2023 11th International Conference on Internet of Everything, Microwave Engineering, Communication and Networks (IEMECON) | 978-1-6654-7512-9/23/$31.00 ©2023 IEEE | DOI: 10.1109/IEMECON56962.2023.10092324

Deep Learning Techniques for Detection of Underwater Acoustic Sources

Sanjana K. R. Prasad and Sanjeev Gurugopinath
Department of Electronics and Communication Engineering, PES University, Bengaluru 560085, India
Emails: sanjanaprasadkr@gmail.com, sanjeevg@pes.edu

***Abstract*—We consider the problem of detecting underwater acoustic sources, using a linear array (LA) of hydrophones, using deep learning. In particular, we consider the following architectures – convolutional neural network (CNN), gated recurrent unit, long short term memory (LSTM), plain deep neural network and recurrent neural network. We present a performance comparison study of these architectures, in terms of probability of detection for a fixed false-alarm probability. We consider three datasets including (i) synthetic data generated from a vertical LA, with generalized Gaussian noise, (ii) ShipsEar: An underwater vessel noise database, and (iii) Historic naval sound and video database. Through extensive experiments, we show that CNN and LSTM outperform the other techniques in low and high SNR scenarios, respectively.**

***Index Terms*—Acoustic signals, deep learning, detection, convolutional neural networks, long short term memory.**

I. Introduction

Underwater source detection is a salient domain in underwater acoustic signal processing, which has been receiving increasing attention in environmental, commercial, military and civilian explorations. However the complexity of the ocean environment poses manifold challenges. Underwater acoustic channels suffer from relative fading and are subjected to ample sound reflection and refraction, which inevitably results in the distortion of radiated noise [1], [2]. Multi-target interference merged with the time-varying and non-linear ambient noise of the ocean exacerbates the problem [3], [4]. Additionally, oceanic environments are characterised by small signal-to-noise ratios (SNR), which hinders the accuracy of traditional detection techniques, and also introduces the problem of frequent false alarms.

Machine learning (ML) [5], [6], a subset of artificial intelligence (AI) that iterates big datasets over complex algorithms to produce reliable predictions [7], has made significant breakthroughs in this field, and is found to perform more robustly than conventional techniques [8]–[11]. The evolution of big data has facilitated tremendous growth in the popularity of sophisticated and data-driven deep learning (DL) methods. The DL is capable of automatic feature extraction and has superior optimization capabilities. For the classification of underwater target signals from noise, it is found that compared to traditional ML methods, DL methods achieve higher accuracy [12]. Along these lines, DL methods such as convolutional neural network (CNN), deep belief network, deep long short-term memory (LSTM) network, deep auto-encoder neural network and using feature extraction module have been explored [13]–[18].

The aim of this work is to experimentally compare the performance of popular deep learning classification algorithms for the detection of an underwater stationary acoustic man-made target. To evaluate the algorithms we use the probability of detection (P_d) as the primary metric, constraining the probability of false alarm (P_f). In particular, we have considered the following algorithms: CNN, plain deep neural network (DNN), LSTM, gated recurrent unit (GRU), and recurrent neural network (RNN). The algorithms have been tested on three datasets; one model-based simulated dataset generating by following the normal-mode theory with generalized Gaussian noise, and two experimental real-world datasets acquired from ShipsEar: An underwater vessel noise database [19], and the historic naval sound and video database website [20]. For the noise model, we choose the generalized Gaussian distribution, which is known to be of practical pertinence in the ocean environment. From the ShipsEar dataset, we use the experimentally recorded natural ambient noise. From our comprehensive experimental results, we infer that the CNN and the LSTM recurrently outperform other supervised deep learning algorithms, and hence are competent candidates for practical research applications. In particular, CNN is preferred, given its lesser constraints on training time and better performance in low SNR scenarios.

