Marine Acoustic Signature Recognition using Convolutional Neural Networks

Alexandre Martins Correia
alexandre.m.correia@tecnico.ulisboa.pt
Instituto Superior Técnico, Universidade de Lisboa, Lisboa, Portugal

July 2021

***Abstract*—In a marine environment, there is a great diversity of sound sources, such as marine animals, natural phenomena and man-made activity. Differentiating these sound sources is an important response to ecological challenges. It also ensures better control of the coastline and global ocean noise. The current work aims to devise a model which analyses acoustic signals from hydrophones and classifies them according to the sound source. To achieve this, a convolution neural network (CNN) composed of three convolutional layers followed by two fully connected layers is proposed as a classifier, using the mel spectrogram representation divided into small intervals (windows) of the acoustic signal and its derivatives as input. Class scores are assigned to each window by the CNN. The developed methodology is applied to two different datasets composed of hydroacoustic data. The first comprises vessel noise data (ShipsEar dataset), where the objective is to detect the presence or absence of vessels and distinguish the vessels according to the size. A classification accuracy of 83.2% and 88.8% is achieved using the mel spectrogram and the mel spectrogram plus its first and second derivatives as features, respectively. The second dataset complements the previous one by adding dolphin and whale vocalizations, in order to extend the diversity of sound sources. Three data augmentation techniques (time stretching, pitch shifting and time shifting) are studied in order to extend the dolphin and humpback whale training data. Whichever of the three techniques is used individually outperform the model without data augmentation. This is equal true when all three techniques are used simultaneously. The impact of the window length is also studied, with five different models being creating where the window length varies between 0.22 s and 1.97 s. The classification accuracy ranges between 66.2% and 78.3%.**

Keywords: Hydroacoustic signal recognition, convolutional neural network, mel spectrogram, ShipsEar dataset, vessel noise, marine animal vocalizations

I. Introduction

KNOWLEDGE of the marine environment, in its various facets, is crucial for its protection and enhancement. Essential part of this knowledge relates to sound, whether it be produced naturally or through man-made activity. The task of detecting relevant sounds and classifying them plays an important role in studying marine biodiversity and on vessel monitoring. Thus, marine acoustic signature recognition complements other ocean monitoring techniques based on underwater image classification [1], [2], [3], and can even complement maritime surface imaging from sources, such as from cameras in ports or drones [4], [5]. We should note that underwater imaging is limited to less than fifty meters at best, whereas sound waves can travel up to as much as thousands of kilometers in seawater.

Over time, researchers have conducted various studies in recognition techniques of vessel sounds and animals vocalizations. These are usually split into two phases. Firstly, the hydroacoustic signals undergo to processing techniques in order to extract features of interest, the so-called feature space, and, then, classification algorithms (classifiers) are applied using as input the features extracted. In general, the processing techniques consist in converting the signal to the frequency domain.

Some approaches are based on the assumptions that a particular known model can fit the data correctly, such as the autoregressive (AR) model. Since the model coefficients may represent the power spectral density (PSD), these can be seen as useful features for classification. Huang et al. [6] proposed a method to classify noise from four different ships using the AR model. However, AR model poles are used as features instead of the coefficients, with the justification that coefficients will vary greatly with the change of environment, position and orientation of ships. The nearest neighbour algorithm is used as classifier. Bennett et al. [7] has also used the AR model, but this time using its coefficients as feature space. In this study, another processing method was applied, based on the wavelet transform. Thus, the AR coefficients and the average energy contained in the wavelet coefficients were used as inputs to a neural network, in order to classify geological processes (earthquake data) and biological signals of five different species of whales.

Another common feature extraction approach is based on computing the PSD with the Fast Fourier Transform (FFT) [8], [9], or using Welch's method [10]. In fact, the PSD is quite a useful tool to extract the characteristics of a signal, because it may tell at which frequency ranges variations are strongest. However, PSD does not give information about the frequencies at each instant of time, since it computes a statistical average of the whole signal. Therefore, it is only recommended for stationary signals, whose variance and mean do not change with time. In the cases here dealt with, underwater signals are always non-stationary. As an alternative, the feature space can be generated by the STFT resulting in a spectrogram, which is a 2D image containing the noise signal frequency content changing with time. In this way, the implementation of the

spectrogram as the feature space permits the application of a new classification algorithm in the field of acoustic signal recognition (ASR), known as convolutional neural networks (CNNs). These are typically used for image recognition [11], and in here, CNNs try to find patterns in the spectrogram which are common to the same class.

In the work of Garcia et al. [12], in which five distinct classifiers were studied in order to distinguish fin whale vocalizations, the application of CNNs using spectrograms as inputs is highlighted. The same methodology was used by Harvey et al. [13] in the detection of humpback whales and by Belghith et al. [14] in the classification of underwater acoustic signals from widely diverse sources (natural sounds, animal vocalization and man-made activity), which is in some respect similar to the objective of this work.

