow j}^{\text{rec}} - u_{x \rightarrow j}^{\text{rec}} u_{i \rightarrow j}}. \end{cases} \quad (20)$$

Thus, the final creditworthiness of task publisher i to worker candidate j is

$$T_{i \rightarrow j}^{\text{final}} = b_{i \rightarrow j}^{\text{final}} + \gamma u_{i \rightarrow j}^{\text{final}}. \quad (21)$$

B. Calculate Quality Opinion

Similar to the reputation opinion calculation, the parameter quality opinion calculation also adopts the tripartite subjective logic model, taking into account the influence of direct quality, location, and time.

Direct quality opinion: The direct quality opinions are represented by a quantified vector that takes into account the three aspects of the interaction of the current task publisher.

1) *Interaction Effect*: In the process of interaction, the task publisher records the updated model parameters uploaded by workers in each iteration, tests the current model with the test set, and obtains the direct quality $q_{i \rightarrow j}$ of worker j , namely the average value of the test accuracy of the model trained in each iteration.

2) *Interaction Location*: The quality of workers changes with the location, and the closer the worker is to the task publisher, the higher the quality of workers is. Calculated like reputation opinion, the quality opinion can be obtained as follows:

$$q_{i \rightarrow j} = (1 - P_e)q_{i \rightarrow j}. \quad (22)$$

3) *Interactive Freshness*: The quality of workers also changes over time, and workers are not always trustworthy and reliable in the interaction between task publishers and workers. Recent interactions have more weight on updated events than past events. Calculated like reputation opinion, the quality opinion can be obtained as follows:

$$q_{i \rightarrow j} = \frac{\sum_{y=1}^Y \vartheta_y q_{i \rightarrow j}}{\sum_{y=1}^Y \vartheta_y}. \quad (23)$$

4) *Indirect Quality Opinion*: Calculate like reputation opinion, the indirect quality opinion of the recommender on worker j can be obtained

$$q_{x \rightarrow j}^{\text{rec}} = \frac{\sum_{x \in X} \varpi_{i \rightarrow x} q_{x \rightarrow j}}{\sum_{x \in X} \varpi_{i \rightarrow x}}. \quad (24)$$

Comprehensive quality opinion: The final quality opinion can be obtained by integrating direct quality opinion and indirect quality opinion like reputation opinion calculation

$$q_{i \rightarrow j}^{\text{final}} = (1 - \delta)q_{i \rightarrow j} + \delta q_{i \rightarrow j}^{\text{rec}} \quad (25)$$

where δ is the weight of recommended quality opinions.

V. PROCESS OF POQ

PoQ consensus mechanism is a new consensus algorithm of blockchain, which is mainly suitable for distributed nodes with less energy. On the one hand, PoQ consensus algorithm is adopted in this article to be more suitable for the MIIOT environment. Compared with PoW consensus algorithm, it can reduce the consumption of meaningless computing resources. Compared with the practical byzantine fault tolerance (PBFT) algorithm, PoQ algorithm does not have too many restrictions on the number of nodes. On the other hand, its purpose is to make the workers stored in the blockchain have higher reputation value. Meanwhile, in conjunction with the parameter quality opinions, workers selected by the task publisher will have higher quality data. At the same time, PoQ measures quality by calculating and verifying the quality of the new block by the committee node using the same record, which is an objective calculation. The process of PoQ algorithm is performed on the basis of cooperative training work by preselected committee nodes (initially, we randomly select the committee nodes). This reduces some communication overhead by sending consensus messages only to the committee nodes rather than to all nodes. In order to balance overhead and security, PoQ algorithm uses proof of training workload on data. The PoQ algorithm selects committee leader based on the quality of the training model update, which is represented by the accuracy of the training model on the test set.

The process of PoQ algorithm is shown in Fig. 3. In each round of training, some nodes are randomly selected as

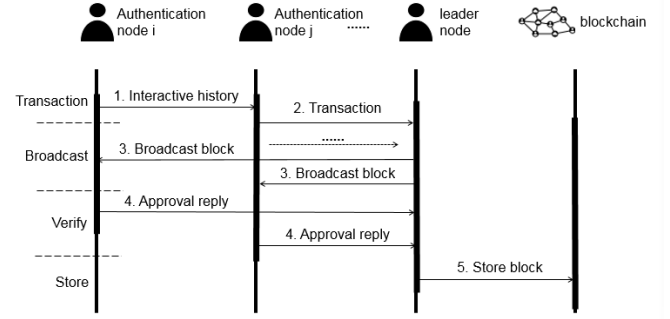


Fig. 3. Consensus process of PoQ algorithm.

certification nodes (i.e., committee nodes), and a leader node is selected from the certification nodes according to the historical training performance. Similar to a traditional blockchain, each training model update and the value of a worker's opinion for the task are stored as a transaction and packaged into a block. The authentication nodes validate the current block based on the data on the blockchain. The validation process is the authentication nodes calculate the current block's quality opinion based on the data, and if the parameter quality opinion value is within a specified threshold range and higher than the latest parameter quality opinion value, the block is considered valid. At the same time, we also set a time threshold. We set the threshold value at present as a fixed value, which is set according to the frequency of federated learning tasks. If the worker's parameter quality opinion is not updated within a period of time above the time threshold (that is, there may be no transaction higher than the latest parameter quality opinion value), the validity of the block is directly verified, and the results are finally returned to the leader node. The lead node combines the results returned by the various committee nodes to determine whether the transaction block is valid or not. The detailed PoQ algorithm of consensus mechanism based on parameter quality is shown in Algorithm 2.