II. Data Model and Problem Formulation

We assume a range-independent ocean. For signal acquisition, consider an array of P hydrophones arranged vertically with uniform interspacing distance s, located at the far field region of an acoustic source [21]. We aim to detect the presence of the acoustic source at unknown range and deptn (r, z) using Q number of snapshots of the received signal from each p^{th} sensor, $p = 1, \ldots, P$. The ocean model considers water layer depth between $0 \leq w \leq W$, sound speed profile $v(w)$ in water, and constant values of density $d(w)$ and other underwater environmental parameters. Detection of this signal is challenging due to the presence of ambient noise from the ocean. Following the theory of normal modes, the signal can be written as

$$z(q) = \alpha(q) \sum_{m=1}^{M} \mathbf{a}_m b_m(r,z) = \alpha(q)\mathbf{A}\mathbf{b}(r,z), \tag{1}$$

where

$$\mathbf{b}(r,z) = \begin{bmatrix} b_1(r,z) & \cdots & b_M(r,z) \end{bmatrix}^T, \tag{2}$$

$$b_m(r,z) = \sqrt{\frac{2\pi}{\zeta_m r}} \psi_m(z) e^{-j\frac{\pi}{4}} e^{j\zeta_m r - \xi_m r}, \tag{3}$$

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}_1 & \cdots & \mathbf{a}_M \end{bmatrix}^T, \tag{4}$$

$$\mathbf{a}_m = \begin{bmatrix} \psi(z) \ \psi(z_0+s) \ \cdots \ \psi(z_0+(P-1)s) \end{bmatrix}^T, \tag{5}$$

$m = 1, \ldots, M$, with M modes, $\alpha(q),\ q = 1, \ldots, Q$ denoting the signal amplitude, $\mathbf{b}(r,z)$ being the mode amplitude vector, z_0 being the topmost sensor depth, and $\{\psi_1, \ldots, \psi_M\}$, $\{\zeta_1, \ldots, \zeta_M\}$, $\{\xi_1, \ldots, \xi_M\}$ denoting the mode functions, wavenumbers and attenuation coefficients, respectively.

It is well-known that the ambient ocean noise does not typically follow a Gaussian distribution. In this work, we assume that the entries of the $P \times 1$ noise vector $\mathbf{n}(q)$ for $q = 1, \ldots, Q$ are distributed according to a single-parameter generalized Gaussian distribution (GGD). This model has ample use in several fields in signal processing and communications due to its flexible parametric form [22]. The probability density function of an entry of $\mathbf{n}(q)$ across q can be written as [23]

$$f(u) = \frac{\beta}{2\beta\Gamma(\frac{1}{\beta})} \exp\left(\frac{-|u|^\beta}{\alpha}\right), \tag{6}$$

where $\Gamma(\cdot)$ is the gamma function, and the parameter $\beta \in (0,2]$ controls the shape of the distribution. In particular, the tails of the distribution become *heavier* when β decreases. Well-known distributions are special cases of GGD, such as the Laplace distribution with $\beta = 1$, and the Gaussian distribution with $\beta = 2$ [24].

We consider the following hypothesis testing problem for our underwater acoustic source detection problem:

$$\begin{aligned} \mathcal{H}_0 &: \mathbf{x}(q) = \mathbf{z}(q), \\ \mathcal{H}_1 &: \mathbf{x}(q) = \mathbf{z}(q) + \mathbf{n}(q), \end{aligned} \tag{7}$$

for $q = 1, \ldots, Q$, where $\mathcal{H}_0$ denotes the noise-only hypothesis and $\mathcal{H}_1$ denotes the signal-present hypothesis. In other words, this is a binary classification problem, with the decision to be made towards one of the above two hypotheses. In the next section, we describe the deep learning architectures considered in this paper to solve the binary classification problem at hand.

III. Deep Learning Techniques

Deep learning uses intricate algorithms that can process large databases of both structured and unstructured data. The model continually updates network parameters to improve its estimate of the predicted outputs. We use the following algorithms to solve our problem of binary classification, with our two labelled classes.