Apart from hydroacoustic signal recognition, a considerable number of studies have been dedicated to ASR of a wide variety of sounds. UrbanSound8k [15] is a known dataset in this area, which consists of a fairly complete data of urban sounds. Several studies have been performed using this dataset, where spectrogram and CNNs have an important role [16], [17]. These two studies provide the basis for the developed methodology which is applied to marine environment acoustic noise. The data collected by the hydrophones is converted to a mel spectrogram representation and classified using a CNN. The model assigns scores to the default classes over time. Figure 1 shows a simplified scheme of the process developed to recognize marine acoustic signatures.

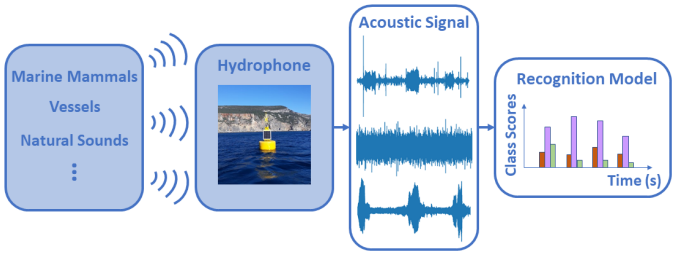

Fig. 1. Marine acoustic signature framework

The remainder of the article is organized as follows. In Section II, the recognition framework is introduced, explaining in detail the general steps of the feature extraction and modeling process. A testing of the recognition framework with a known dataset of non-marine sounds is also analysed. In Section III and IV, two datasets composed of hydroacoustic data used are described and several models applying the recognition framework are assembled for these two datasets. The results obtained are presented. Finally, in Section V, an overall inference of the work performed is described and there are a few recommendations for future work.

II. Knowledge Discovery Process of Recognition Framework

The recognition framework was constructed based on the knowledge discovery process described in [18], which can be divided into four phases: a) Data selection, b) Feature extraction, c) Modeling and d) Evaluation. In Figure 2, the procedures adopted in each of the phases are shown.

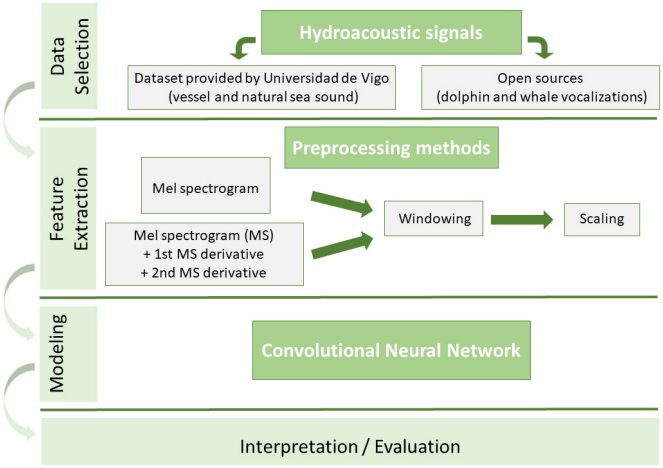

Fig. 2. Knowledge discovery process of the recognition framework

A. Data selection

Two main datasets have been used. One is provided by Universidad de Vigo, which is composed by vessel recordings and background noise (ShipsEar dataset [19]). The other complements the previous one by adding dolphin and whale vocalizations from different open sources, in order to spread the diversity of sound sources. These two datasets are described in detail in Chapter III and IV, where the several models based on the application of recognition framework to these datasets are shown.

In addition, in order to test the recognition framework, the developed methodology is applied to non-marine data (UrbanSound8k dataset).

B. Feature extraction

According to [20], feature extraction can be decomposed in two steps: *feature construction* and *feature selection*. The former involves converting ”raw” data into significant features and it is usually performed by preprocessing methods. The latter chooses the most relevant features which will have a higher influence on the desired outcomes of the dataset. In the recognition framework, *feature construction* includes three preprocessing transformations: conversion to mel spectrogram, windowing and scaling. In addition, an extra step is also embraced giving rise to a new approach, which corresponds to the calculation of the first and second mel spectrogram derivatives. Therefore, two different feature extraction methods are part of the recognition framework.

The modeling process also integrates a preprocessing transformation, since CNNs are composed by a set of convolution layers which applies filters to the input in order to extract knowledge and converting into features. Hence, *feature selection* is not taken into account, since the last preprocessing method is inserted into the modeling process.

The spectrogram is computed by applying the short-time fourier transform into the signal, which is framed using the

Hann window with a length of $M = 2048$ $samples$ (39 ms at 52734 Hz) and a hop between frames of $R = 1024$ $samples$ (19 ms at 52734 Hz). Then, the spectrogram is converted into a mel scale with 60 bands. Thus, the mel spectrogram corresponds to a matrix of size $(60,\ \#frames)$. Finally, the mel spectrogram coefficients are converted to decibels. The mel spectrogram is produced using the *Librosa* library [21].