The PoQ consensus mechanism maximizes the exclusion of mass attacks, ensuring that the low-quality workers with high reputation scores when the majority of workers are of low quality will be excluded when the consensus algorithm running. It makes up for the lack of credibility judgments when most workers are of low quality.

VI. EXPERIMENTS

In order to verify the effectiveness of the method we designed for secure data sharing in the MIIOT based on federated learning and blockchain, we conduct experiments on the MNIST handwritten image dataset [35].

A. Experiment Setup

MNIST dataset including 60000 training samples and 10000 testing image samples, each sample is a $28 * 28$ gray image, and display the handwritten numbers within 0–9. And take it out it will be a row vector for a 1-D, these row vectors are the lines of image arrays (784 values in each row and each

Algorithm 2 PoQ Consensus Algorithm**Input:**

A block to add to the opinion blockchain, B ;
 The set of committee nodes in the opinion blockchain, C ;
 The set of nodes' historical interaction records in the opinion blockchain, R ;
 A time threshold, T ;
 The current time, T_b ;
 A quality threshold, Q ;

Output:

The result of this block addition, X ;

```

1: Select a leader node  $L$  from  $C$  based on  $R$  and some
   random factors;
2:  $L$  broadcasts  $B$  to the node in  $C$ ;
3: for each  $i \in C$  do
4:   Calculate the quality opinion value  $Q_i$  of  $B$  according to
      $R$ ;
5:   if  $Q_i \geq Q$  then
6:     Returns the authentication result as passed to  $L$ ;
7:   else
8:     Get the current worker's latest opinion update time  $T_i$ 
       from  $R$ ;
9:     if  $T_b - T_i \geq T$  then
10:      Returns the authentication result as passed to  $L$ ;
11:    else
12:      Returns the authentication result that does not pass
        to  $L$ ;
13:    end if
14:  end if
15: end for
16:  $L$  synthesizes all the authentication results of nodes in  $C$ 
   and decides whether to add  $B$  to the opinion blockchain;
17: return true or false;
```

row represents a picture) [35]. MNIST is a standard dataset that is widely used to test machine learning algorithms, and it is very suitable for IoT applications. Therefore, MNIST dataset is used in this article to test the performance of the proposed scheme.

In our experiment, the model we set up federated learning training is the multilayer perceptron (MLP) model. An MLP, also known as an artificial neural network (ANN), in addition to input and output layers, it also has multiple hidden layers. In the federated learning simulation experiment of this article, the parameters are set as follows. The learning rate is 0.01. The sizes of the cores used for convolution are 3, 4, and 5, and the number of cores for each type defaults to 9. The default number of filters for the conv network is 32. The default number of global epochs is 10 and the default number of local epochs is 10. The default number of local batch size is 10 and the default number of test batch size is 128. The default number of SGD momentum is 0.5.

In the whole federated learning task, we set up 100 workers, each federated learning task select ten workers to learn. In the environmental model, the positions of the 100 workers are evenly distributed, and the wind speed is set between 0

and 36 m/s. Among the 100 workers, we set two malicious workers, eight workers with low-quality data, and the rest are workers with good behavior. Malicious workers mislead model updates by changing their training set tags. The higher the percentage of changed tags, the stronger the attack intensity. There may be occasional instances of poor data quality among workers with poor data quality. The training set of well-behaved workers is randomly assigned, following a uniform distribution. At the same time, we built the opinion blockchain, and design the PoQ algorithm suitable for the characteristics of the MIoT as the consensus algorithm of the opinion blockchain. The blockchain for this article is a federated chain. It runs on the edge server (task publisher). At present, 6 edge servers are tested in the experiment. The blockchain runs on each task publisher. According to the characteristics of edge computing, the distributed blockchain is guaranteed.

In the opinion calculation section, the frequency of interaction between task publishers and workers is 10–20 federated learning tasks per week. The probability of unsuccessful transmission of packets is between 0% and 40%, and the initial reputation opinion is 0.5 for all workers, and the initial parameter quality opinion is 50%.

B. Experimental Results

In this article, a malicious worker in a federated learning task was simulated and tested respectively when the worker launched 0%, 20%, 40%, 60%, and 80% attack intensity, which was carried out by changing the label of the worker's training set. The attack intensity was the percentage of the change of the label. Fig. 4 shows the change of the loss function of federated learning task under these five conditions. It can be seen that the loss function of federated learning increases with the increase of the attack intensity of malicious workers. Under the attack intensity of 20%, the final loss function of federated learning is 0.092, while under the attack intensity of 80%, the final loss function of a federated learning task is 0.195. The final loss function of the federated learning task can reach 0.015. Therefore, it is very necessary to exclude malicious workers from the set of workers in the federal learning task.