A. Convolutional Neural Network

Convolutional neural networks (CNNs) are capable of automatic feature fusing and extraction. The architecture of CNNs is characterized by convolutional layers that apply convolutions to the input to create a feature map, pooling layers that summarize the most important information, and a fully connected layer that predicts the final output after flattening the outputs of the previous layers. The CNN architecture also makes use of kernels, stride and padding. CNN boasts of local dependency and scale invariance. Its operational expression is given by

$$h = i * w, \tag{8}$$

where $*$ denotes the convolution operation, w denotes the kernel, i denotes the input and h denotes the feature map, which can indicate local interactions [25]. Each feature map is computed as

$$h_j = i * w_j + b_j \tag{9}$$

where $j = 1, ..., L$ and b_j represent the bias parameters. 1D CNNs perform well with raw time series data, and compared to 2D CNNs offer advantages such as low computational complexity and faster training.

B. Plain Deep Neural Network

Plain deep neural networks (DNNs) can learn and model non-linear and complex interactions. DNNs are organized into an input layer through which the input dataset is passed, hidden layers consisting of neural connections, and the output layer which gives the prediction. Input data is fed across intermediary neurons, subjected to weights and bias:

$$Y = \sum wi + b, \tag{10}$$

where w stands for weight and decides the amount of influence the input has on the output, i stands for input and b stands for bias which helps in shifting the activation function. The sum of these terms then form the argument of an activation function, which injects non-linearity thus aiding in the output decision making process. Another essential component is back-propagation, which minimises the cost function by deploying the mathematical chain rule.

C. Recurrent Neural Network

Recurrent neural networks (RNNs) utilize memory, which allow them to retain information on previous calculations. RNNs contain recurrent loops over time (or sequence) called feedback loops. Simple RNN architecture comprises of three layers: input layer which takes input as a sequence of vectors through time, recurrent hidden layers where recurrent connections link hidden units together, and finally the output layer which gives the prediction. A simple model is:

$$y_n = \sigma(Gh_{n-1} + Ws_n + b), \tag{11}$$

where n represents the discrete index and $n = 0, \ldots, M$. The input vector sequence is denoted by s_n and the output is denoted by h_n, which is calculated using the nonlinear function σ_n. Parameters G, W and b are to be determined by training [26]. RNNs have the ability to identify a sequential patterns and data correlations, however the vanishing gradient problem is a limitation.

D. Long Short Term Memory

Long short term memory (LSTM) successfully addresses the problem of gradient disappearance in RNNs, thus enabling the capture of long-term dependencies in data. LSTM contains special units of connected memory blocks in the recurrent hidden layer, which maintain their state over time. The model is characterized as:

$$\tilde{c}_t = g(x_t W_c + h_{t-1} U_c + b_c), \tag{12}$$
$$c_t = \sigma(f_t c_{t-1} + i_t \tilde{c}_t), \tag{13}$$
$$h_t = g(c_t) o_t, \tag{14}$$

where the hidden state vector, the cell input activation vector and the cell state vector are denoted by h_t, $\tilde{c}_t$, and c_t, respectively [27]. A vanilla LSTM unit consists of an input gate, a forget gate and an output gate. The forget gate adaptively resets the cell's memory which prevents the processing of input streams that were not segmented into subsequences.

E. Gated Recurrent Unit

Gated recurrent unit (GRU) networks overcome the problem of vanishing gradient and can capture dependencies on large sequences of data. GRUs consist of the reset gate which is analogous to the combination of input and forget gate present in LSTM architecture, and the update gate which is analogous to the output gate. Using the update and the reset gate, the model controls information flow. They retain information for large periods of time. GRUs also have lesser parameters, which enables faster training. It is also found that GRUs perform better than LSTM networks when the complexity of the sequences are low.

IV. Performance Metrics

In order to comparatively understand and analyze the considered deep learning classification algorithms, we consider the probability of detection as the primary metric for evaluation, whilst fixing the probability of false-alarm. We consider the "positive" case to be when the algorithm classifies an instance as "signal present", and "negative" when classified as "noise-only".