The feature extraction can be expanded by adding the first and second derivatives of the mel spectrogram, in order to incorporate information about the dynamic behaviour of the parameters throughout the frequency axis. This process is based on [22], which combined the cepstral coefficients and its first and second derivatives as feature vector of a Gaussian mixture model in order to classify dolphin vocalizations. The derivatives are computed from the difference between the mel spectrogram coefficients (c), before being converted to decibels, along the frequency axis. As a 60 mel scale filter bank is used, the first derivative can by determined as follows:

$$\begin{cases} d_i = \frac{c_{i+1} - c_i}{2}, & i = 1, \\ d_i = \frac{c_{i+1} - c_{i-1}}{2}, & 2 \leq i \leq 59, \\ d_i = \frac{c_i - c_{i-1}}{2}, & i = 60. \end{cases} \quad (1)$$

This process is repeated for each frame, creating a matrix with the first derivative values, which has the same size as the mel spectrogram matrix. The second derivative is also computed using the previous formula, but applied to the first derivative matrix rather than to the mel spectrogram coefficients. Thus, the mel spectrogram and its first and second derivatives form a three-dimensional matrix of size $(60,\ \#frames,\ 3)$.

Subsequently, to the mel spectrogram of each signal is applied a *rectangular window function* with a specific length in frames, dividing the mel spectrogram into a set of windows with the same size for a complete recording. The duration of each window in seconds corresponds to a number of frames as follows:

$$window\ length\ (frames) = \frac{window\ length\ (seconds) \times sr - O}{R}, \quad (2)$$

where sr is the sampling rate and O is the overlap length between segments ($O = M - R$). Windowing process is also applied to the first and second derivative matrices in the same way as with the mel spectrogram.

Finally, in order to bring all the features to the same level of magnitude, standardization is applied as *scaling* process. The formula is given by:

$$c_i{}' = \frac{c_i - mean(c)}{stdev(c)}. \quad (3)$$

Standardization changes the mean and standard deviation of the features to 0 and 1, respectively, and since it does not have boundaries, outliers do not have a negative impact on scaling. When each window is composed by a three-dimensional matrix (mel spectrogram and first and second derivatives), scaling is applied to each one separately.

C. Modeling

The CNN architecture consists in three convolutional layers followed by two fully connected layers. The first convolutional layer uses 24 filters and the second and the third convolutional layers use 48 filters. The filter size (F) is 5x5 and there is a zero-padding (P) equal to 2. The filter slices along the input with a stride (S) of 1. In the three convolutional layers and in the first fully connected layer, *ReLU* is used as an activation function. For the output layer, the typical activation function in multiclass classification is used: *softmax* function. The size of the fully connected output layer takes into account the number of classes.

In addition, there are two max-pooling layers interspersed among the three convolutional layers, with a filter size and a stride depending on the size of the input window. Batch normalization is performed on each convolutional layer. With regard to the two fully connected layers, in order to avoid overfitting, dropout and L2-regularization are applied. The former uses probability 0.5, meaning that one in two inputs will be randomly excluded from each update cycle. The latter is applied to the weights with a penalty factor of 0.001. Cross-entropy loss is used as loss function and gradient descent with Nesterov momentum set to 0.9 is used to perform optimization.

The architecture of the proposed CNN model is represented in Fig. 3.

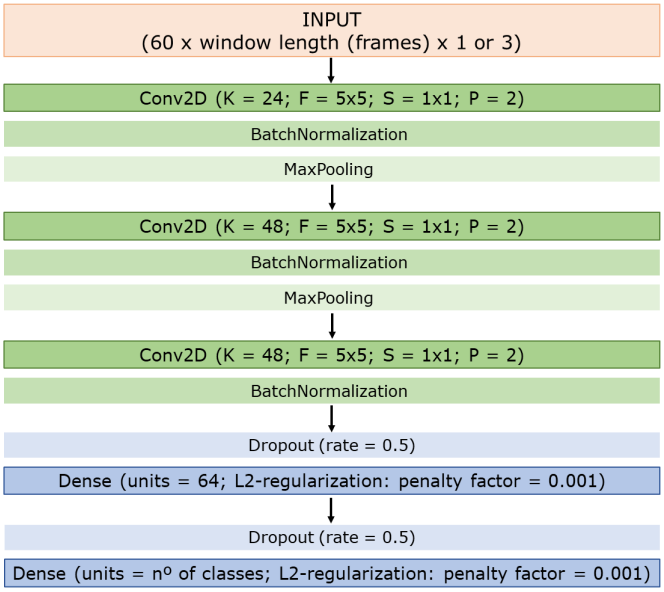

Fig. 3. Architecture of the CNN

Therefore, the CNN receives windows with the mel spectrogram representation, or the mel spectrogram with the addition of the first and second derivatives as input, and it assigns class scores to each window. The class with the highest score is the class predicted for the specific window. In this way, the classifier makes predictions for each individual window instead of doing a classification for the signal as a whole. CNN is built using Keras library with TensorFlow.

D. Evaluation

Taking into account the several indicators for multi-class classification reviewed in [23], four metrics are used to evaluate the performance of the classifiers:

$$Accuracy = \frac{Total\ number\ of\ correct\ classifications}{Total\ number\ of\ windows}. \quad (4)$$

$$Precision_{class} = \frac{True\ positives\ (class)}{Total\ number\ of\ predicted\ positives\ (class)}. \quad (5)$$

$$Recall_{class} = \frac{True\ positives\ (class)}{Total\ number\ of\ actual\ positives\ (class)}. \quad (6)$$

$$F1\text{-}score_{class} = 2 \cdot \left(\frac{Precision_{class} \cdot Recall_{class}}{Precision_{class} + Recall_{class}} \right). \quad (7)$$

The accuracy corresponds to the percentage of correct classification for the entire set of testing data. Precision and recall for each class are computed considering true positives as the only correctly classified windows for the class under study. Precision expresses the probability of a prediction being correct for that specific class. Recall measures the accuracy for the class under study. The F1-score is computed as the harmonic mean between precision and recall.