Fig. 5 shows the relationship between the loss function of federated learning and the number of iterations in four cases. We simulate the following four situations: 1) none of the workers on the federated learning task behaved badly; 2) the federated learning task have a worker with low-quality data; 3) one of the workers on the federated learning assignment is a maliciously assaulted worker; and 4) among the workers on the federated learning assignment are two hostile workers. As can be seen from the figure, with the increase of the number of workers with bad behaviors, the federated learning loss function also increases, which seriously affects the accuracy of federated learning. In the case of no malicious workers, the loss function of federated learning is reduced to 0.015 after ten iterations. And because low-quality data workers are contingent, their behaviors may change over time, and the model update by one low-quality data worker can be minimized by

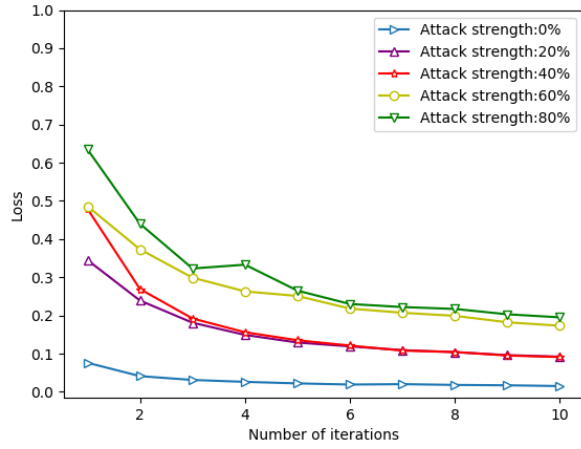


Fig. 4. Loss function of federated learning varies with the attack strength of the malicious worker.

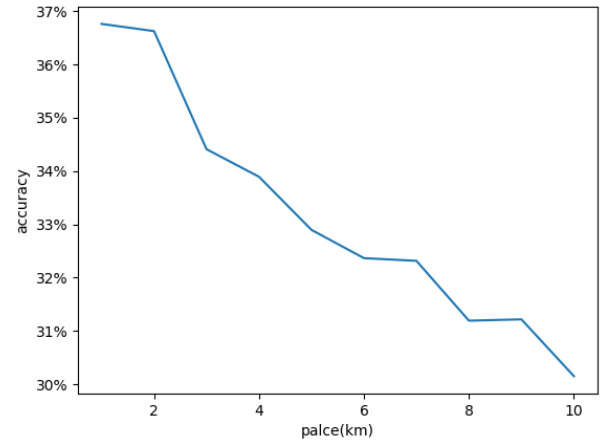


Fig. 6. Accuracy of local training for workers in different locations.

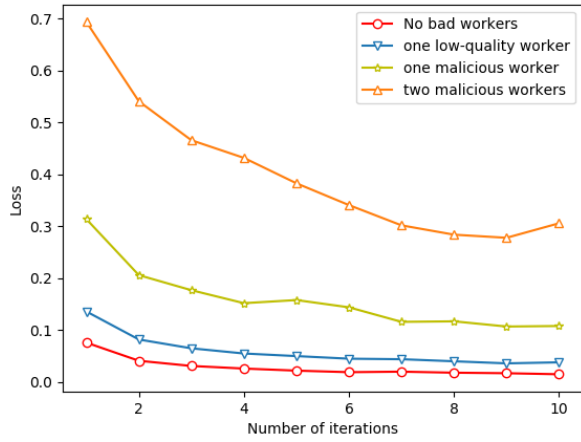


Fig. 5. Loss function of federated learning varies with whether or not there are bad workers among the workers.

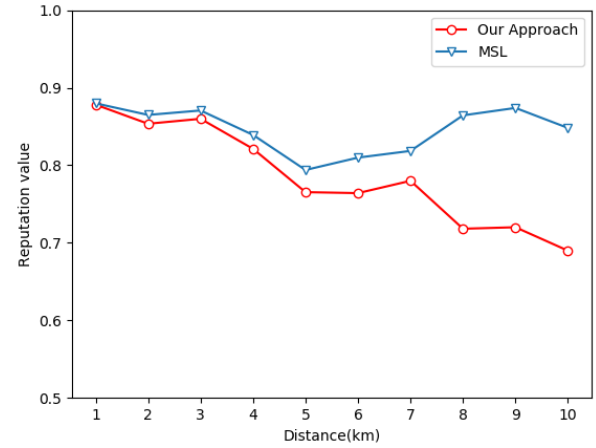


Fig. 7. Reputation opinions of workers under different schemes changed with the change of geographical location.

other well-behaved workers. Therefore, the loss function of federated learning with one low-quality data worker is only higher than the loss function of federated learning for well-behaved workers, and with a final value of 0.038. However, the malicious attack workers directly tamper with the labels of the dataset and directly mislead the model update, so it has a great impact on the loss function of the overall model of federated learning. The final loss function of one malicious worker's federated learning task is 0.108, and the final loss function of two malicious workers' federated learning task is 0.306. So the more malicious workers there are, the worse the impact.