Probability of detection (P_d): The probability of detection indicates the number of samples that the algorithm manages to correctly classify as positive, out of all the positive samples present in the dataset. We aim to attain a high P_d, which is indicative of the algorithms correctly detecting the presence of the source signal. It is given by

$$P_d = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{15}$$

where TP and FN indicate true positives and false negatives, respectively.

Probability of false alarm (P_f): The probability of false alarm indicates the number of samples that the algorithms incorrectly classifies as positive, out of all the negative samples present in the dataset. A false alarm occurs when the signal from the source is detection as present, when in fact there was only noise received. It is given by

$$P_f = \frac{\text{FP}}{\text{FP} + \text{TN}}, \tag{16}$$

where FP and TN indicate false positives and true negatives, respectively. Recall that the number of correctly detected source signals is given by the parameter TP, the number of correctly detected noise only signals is given by false positive TN, the number of wrongly detected source signals is given by FP, and the number of wrongly detected noise only signals is given by FN. A large number of false alarms adversely affect signal detection, and we therefore we constraint the P_f to be a fixed value, and subsequently obtain the P_d values corresponding to this constraint.

In the next section, we discuss the three datasets considered in this paper for performance evaluation, and also describe the preprocessing carried on the experimental datasets. In particular, we consider one model-based synthetic data, and two experimental datasets namely ShipsEar and historic naval sound and video.

V. Datasets and Preprocessing

A. Model-Based Simulated Dataset

We follow the normal mode theory to construct our simulated dataset of acoustic signal samples using Monte Carlo simulations, which uses repeated random sampling to obtain reliable results. We consider signals ranging from -15 dB to 15 dB, as these are representative of practical SNR values in the realms of the ocean. We pick the following ocean parameters for the dataset: ratio of densities towards the ocean floor and water = 1.5, attenuation towards the ocean floor = 0.2 dB/wavelength, ocean depth = 150 m speed of sound in water = 1500 m/s and speed of sound towards the ocean floor = 1700 m/s. We have computed the eigenfunctions, wavenumbers and attenuation coefficients using the Kraken normal mode program [28], which efficiently models range-independent oceanic environments. The noise samples following GGD have been generated for three values of the shape parameter β, in particular β = 0.5, 1 and 2.

B. ShipsEar: An Underwater Vessel Noise Database

To verify the authenticity of our results, we perform a similar classification task on real-world experimental datasets. We obtain samples from the ShipsEar database [19] of real underwater vessel sounds captured on the Atlantic coast of Spain, which consists of 91 records of 11 types of vessels. The signals were recorded at shallow waters having maximum water depth under 45 m, with both natural as well as man-made environmental noises. DigitalHyd SR-1 hydrophones were used for recording. Three hydrophones were secured to the bottom of a submerged buoy at varying depths of 5.5 m, 6.5 m and 7.5 m from the sea bottom. For depths below 10 m, two hydrophones were placed at depths 2.5 m and 3.5 m. We consider the following six classes of recorded sound clips: Motorboat, Passenger, Sailboat, Tugboat, Trawler, Dredger. Clean noise samples are available which were used for obtaining "noise-only" samples. The clean noise samples are manually added to the clean signal, to set a desired SNR.

C. Historic Naval Sound and Video

We also obtain real underwater sounds from the historic naval sound and video database website [20], where we consider three recordings of United State Navy submarines from the sonar training record series D16. These recordings were provided by the Naval undersea warfare center (NUWC) new port. We analyze these targets under different circumstances, each with a few modifications. We consider a submarine that is: (a) approaching a hydrophone, (b) moving towards and away from a hydrophone subjected to slight environmental noise, and (c) with propulsion subjected to slight environmental noise. These datasets are referred to in this paper as *real dataset* A, B and C, respectively.