E. Testing the framework on urban sounds

In the marine environment, there are diverse types of sound sources, where the propagation of sound waves differs depending on the depth, temperature, pressure and salinity. So, a high number of samples for each class, covering several types of situations, is needed. However, these are difficult to obtain during this type of work. Thus, in order to have more confidence whenever the dataset size is extended, finding a way to test the recognition framework for a large dataset is necessary. To this end, an open-source dataset composed of urban sounds (UrbanSound8K dataset [15]) was used. This is constituted by sounds that can be heard in a city environment with a total of 8732 labeled recordings and each one labeled to one of ten distinguished classes: Air Conditioner, Car Horn, Children Playing, Dog Bark, Drilling, Enginge Idling, Gun Shot, Jackhammer, Siren and Street Music. The duration of each recording is up to 4.00 s.

The sampling rate of each recording is highly variable. The most common value is 44100 Hz. All the data was resampled to 22050 Hz, since this decreases the computational time needed and is in line with Nordby's work [16]. A different sampling rate compared to the default (52734 Hz) leads to a few adjustments in the parameters of the mel spectrogram. Thus, for the present dataset, the signals were framed using a window size of 1024 samples (46 ms) and a hop between frames of 512 samples (23 ms). The frequency range was set to 0 - 8000 Hz.

In total, five similar models were built where the difference was in the windowing process. Only the mel spectrogram (no derivatives) was used as feature extraction approach. To each model a *rectangular window function* of a different length, **0.75 s**, **1.42 s**, **2.21 s**, **3.00 s** and **4.00 s**, was applied, which, according to Equation 2, corresponds to **31 frames**, **60 frames**, **94 frames**, **128 frames** and **172 frames**, respectively. For recordings that have a duration shorter than the rectangular window length, the mel spectrogram was extended applying zero-padding until the rectangular window length was reached so as to insure at least one window for each recording. Each model was trained for up to 50 epochs and with a batch size of 200.

The CNN had to be adapted taking into account the number of frames of the input, more precisely the two max-pooling layers. The dimensions of its filter size and stride are described in Table I.

TABLE I
MAX-POOLING LAYER PARAMETERS (URBANSOUND8K DATASET)

Input dimension	Filter size	Stride
(60 x 31 x 1)	3 x 2	3 x 2
(60 x 60 x 1)	3 x 3	3 x 3
(60 x 94 x 1)	3 x 3	3 x 3
(60 x 128 x 1)	3 x 4	3 x 4
(60 x 172 x 1)	3 x 5	3 x 5

All models showed a stable performance with a very slight improvement when increasing the window length. The average accuracy of each model with the increase in the window length was 69.3%, 70.4%, 70.6%, 71.1% and 72.2%, respectively. These results can be compared to [16], which proposed a CNN model using mel spectrogram windows of 0.72 s as input, and to [17], which proposed the same approach but with windows of 3.00 s. The best performance of the two studies without data augmentation was 72.3% and 73%, respectively, which are slightly higher than the accuracy of the present work. Nevertheless, it is possible to conclude that the performance of the present models are actually very close to the expected range. This gives confidence regarding the devised recognition framework.

III. APPLICATION OF THE RECOGNITION FRAMEWORK IN SOUNDS PRODUCED BY VESSELS

ShipsEar is a dataset composed of vessel sound recordings (available at http://atlanttic.uvigo.es/underwaternoise/), and it was introduced by Santos-Domínguez et al. [19]. The same authors also developed a vessel classifier for the same dataset based on the cepstral coefficients plus its first and second derivatives and the Gaussian mixture model. Since 2016, further studies with different approaches have been published using the ShipsEar dataset. Two studies published in 2020 (Li et al. [24]; Ke et al. [25]) are noteworthy. Li et al. [24] proposed a feature extraction based on the filter banks and the mel frequency cepstral coefficients, using as classifier a deep neural network. In addition, after extracting features from the acoustic signal, there was an optimization process based on the triplet loss concept in order to increase the inter-class distance and reduce the intra-class distance. On the other hand, Ke et al. [25] paid more attention to the feature