To show the effect of location in federated learning tasks, we considered a task publisher and ten workers at different distances. And release a federated learning task to these workers, each worker uses the local data to train the model. Then we use test sets to test their local models and document their accuracy. Fig. 6 shows the relationship between the distance of the ten workers and the task publisher and accuracy. It can be seen that the farther workers have lower accuracy of the local model in the federated learning task. Therefore, it is necessary to take the location factor into account in the calculation of

workers' opinions. Therefore, the calculation scheme proposed in this article takes three factors of an interaction effect, time, and geographical location into consideration, making its value more comprehensive and reliable.

Then, we compare the reputation opinion calculation of the proposed scheme with the MSL scheme proposed by Kang *et al.* [29], which only considers the interaction effect and time effect. As shown in Fig. 7, we show the relationship between the reputation opinions of nodes at some distance from the task publisher and their geographical location under both calculation scenarios. It can be seen that, with the increase of the distance between the worker and the task publisher, the reputation opinion calculated by the MSL method has no significant change, and it is not sensitive to the distance factor. However, the reputation opinion calculated by our method obviously decreases with the increase of distance between the worker and the task publisher, which is sensitive to the distance factor. The effect of distance is well represented in our method.

Next, we conduct multiple federated learning tasks by using different programs to aggregate the reputation opinions of the same worker. We compare our proposed scheme with the no-reputation defenseless baseline scheme, the TSL

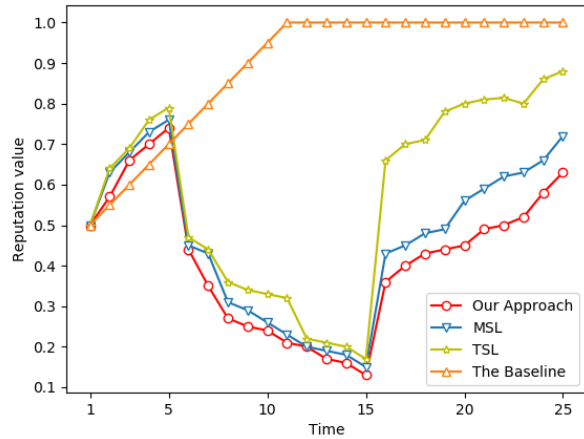


Fig. 8. Workers' reputation opinions under different schemes changed over time.

scheme [36], and the MSL scheme. In the TSL scheme, the TSL model is used to calculate the reputation opinion, that is, to judge directly from the effect. In the MSL scheme, it uses the multiple subjective logic model and consider interaction effect, frequency and schedule. We track a federated learning worker and keep it on the federated learning assignment for a period of time, and let it begin to show bad behavior on the fifth assignment and return to good behavior on the fifteenth assignment. Fig. 8 shows the reputation opinion values of this worker under different scenarios. It can be seen that when a worker has bad behavior, their reputation opinion starts to decline in MSL, TSL, and our scheme, but in the absence of reputation defense (the baseline scheme), its reputation opinion still increases linearly. And due to the consideration of interaction effects and time, the reputation of MSL solutions declines faster than TSL solutions in a short time. However, due to the consideration of our scheme of geographical location, the reputation opinion calculated by our scheme decreased faster than it is calculated by the MSL scheme in a short time. Moreover, when the work return to good behavior at the 15th federated learning session, the reputation of the TSL solution rose fastest because it only considers interactions and does not consider historical records. The reputation of MSL solutions has risen more slowly than TSL, and our scheme has risen more slowly than MSL solutions.

As shown in Fig. 9, we also compared the discrimination rates of bad workers using three different schemes by setting different quality thresholds, which are the scheme in this article, the scheme without reputation opinions and the scheme without PoQ mechanism. It can be seen that the higher the quality opinion threshold under all schemes, the more bad workers will be screened out. In the scheme of this article, when the quality opinion threshold reaches 40%, the discrimination rate of bad workers has reached 100%, that is, all the bad workers have been excluded. In the scheme without PoQ mechanism, the discrimination rate is 100% when the quality threshold is 50%, and in the scheme without reputation opinion, the discrimination rate is 100% when the quality threshold is 45%. It can be seen that under all quality

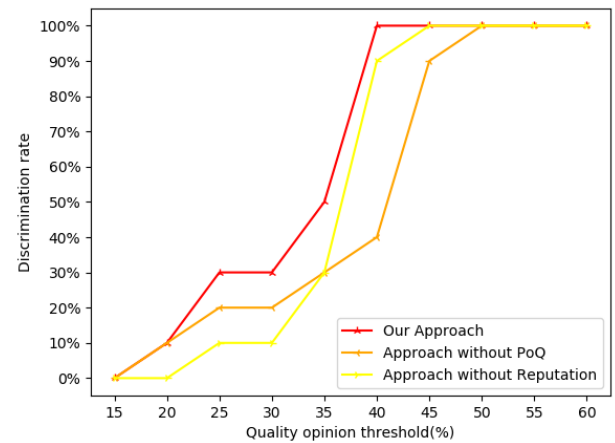


Fig. 9. Discrimination rate of bad workers under different schemes changes with the change of the quality opinion threshold chosen by workers.