D. Preprocessing on Real World Data

Raw recordings from both the real-world experimental datasets in Sections V-B and V-C are preprocessed to improve the performance of the classifiers. The recordings are converted to .wav format, after which they are resampled to 2560 Hz. We then take the fast Fourier transform (FFT), following

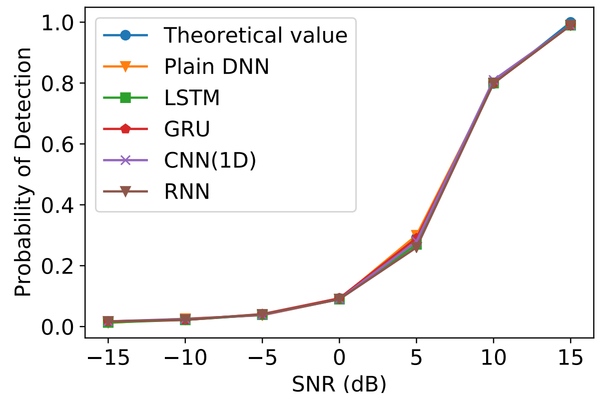


Fig. 1: Model-based simulated dataset with theoretical limit.

which all the samples are normalized to obtain a common scale. For these datasets, we add ambient noise samples present in the ShipsEar database. We generate $200,000$ samples, which are split into training and testing sets in an 80% - 20% ratio respectively, for all the three datasets.

VI. Results and Discussion

In this section, we present a study on the performance of each of the considered deep learning techniques and compare them across the three databases. Throughout the experiements, we set $P_f = 0.01$.

A. Model-Based Simulated Dataset

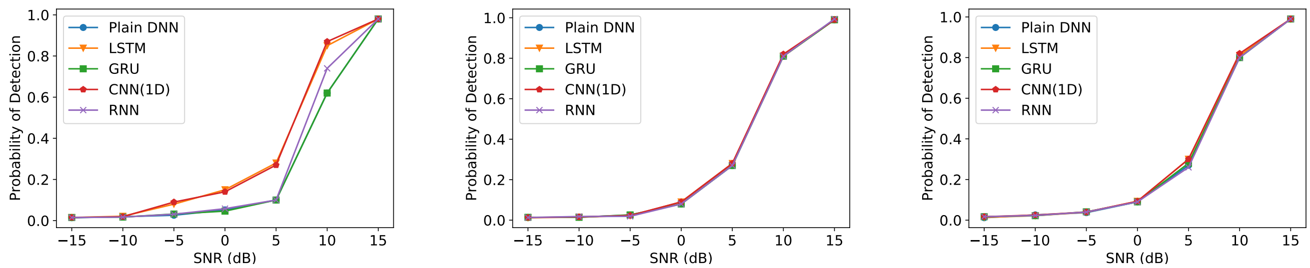
First, we start with the simple case of Gaussian noise with $\beta = 2$. Figure 1 shows the variation of P_d for each technique with respect to SNR. In the case of $\beta = 2$, the theoretical bound on P_d can be obtained as follows:

$$P_d^{\text{theory}} = Q\left(Q^{-1}(P_f) - \sqrt{\text{SNR}}\right), \tag{17}$$

which is achieved by using the matched filter, where $Q(\cdot)$ denotes the complementary cumulative function of a standard Gaussian. It is evident from the graph that all the algorithms achieve the theoretical limit with marginal differences, thus exhibiting their strong reliability. Next, we consider the effect of β. Figures 2a, 2b and 2c shows the variation of P_d for β = 0.5, 1 and 2, respectively. It is observed that across SNR, the performance of every algorithm improves with a decrease in β. For low β, it is observed that CNN and LSTM significantly outperform all the other algorithms. At higher SNR, RNN is observed to be suboptimal.