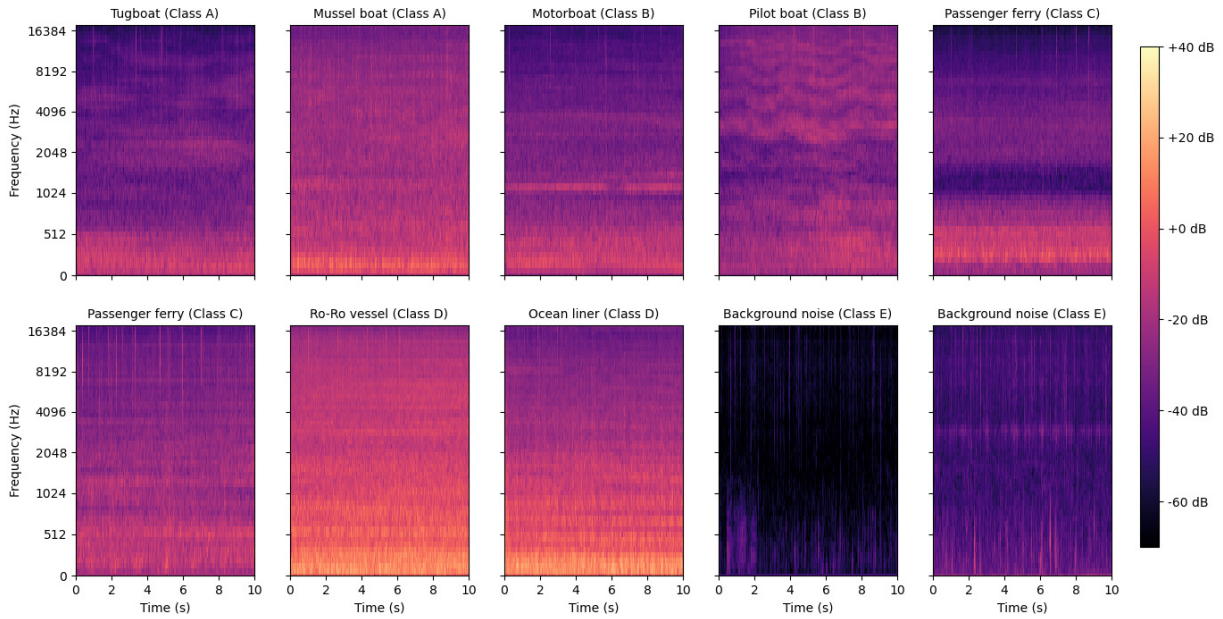

Fig. 4. Mel spectrogram representations of each class from the ShipsEar dataset

extraction, computing different kinds of features: temporal, statistical, spectral, cepstral, Hilbert spectral, wavelet and deep neural network features. The k-nearest neighbour was used as classifier method.

A. Dataset description

The ShipsEar dataset is made up of 90 recordings, including 11 vessel types and different background sea noises with the presence of atmospheric elements such as wind, rain, waves or currents. The recordings were collected near the port of Vigo and in La Coruña outer harbour. The 11 vessel types plus the natural noises were divided into five classes based on [19]. Four classes were allocated to vessel sounds and one to natural sea noises, as shown in Table II. Vessels were classified according to size. The sampling rate corresponds to 52734 Hz.

TABLE II
CLASSES OF SHIPSEAR DATASET

Classes	Recording sources
A	fishing boats, trawlers, mussel boats, tugboats, dredgers
B	motorboats, pilot boats, sailboats
C	passenger ferries
D	ocean liners, ro-ro vessels
E	background noise recordings

Santos-Dominguez et al. [19] explained that "*The recordings were segmented with wide margins to preserve information from the beginning to the end of the event*". However, using these margins would have had negative effects on the current methodology. The developed classifier runs on predictions per windows along the signal, as opposed to having a single classification for the whole signal. In order to be absolutely certain that all windows are labeled correctly, those margins were excluded, working only with the relevant intervals, *i.e*, those containing vessel sounds. Therefore, an interval for each recording, discarding these margins, was defined. The intervals also took into account the difference in the data collected. For example, smaller intervals for recordings of class C were chosen, which had the greatest amount of recordings, to avoid large differences in the class distribution. In addition, five recordings were excluded that were considered as outliers, either because the vessels were not moving (three from class C and one from class D) or because the vessel signal occurred for too short a time and with a low amplitude (one from class B). The final total of recordings was 85. The initial and final class distribution is described in Table III

TABLE III
CLASS DISTRIBUTION OF SHIPSEAR DATASET

	Initial class distribution		**Final class distribution**	
	Recordings (nº)	Time (s)	Recordings (nº)	Time (s)
Class A	17	1881	17	1216
Class B	19	1567	18	1198
Class C	30	4278	27	1204
Class D	12	2460	11	996
Class E	12	1146	12	1146

In Figure 4, the mel spectrogram of two examples of each class is represented with an interval of 10 s. Finding patterns that distinguish different types of vessels according to their size is not easy. However, there is a clear difference between the vessel sounds and the background noise, since the former consists in horizontal lines near the bottom of the mel spectrogram and the latter is not represented by any particular frequency.