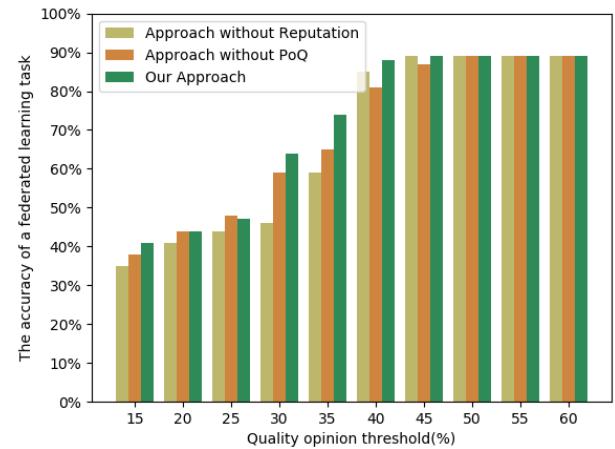


Fig. 10. Federated learning accuracy under different schemes changes with the change of the quality opinion threshold chosen by workers.

thresholds, the discrimination rate of the scheme proposed in this article is no lower than that of the other two schemes, which can better exclude undesirable workers. Therefore, the combination of PoQ mechanism and reputation-quality measures in this scenario enables task publishers to select more reliable and high-quality workers.

At the same time, as shown in Fig. 10, under three different schemes, with the increase of quality opinion threshold, the accuracy of federated learning under the three schemes also naturally increases. In the case of the same quality threshold, both the scheme without reputation opinion and the scheme without PoQ have lower accuracy than the scheme in this article. This is because the quality opinions generally have little difference for all nodes. Without the evaluation of reputation opinions, it is difficult to exclude workers with bad behaviors at a lower quality threshold, while it can be excluded at a higher quality threshold. In the non-PoQ scenario, all workers' quality comments were generally lower, so the corresponding federated learning was less accurate. As shown in the figure, among the three schemes, the scheme in this article has reached a high accuracy when the quality

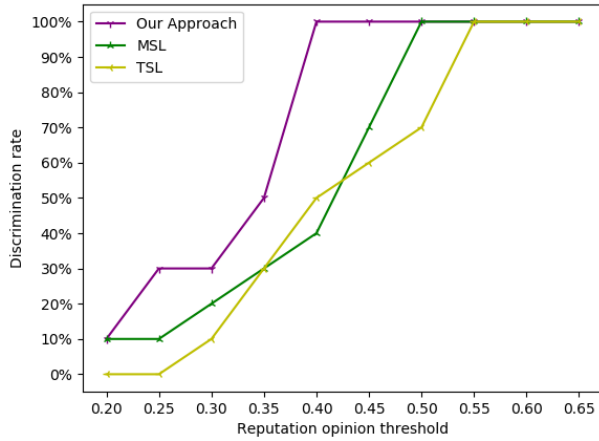


Fig. 11. Discrimination rate of bad workers under different schemes changes with the change of the reputation opinion threshold chosen by workers.

threshold reaches 40%, while the scheme without reputation opinion can only reach it when the quality threshold reaches 45%, and the scheme without PoQ can only reach it when the quality threshold reaches 50%. Therefore, the scheme in this article combines the dual evaluation of quality and reputation, and uses the PoQ mechanism in conjunction with the quality opinion to make the workers of the federated learning more reliable and high quality.

Then, as shown in Fig. 11, we set different credibility thresholds to compare the discrimination rates of bad workers under three different schemes, which are the scheme in this article, the TSL scheme and the MSL scheme. It can be seen that under either scenario, the higher the threshold of reputable opinions, the more bad workers will be screened out. Under the scheme in this article, when the credibility opinion threshold is 0.4, the discrimination rate of bad workers has reached 100%. The MSL scheme did not have a 100% discrimination rate until the credibility opinion threshold reached 0.5, while the TSL scheme did not have a 100% discrimination rate until the credibility opinion threshold reached 0.55. It can be seen that under all the credibility opinion thresholds, the discrimination rate of the scheme in this article is always greater than or equal to that of the TSL scheme and MSL scheme, and more bad workers will be excluded.

Fig. 12 shows the effect of the opinion thresholds selected for workers under different scenarios on the accuracy of the federated learning task. It can be seen that the accuracy of the proposed scheme is higher than that of the TSL scheme and MSL scheme under different thresholds. As shown in the figure, the scheme in this article will reach a high accuracy when the opinion threshold reaches 0.4, while the MSL scheme will reach a high accuracy when the threshold reaches 0.5, while the TSL scheme will reach a high accuracy when the threshold reaches 0.55. Therefore, through comparison, it can be seen that the scheme in this article can evaluate workers more accurately and comprehensively, so that the federated learning task can achieve better effect and accuracy.