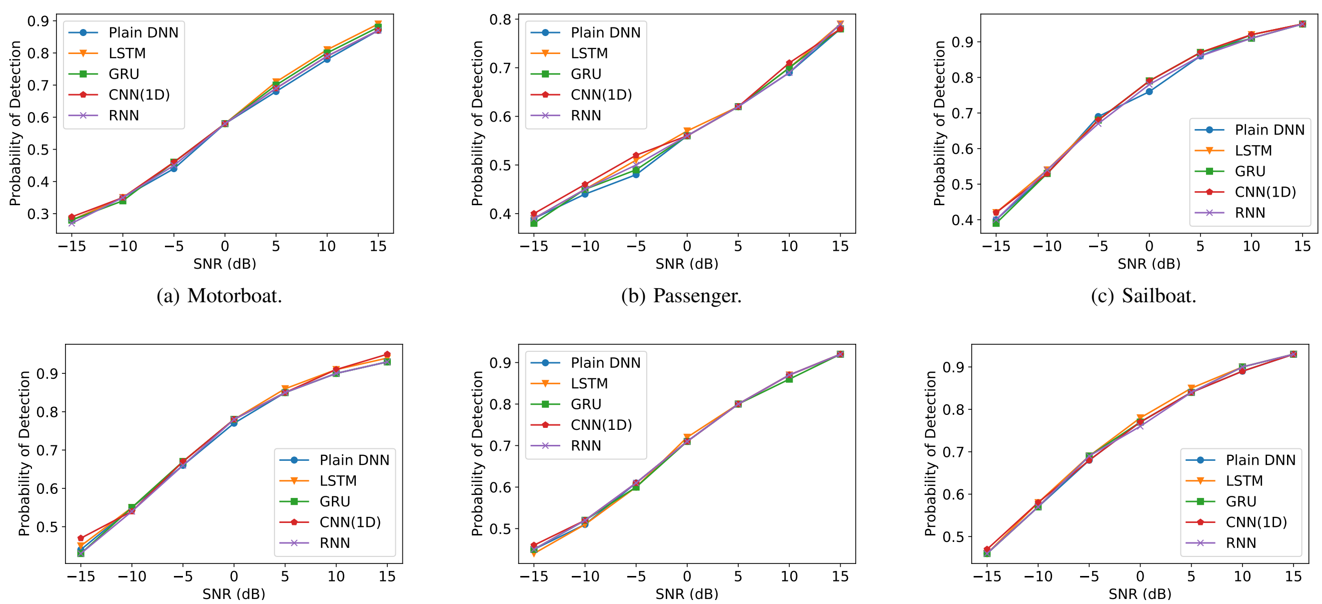
B. Experimental Datasets

We consider the ShipsEar dataset, and the corresponding performances across the six chosen scenarios are given in



(a) Variation with SNR, with $\beta = 0.5$.
(b) Variation with SNR, with $\beta = 1$.
(c) Variation with SNR, with $\beta = 2$.

Fig. 2: Performance comparison of the deep learning algorithms for the model-based dataset under GGN, for different values of β.



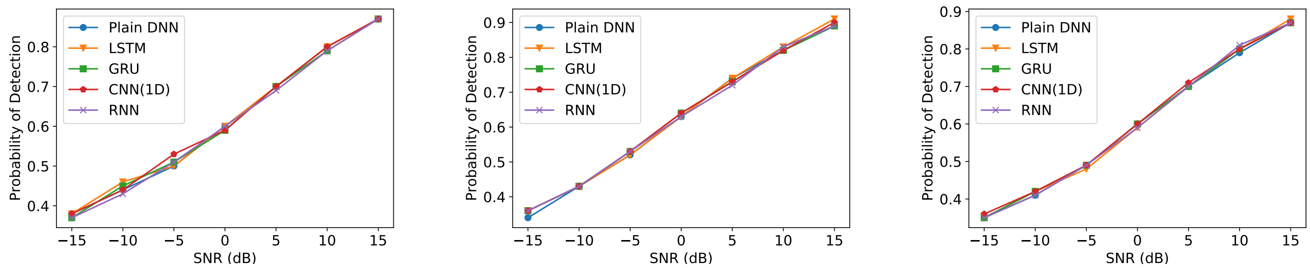
(a) Motorboat.
(b) Passenger.
(c) Sailboat.
(d) Tugboat.
(e) Trawler.
(f) Dredger.