B. Parametrization

Two models were built using the mel spectrogram and the mel spectrogram and its first and second derivatives as features. The window length was set approximately to **1.00**

TABLE IV
CLASSIFIER PRECISION, RECALL AND F1-SCORE OF THE TWO FEATURE EXTRACTIONS

Metric	Features	Class A	B	C	D	E
Precision (%)	Mel spectrogram (MS)	78.7	83.1	79.2	77.5	97.5
	MS + 1st and 2nd MS derivatives	81.8	91.4	85.4	87.1	98.6
Recall (%)	Mel spectrogram (MS)	74.4	82.9	75.4	87.1	97.8
	MS + 1st and 2nd MS derivatives	85.3	87.3	81.6	91.2	99.5
F1-score (%)	Mel spectrogram (MS)	76.5	83.0	77.2	82.0	97.7
	MS + 1st and 2nd MS derivatives	83.5	89.3	83.4	89.1	99.1

s, which according to Equation 2 corresponds to **50 frames**. Hence, the input dimension of the CNN was (60 x 50 x 1) or, if the mel spectrogram is coupled with its first and second derivatives, (60 x 50 x 3). Whichever the input size, the two max pooling layers of the CNN were defined with a filter size of 3x3 and stride 3. The frequency range was set between 0 Hz and 8000 Hz, because most of the relevant frequencies are below 8000 Hz.

C. Data splitting

Each model was built and evaluated using a 10-fold cross-validation [26]. The same process was also used in [25]. In data splitting, there was random sampling of windows for training and testing data. In each split, 10% of training data was used as validation data in order to identify the training epoch that yields the best model parameters based on validation accuracy. Each model was trained for up to 50 epochs and with a batch size of 200.

D. Results

In Figure 5, an overview of the performance for the two models with different feature extraction is shown as a box-and-whisker plot of the ten estimates of the accuracy generated from the cross-validation. The cross and horizontal line represent the average and median accuracy, respectively. The feature combination between the mel spectrogram and the first and second derivatives performed better than when only using the mel spectrogram. The average accuracy of the two models was 88.8% and 83.2%, respectively.

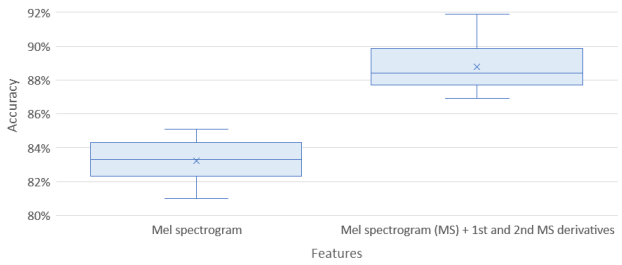

Fig. 5. Test accuracy of the two models with different feature extraction

The precision, recall and F1-score metrics are used to evaluate the performance for each class individually (Table IV). The F1-score confirms that the feature combination between the mel spectrogram and its derivatives performed better than solely using the mel spectrogram as features. The F1-score shows an improvement for all classes, especially for vessel classes, where this metric increased between 6.2% and 7.1%. The precision and recall of class E indicates that there is a reliable identification of the presence and absence of vessel noise for both models. The precision shows that most of the samples classified as class E were correct, and the recall indicates that the majority of class E data was predicted accurately. Any misclassifications in the four vessel classes were, thus, for the most part, between these four classes.

IV. APPLICATION OF THE RECOGNITION FRAMEWORK IN SOUNDS PRODUCED BY MARINE ANIMALS AND VESSELS

In order to study the capacity of the recognition framework in a wider scenario, two marine animals were integrated in the ShipsEar dataset: dolphins and humpback whales.

A. Dataset description

In total, five classes were used: **dolphins**, **humpback whales**, **small vessels**, **large vessels** and **background noise**. The dolphin class covers recordings of three different species: common dolphin, striped dolphin and bottlenose dolphin. The marine animal data was collected from open sources, which are acknowledged in the Appendix of this dissertation. In addition, two recordings of common dolphin sounds provided by WavEC were added to the dolphin data.

In the previous application of the current methodology, the vessel recordings were distributed into four classes according to vessel size. Now, the same criteria was followed. However, the four classes were merged into just two. Class A and B were merged into **small vessels** class and class C and D into **large vessels** class. Since the data sample of marine animals was very small, parts of the ShipsEar recordings have to be selected in order to avoid large differences in the class distribution. Thus, from the 85 recordings used in the application of the recognition framework for vessel sounds, 11 recordings from each class were selected with an interval of approximately 20 s each. Regarding the background noise, it was used all the background noise recordings (12) with an interval of approximately 40 s each. In Table V, the data is summarized with information about the duration and the sampling rate of each recording. The sampling rate is quite variable. Thus, it was decided to set it to 52734 Hz, in order to keep the ShipsEar dataset sampling rate. In a few cases of dolphin and humpback

TABLE V
CLASS DISTRIBUTION

Dolphins	Humpback Whales	Small vessels	Large vessels	Background noise
Dolphin1.wav (7.6s;44100Hz) Dolphin2.wav (6.9s;44100Hz) Dolphin3.wav (9.4s;44100Hz) Dolphin4.wav (7.9s;22050Hz) Dolphin5.wav (49.7s;44100Hz) DolphinWavEC1.wav (37.3s;192000Hz) DolphinWavEC2.wav (36s;192000Hz)	Whale1.wav (11s;22050Hz) Whale2.wav (10.1s;22050Hz) Whale3.wav (10.6s;22050Hz) Whale4.wav (12.1s;22050Hz) Whale5.wav (17s;44100Hz) Whale6.wav (26.2s;10000Hz) Whale7.wav (10.6s;8000Hz) Whale8.wav (29.5s;44100Hz)	22 intervals of $\approx$ 20 s extracted from Class A and B of ShipsEar dataset (52734Hz)	22 intervals of $\approx$ 20 s extracted from Class C and D of ShipsEar dataset (52734Hz)	12 intervals of $\approx$ 40 s extracted from Class E of ShipsEar dataset (52734Hz)
154.8 s (total)	127.1 s (total)	440 s (total)	439 s (total)	476 s (total)

whale data, the duration of the recordings is lower than that available on the original recordings, since intervals without vocalizations were excluded.