Fig. 13 shows the changes of the accuracy of three different schemes (the scheme in this article, the MSL scheme, and the TSL scheme) under different attack ranges and using the same

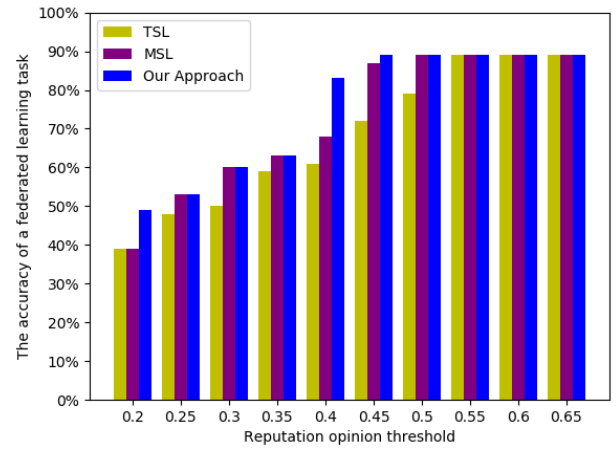


Fig. 12. Federated learning accuracy under different schemes changes with the change of the reputation opinion threshold chosen by workers.

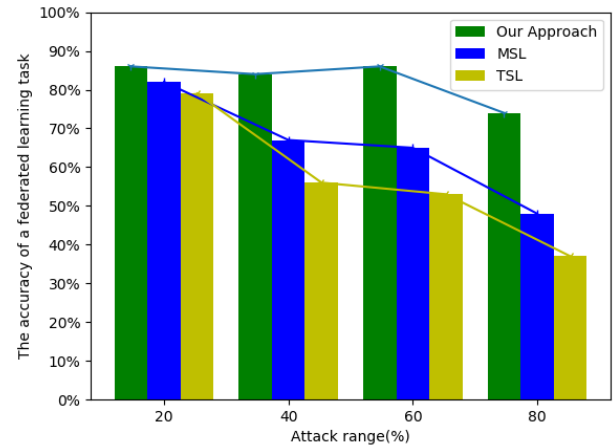


Fig. 13. Federated learning accuracy under different schemes changes with the change of the attack range of workers.

worker opinion threshold. We set up situations in which 20%, 40%, 60%, and 80% of the total workforce had bad behavior. It can be seen that the scheme in this article can achieve high accuracy under each attack range, while the accuracy of federated learning under MSL scheme and TSL scheme will decrease with the expansion of the attack range. In particular, when the attack range is 60%, the accuracy of federated learning under the proposed scheme can still reach 86%, while the accuracy under MSL and TSL schemes is significantly lower. This is because the scheme in this article considers the dual measurement of reputation and quality opinions, and the use of quality measurement makes up for the possibility of a wide range of attacks under the condition of a single reputation measurement, which cannot be compared and leads to high reputation evaluation. Therefore, the scheme in this article can deal with the situation of large-scale attack and can ensure that when most nodes are malicious nodes, they can be identified and eliminated.

VII. CONCLUSION

In this article, aiming at the MIoT environment, we propose a data security sharing method based on federated learning and blockchain technology to solve the problem of worker

selection in federated learning tasks, so as to ensure the reliability of federated learning. We use a variety of advanced technologies to construct our system, including mobile edge computing, blockchain, and federated learning. We design an opinion-based program to select reliable, high-quality workers. In order to be more suitable for the marine environment, we design an environmental model to simulate the marine environment. We also design a proof of parameters quality (PoQ) consensus mechanism for opinion blockchain. By adopting PoQ mechanism, better quality workers are added to the blockchain. So, the task publisher when the choice can be carried out in accordance with the quality. And PoQ mechanism also makes up for the deficiency of the single credit mechanism when the judge, that excludes the possibility of a large-scale attack. In addition, the PoQ mechanism ensures that low-quality workers are excluded from their operations when the majority of workers are of low quality, resulting in a situation in which they have a high reputation.

In the future, our goal is to conduct more experiments and tests. In addition, due to the high costs associated with large numbers of workers participating in federated learning, we will also consider the design of programs to optimize the number of workers in order to save resources costs. We will also consider how to dynamically change reputation thresholds to minimize the negative impact of malicious workers. We will also consider the time threshold by analyzing the