Fig. 3: Performance comparison of the deep learning algorithms for the ShipsEar dataset.

Figs. 3a–3f. The performances corresponding to historic naval sound and video database are given in Figs. 4a–4c. All the algorithms are able to achieve a good P_d, ranging from 0.75 to 0.95 at 15 dB. Even in this case, it is observed that CNN and LSTM slightly outperform other algorithms. In particular, CNN is observed to preform particularly well in low SNR scenarios, while LSTM offers a good performance at high SNR values. However, compared to LSTM, CNN has low computational complexity with a faster training time and is thus a strongly preferable choice. Moreover, note that low SNR scenarios are practically more meaningful than high SNR conditions.

VII. Conclusion

We compared the performances of five deep learning techniques for detection of underwater acoustic sources. We considered three datasets: one based on model-based synthetic dataset with GGD noise, and two datasets from real-world experiments. CNN and LSTM techniques offer good probability of detection performances in low and high SNR scenarios, respectively. Given the practical relevance of a low SNR scenario, it can be concluded that CNN gives the best performance. Less computational complexity and less training time in CNN makes it a suitable choice.



(a) Real Dataset A. (b) Real Dataset B. (c) Real Dataset C.

Fig. 4: Performance comparison of the deep learning algorithms for the Historic Naval Sound and Video dataset.

References

[1] P. Qarabaqi and M. Stojanovic, "Statistical characterization and computationally efficient modeling of a class of underwater acoustic communication channels," *IEEE Journal of Oceanic Engineering*, vol. 38, no. 4, pp. 701–717, 2013.

[2] J. Li and H. Yang, "The underwater acoustic target timbre perception and recognition based on the auditory inspired deep convolutional neural network," *Applied Acoustics*, vol. 182, p. 108210, 2021.

[3] S. Siddagangaiah, L. Yaan, X. Guo, and K. Yang, "On the dynamics of ocean ambient noise: Two decades later," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 25, p. 103117, 10 2015.

[4] S. Banerjee and M. Agrawal, "Underwater acoustic noise with generalized gaussian statistics: Effects on error performance," in *2013 MTS/IEEE OCEANS - Bergen*, 2013, pp. 1–8.

[5] A. Singh, N. Thakur, and A. Sharma, "A review of supervised machine learning algorithms," in *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, 2016, pp. 1310–1315.

[6] S. Kotsiantis, "Supervised machine learning: A review of classification techniques." *Informatica (Slovenia)*, vol. 31, pp. 249–268, 01 2007.

[7] J. Qiu, Q. Wu, G. Ding, Y. Xu, and S. Feng, "A survey of machine learning for big data processing," *EURASIP Journal on Advances in Signal Processing*, vol. 2016, 05 2016.

[8] W. Zhang, Y. Wu, J. Shi, H. Leng, Y. Zhao, and J. Guo, "Surface and underwater acoustic source discrimination based on machine learning using a single hydrophone," *Journal of Marine Science and Engineering*, vol. 10, no. 3, 2022. [Online]. Available: https://www.mdpi.com/2077-1312/10/3/321

[9] M. J. Bianco, P. Gerstoft, J. Traer, E. Ozanich, M. A. Roch, S. Gannot, and C.-A. Deledalle, "Machine learning in acoustics: Theory and applications," *The Journal of the Acoustical Society of America*, vol. 146, no. 5, pp. 3590–3628, 2019.

[10] W. Zhang, Y. Wu, Y. Lin, L. Ma, K. Han, Y. Chen, and C. Liu, "Underwater target detection based on machine learning," in *2020 IEEE 3rd International Conference on Information Communication and Signal Processing (ICICSP)*, 2020, pp. 210–214.