In Figure 6, the mel spectrogram of some dolphin and humpback whale vocalizations is represented. The two most common dolphin vocalizations are the whistles and the echolocation clicks. Whistles give rise to high frequency contours in the mel spectrogram (for example Figure 6 (a)), while the echolocation clicks create vertical line patterns. In Figure 6 (e), there is a mix of whistles and clicks. The humpback whale vocalizations are identified as arcs, which sometimes are close to horizontal bars, at low frequencies in the mel spectrogram.

B. Data augmentation

There is a significant difference in the amount of data between the two marine animal classes and the other three classes. In order to balance data, three common data augmentation techniques based on [27], [28] were applied to dolphin and humpback whale data, which enlarge the dataset artificially.

- Time stretching: consists in speeding up and slowing down the signal without changing its pitch. Each recording of the dolphin and humpback whale classes was stretched by two factors: $\{0.8, 1.2\}$. If the factor is lower than 1, the signal is slowed down. If the factor is higher than 1, the signal is speeded up.
- Pitch shifting: consists in raising and lowering the pitch of the audio signal without changing its duration. Each recording of the dolphin and humpback whale classes was pitch shifted by two values in semitones: {-2, -4}.
- Time shifting: each recording of the dolphin and humpback whale classes is shifted over time, changing the original order of the signal. This is conducted by moving the last (or first, randomly selected) interval of each recording with a duration of $1/2$ and $3/4$ of the window length in seconds to the beginning (ending). In this way, the marine animal vocalizations will be placed in different positions in the mel spectrogram in contrast to the original signal.

Time stretching and pitch shifting were applied using the module *effects* of the *Librosa* library [21].

C. Parametrization

Only the feature combination of the mel spectrogram and the first and second derivatives was used here. Five models were developed with a difference in the windowing process. To each model a *rectangular window function* of a different length, **0.22 s**, **0.61 s**, **1.00 s**, **1.48 s** and **1.97 s** was applied, which, taking into account Equation 2, corresponds to **10 frames**, **30 frames**, **50 frames**, **75 frames** and **100 frames**. The window size did not go beyond 2.00 s, in order to make a more detailed analysis of smaller windows. This study was important since the objective is to develop a model that may have a future use as real-time classification. Different dimensions in the windowing process requires an adaptation of the two max-pooling layers in CNN with respect to filter size and stride as described in Table VI. The frequency ranged was set to 0 - 18000 Hz, since dolphin vocalizations appear in the high frequencies of the mel spectrogram.

TABLE VI
MAX-POOLING LAYER PARAMETERS

Input dimension	Filter size	Stride
(60 x 10 x 3)	3 x 1	3 x 1
(60 x 30 x 3)	3 x 2	3 x 2
(60 x 50 x 3)	3 x 3	3 x 3
(60 x 75 x 3)	3 x 3	3 x 3
(60 x 100 x 3)	3 x 4	3 x 4

D. Data splitting

A 3-fold cross validation [29] at the level of the dolphin and humpback whale classes was used to test the classifiers. All recordings of these classes were divided into three subsets. This division was different to those used to the remaining classes (2 classes of vessels and the background noise), since the number of recordings of the other classes is much larger and it is typically higher in relation to the time duration. Hence, in order to have a balanced testing data, only two recordings of the small vessels, large vessels and background noise classes were added to each one of the three subsets. Therefore, the evaluation was based on using one of the three subsets as testing data, with the remaining data being used for training (including the remaining recordings that were not added to the three subsets). The same process was repeated with each of the other two subsets as testing data.

Each model was trained for up to 50 epochs and with a batch size of 200. The model parameters of epoch 50 were the ones selected to be evaluated in the testing data, since training loss