REFERENCES

- [1] T. Xia, M. M. Wang, and X. You, "Satellite machine-type communication for maritime Internet of Things: An interference perspective," *IEEE Access*, vol. 7, pp. 76404–76415, 2019.
- [2] Y. Huo, X. Dong, and S. Beatty, "Cellular communications in ocean waves for maritime Internet of Things," *IEEE Internet Things J.*, vol. 7, no. 10, pp. 9965–9979, Oct. 2020.
- [3] F. Samie, L. Bauer, and J. Henkel, "From cloud down to things: An overview of machine learning in Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4921–4934, Jun. 2019.
- [4] Z. Ning *et al.*, "Partial computation offloading and adaptive task scheduling for 5G-enabled vehicular networks," *IEEE Trans. Mobile Comput.*, early access, Sep. 18, 2020, doi: [10.1109/TMC.2020.3025116](https://doi.org/10.1109/TMC.2020.3025116).
- [5] X. Wang, Z. Ning, S. Guo, and L. Wang, "Imitation learning enabled task scheduling for online vehicular edge computing," *IEEE Trans. Mobile Comput.*, early access, Jul. 28, 2020, doi: [10.1109/TMC.2020.3012509](https://doi.org/10.1109/TMC.2020.3012509).
- [6] M. Alrowaily and Z. Lu, "Secure edge computing in IoT systems: Review and case studies," in *Proc. IEEE/ACM Symp. Edge Comput. (SEC)*, Oct. 2018, pp. 440–444.
- [7] H. Li, G. Shou, Y. Hu, and Z. Guo, "Mobile edge computing: Progress and challenges," in *Proc. 4th IEEE Int. Conf. Mobile Cloud Comput., Services, Eng. (MobileCloud)*, Mar. 2016, pp. 83–84.
- [8] X. Wang, Z. Ning, S. Guo, M. Wen, and V. Poor, "Minimizing the age-of-critical-information: An imitation learning-based scheduling approach under partial observations," *IEEE Trans. Mobile Comput.*, early access, Jan. 21, 2021, doi: [10.1109/TMC.2021.3053136](https://doi.org/10.1109/TMC.2021.3053136).
- [9] Y. Dai, D. Xu, S. Maharjan, Z. Chen, Q. He, and Y. Zhang, "Blockchain and deep reinforcement learning empowered intelligent 5G beyond," *IEEE Netw.*, vol. 33, no. 3, pp. 10–17, May/Jun. 2019.
- [10] T. Salaman, M. Zolanvari, A. Erbad, R. Jain, and M. Samaka, "Security services using blockchains: A state of the art survey," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 858–880, Jan. 2018.
- [11] J. Wan, J. Li, M. Imran, and D. Li, "A blockchain-based solution for enhancing security and privacy in smart factory," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3652–3660, Jun. 2019.
- [12] Z. Ning *et al.*, "Intelligent resource allocation in mobile blockchain for privacy and security transactions: A deep reinforcement learning based approach," *Sci. China Inf. Sci.*, vol. 64, no. 6, pp. 1–16, 2021.
- [13] R. Yang, F. R. Yu, P. Si, Z. Yang, and Y. Zhang, "Integrated blockchain and edge computing systems: A survey, some research issues and challenges," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 2, pp. 1508–1532, Feb. 2019.
- [14] U. Majeed and C. S. Hong, "FLchain: Federated learning via MEC-enabled blockchain network," in *Proc. 20th Asia-Pacific Netw. Oper. Manage. Symp. (APNOMS)*, Sep. 2019, pp. 1–4.
- [15] W. Y. B. Lim *et al.*, "Federated learning in mobile edge networks: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 3, pp. 2031–2063, 3rd Quart., 2020.
- [16] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, "Blockchain and federated learning for privacy-preserved data sharing in industrial IoT," *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 4177–4186, Jun. 2020.
- [17] J. Kang, Z. Xiong, D. Niyato, Y. Zou, Y. Zhang, and M. Guizani, "Reliable federated learning for mobile networks," *IEEE Wireless Commun.*, vol. 27, no. 2, pp. 72–80, Apr. 2020.
- [18] J. Xie *et al.*, "A survey of blockchain technology applied to smart cities: Research issues and challenges," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2794–2830, 3rd Quart., 2019.
- [19] F. Tschorsch and B. Scheuermann, "Bitcoin and beyond: A technical survey on decentralized digital currencies," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 3, pp. 2084–2123, 3rd Quart., 2016.
- [20] M. Zhaofeng, W. Lingyun, W. Xiaochang, W. Zhen, and Z. Weizhe, "Blockchain-enabled decentralized trust management and secure usage control of IoT big data," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 4000–4015, May 2020.
- [21] H.-N. Dai, Z. Zheng, and Y. Zhang, "Blockchain for Internet of Things: A survey," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8076–8094, Oct. 2019.
- [22] Z. Ning *et al.*, "Blockchain-enabled intelligent transportation systems: A distributed crowdsensing framework," *IEEE Trans. Mobile Comput.*, early access, May 13, 2021, doi: [10.1109/TMC.2021.3079984](https://doi.org/10.1109/TMC.2021.3079984).
- [23] M. Shen, Y. Deng, L. Zhu, X. Du, and N. Guizani, "Privacy-preserving image retrieval for medical IoT systems: A blockchain-based approach," *IEEE Netw.*, vol. 33, no. 5, pp. 27–33, Sep. 2019.
- [24] K. Zhang, Y. Zhu, S. Maharjan, and Y. Zhang, "Edge intelligence and blockchain empowered 5g beyond for the industrial Internet of Things," *IEEE Netw.*, vol. 33, no. 5, pp. 12–19, Oct. 2019.
- [25] J. Huang, L. Kong, G. Chen, L. Cheng, K. Wu, and X. Liu, "B-IoT: Blockchain driven Internet of Things with credit-based consensus mechanism," in *Proc. IEEE 39th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jul. 2019, pp. 1348–1357.
- [26] S. Abdulrahman, H. Tout, H. Ould-Slimane, A. Mourad, C. Talhi, and M. Guizani, "A survey on federated learning: The journey from centralized to distributed on-site learning and beyond," *IEEE Internet Things J.*, vol. 8, no. 7, pp. 5476–5497, Apr. 2021.
- [27] A. Hard *et al.*, "Federated learning for mobile keyboard prediction," 2018, *arXiv:1811.03604*. [Online]. Available: <http://arxiv.org/abs/1811.03604>
- [28] Y. Lu, X. Huang, K. Zhang, S. Maharjan, and Y. Zhang, "Blockchain empowered asynchronous federated learning for secure data sharing in internet of vehicles," *IEEE Trans. Veh. Technol.*, vol. 69, no. 4, pp. 4298–4311, Apr. 2020.
- [29] J. Kang, Z. Xiong, D. Niyato, S. Xie, and J. Zhang, "Incentive mechanism for reliable federated learning: A joint optimization approach to combining reputation and contract theory," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 10700–10714, Dec. 2019.
- [30] Y. J. Kim and C. S. Hong, "Blockchain-based node-aware dynamic weighting methods for improving federated learning performance," in *Proc. 20th Asia-Pacific Netw. Oper. Manage. Symp. (APNOMS)*, 2019, pp. 1–4.
- [31] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, "Federated learning for data privacy preservation in vehicular cyber-physical systems," *IEEE Netw.*, vol. 34, no. 3, pp. 50–56, May 2020.
- [32] H. Xing, O. Simeone, and S. Bi, "Decentralized federated learning via SGD over wireless D2D networks," in *Proc. IEEE 21st Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, May 2020, pp. 1–5.
- [33] G. Han, S. Li, C. Zhu, J. Jiang, and W. Zhang, "Probabilistic neighborhood-based data collection algorithms for 3D underwater acoustic sensor networks," *Sensors*, vol. 17, no. 2, p. 316, Feb. 2017, doi: [10.3390/s17020316](https://doi.org/10.3390/s17020316).
- [34] Y. Liu, K. Li, Y. Zhang, and W. Qu, "A novel reputation computation model based on subjective logic for mobile ad hoc networks," in *Proc. 3rd Int. Conf. Netw. Syst. Secur.*, 2009, pp. 294–301.