[11] J. Choi, Y. Choo, and K. Lee, "Acoustic classification of surface and underwater vessels in the ocean using supervised machine learning," *Sensors*, vol. 19, no. 16, 2019. [Online]. Available: https://www.mdpi.com/1424-8220/19/16/3492

[12] I. McLoughlin, H. Zhang, Z. Xie, Y. Song, and W. Xiao, "Robust sound event classification using deep neural networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 3, pp. 540–552, 2015.

[13] S. Kamal, S. K. Mohammed, P. R. S. Pillai, and M. H. Supriya, "Deep learning architectures for underwater target recognition," in *2013 Ocean Electronics (SYMPOL)*, 2013, pp. 48–54.

[14] H. Yang, G. Xu, S. Yi, and Y. Li, "A new cooperative deep learning method for underwater acoustic target recognition," in *OCEANS 2019 - Marseille*, 2019, pp. 1–4.

[15] H. Yang, S. Shen, X. Yao, M. Sheng, and C. Wang, "Competitive deep-belief networks for underwater acoustic target recognition," *Sensors*, vol. 18, no. 4, 2018.

[16] H. Yang, J. Li, S. Shen, and G. Xu, "A deep convolutional neural network inspired by auditory perception for underwater acoustic target recognition," *Sensors*, vol. 19, no. 5, 2019. [Online]. Available: https://www.mdpi.com/1424-8220/19/5/1104

[17] X. Luo, Y. Feng, and M. Zhang, "An underwater acoustic target recognition method based on combined feature with automatic coding and reconstruction," *IEEE Access*, vol. 9, pp. 63 841–63 854, 01 2021.

[18] D. Li, F. Liu, T. Shen, L. Chen, X. Yang, and D. Zhao, "Generalizable underwater acoustic target recognition using feature extraction module of neural network," *Applied Sciences*, vol. 12, no. 21, 2022. [Online]. Available: https://www.mdpi.com/2076-3417/12/21/10804

[19] D. Santos, S. Torres-Guijarro, A. Cardenal, and A. Pena, "Shipsear: An underwater vessel noise database," *Applied Acoustics*, vol. 113, pp. 64–69, 12 2016.

[20] H. Yue, L. Zhang, D. Wang, Y. Wang, and Z. Lu, "The classification of underwater acoustic targets based on deep learning methods," in *Proceedings of the 2017 2nd International Conference on Control, Automation and Artificial Intelligence (CAAI 2017)*. Atlantis Press, 2017/06, pp. 526–529.

[21] P. V. Nagesha, G. V. Anand, N. Kalyanasundaram, and S. Gurugopinath, "Detection, enumeration and localization of underwater acoustic sources," in *2019 27th European Signal Processing Conference (EUSIPCO)*, 2019, pp. 1–5.

[22] A. Dytso, R. Bustin, H. V. Poor, and S. Shamai, "Analytical properties of generalized gaussian distributions," *Journal of Statistical Distributions and Applications*, vol. 5, 12 2018.

[23] S. Gurugopinath, R. Muralishankar, and H. N. Shankar, "Differential entropy-driven spectrum sensing under generalized gaussian noise," *IEEE Communications Letters*, vol. 20, no. 7, pp. 1321–1324, 2016.

[24] R. Sugiura, Y. Kamamoto, and T. Moriya, "Shape parameter estimation for generalized-gaussian-distributed frequency spectra of audio signals," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 736–740.

[25] D. Finol, Y. Lu, V. Mahadevan, and A. Srivastava, "Deep convolutional neural networks for eigenvalue problems in mechanics," *International Journal for Numerical Methods in Engineering*, vol. 118, 01 2018.

[26] F. M. Salem, "A basic recurrent neural network model," 2016. [Online]. Available: https://arxiv.org/abs/1612.09022

[27] B. C. Mateus, M. Mendes, J. T. Farinha, R. Assis, and A. M. Cardoso, "Comparing lstm and gru models to predict the condition of a pulp paper press," *Energies*, vol. 14, no. 21, 2021. [Online]. Available: https://www.mdpi.com/1996-1073/14/21/6958

[28] M. B. Porter, *The KRAKEN normal mode program*, 1992.