- [35] L. Deng, "The MNIST database of handwritten digit images for machine learning research [best of the web]," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 141–142, Nov. 2012, doi: [10.1109/MSP.2012.2211477](https://doi.org/10.1109/MSP.2012.2211477).
- [36] J. Kang *et al.*, "Blockchain for secure and efficient data sharing in vehicular edge computing and networks," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4660–4670, Jun. 2019.



Jie Meng is currently a Senior Student with the School of Software Technology, Dalian University of Technology, Dalian, China.

Her research interests include gesture recognition and federated learning.



Zhenquan Qin (Member, IEEE) received the B.S. degree in security engineering and the Ph.D. degree from the University of Science and Technology of China, Hefei, China, in 2002 and 2007, respectively.

Currently, he is an Associate Professor with the School of Software, Dalian University of Technology, Dalian, China. His current research interests include unmanned aerial vehicle communications, cross-layer network optimization, and Internet of things.



Bingxian Lu (Member, IEEE) received the B.S., M.E., and Ph.D. degrees from the Dalian University of Technology, Dalian, China, in 2012, 2014, and 2019, respectively.

He is currently an Assistant Professor with the School of Software, Dalian University of Technology. His research interests include wireless network, mobile computing, and pervasive computing applications.

Dr. Lu is a member of the ACM.



Jin Ye received the B.S. degree in computer science and technology from Dalian Ocean University, Dalian, China, in 2019. She is currently pursuing the M.S. degree with the School of Software, Dalian University of Technology, Dalian.

Her current research interests include blockchain and federated learning.



Lei Wang (Member, IEEE) received the B.S., M.S., and Ph.D. degrees from Tianjin University, Tianjin, China, in 1995, 1998, and 2001, respectively.

He is currently a Full Professor and Vice Dean with the School of Software, Dalian University of Technology, Dalian, China. He was a Member of Technical Staff with Bell Labs Research China, Beijing (2001–2004), a Senior Researcher with Samsung, Seongnam, Gyeonggi, South Korea (2004–2006), a Research Scientist with Seoul National University, Seoul, South Korea (2006–2007), and a Research Associate with Washington State University, Vancouver, WA, USA (2007–2008). His research interests involve wireless ad-hoc network, sensor network, social network, and network security.



PERPUSTAKAAN SULTANAH NUR ZAHIRAH

Bahagian Pengurusan Dan Perkhidmatan Maklumat, PSNZ UMT

SELECTIVE DISSEMINATION OF INFORMATION (SDI)

About UMT Faculty SDI

Selective Dissemination of Information (SDI) service is a current-awareness service offered by the PSNZ for UMT Faculty Members. The contents selection criteria include current publications (last 5 years), highly cited and most viewed/downloaded documents. The contents with pdf full text from subscribed databases are organized and compiled according to a monthly theme which is determined based on the topics of specified interest.

For more information or further assistance, kindly contact us at 09-6684185/4298 or email to psnz@umt.edu.my/sh_akmal@umt.edu.my

Thank you.

**Perpustakaan Sultanah Nur Zahirah
Universiti Malaysia Terengganu
21030 Kuala Nerus, Terengganu.
Tel. : 09-6684185 (Kaunter Utama)
Fax : 09-6684179
Email : psnz@umt.edu.my**