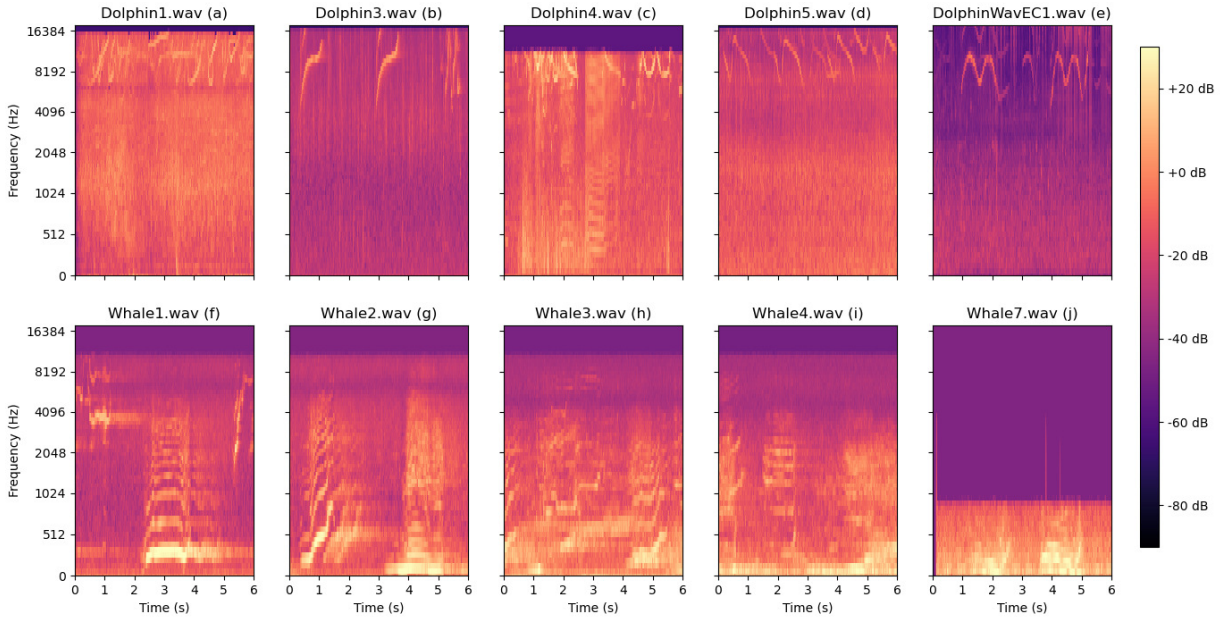

Fig. 6. Mel spectrogram representations of dolphin and humpback whale vocalizations

and accuracy suffered minor changes after epoch 40. There was a little uncertainty in the performance of the model in each evaluation of the three testing subsets due to the small size of the training data set. This led to some differences in the performance estimation from one run of the three testing subsets to another one. In order to reduce this uncertainty and to have a more reliable estimation, the model performance was evaluated using five repeats of the three testing subsets [26], generating 15 different estimates of the accuracy.

E. Data augmentation results

Before analysing the performance achieved by the five models with different window length, the impact of the three data augmentation techniques (time stretching, pitch shifting and time shifting) was studied. In this way, using the current recognition framework, the three techniques were applied individually and simultaneously to training data of dolphin and humpback whale classes. Thus, four models were built. An additional model was also assembled without data augmentation. Augmentation techniques were only applied to training data, in order to avoid biased results in testing data. The five models were constructed using the same window length, **1.00 s** (50 frames).

The results of the five models are reported in Figure 7 as a box-and-whisker plot generated from the 15 different estimates of the accuracy for each model. The cross and horizontal line represent the average and median accuracy, respectively. The three augmentation techniques applied individually achieved an average accuracy equal to 73.7%, 76.3%, 75.5%, corresponding to the application of time stretching, pitch shifting and time shifting, respectively. This corresponds to an improvement in comparison with the model without data augmentation, which achieved an average accuracy of 71.6%. The highest improvement was achieved by using the three augmentation techniques simultaneously, with a corresponding average accuracy of 76.5%.

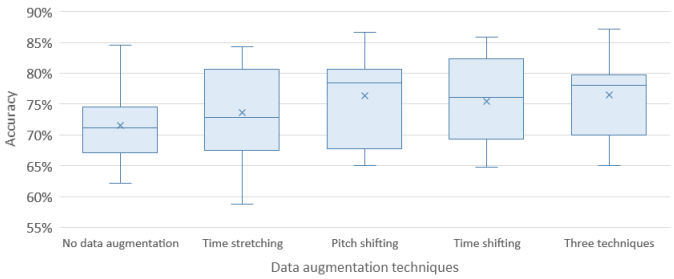

Fig. 7. Repeated accuracy results using different data augmentation techniques

In the following study about the impact of the window length, all models were constructed by applying the three augmentation techniques simultaneously to training data of dolphin and humpback whale classes.

F. Results of window length variation

Table VII shows the average accuracy and recall results for each model. The former is the average of the 15 different estimates of the accuracy and the latter corresponds to the average recall of each class among the five repeats. The best performances were reached by the two models with intermediate windows (1.00 s or 1.48 s), which achieved an average accuracy of 76.5% and 78.3%, respectively.

In the case of the dolphin class, there was a similar performance for the five models, where the recall value ranged between 60.0% and 70.5%. On the other hand, there was a clear enhancement in the classification of humpback whale vocalizations, which improved from 56.3% at a window duration 0.22 s to 100% at 1.97 s. In the case of the vessel noise, a better performance was achieved by the small vessel class.

TABLE VII
CLASSIFIER RECALL AND AVERAGE ACCURACY

	Recall (%)					Average Accuracy (%)
	Dolphins	Humpback Whales	Small Vessels	Large Vessels	Background Noise	
Window length: 0.22 s (10 frames)	70.5	56.3	60.6	60.6	76.1	66.2
Window length: 0.61 s (30 frames)	60.0	64.8	66.2	66.1	80.6	69.4
Window length: 1.00 s (50 frames)	64.1	90.1	80.3	64.0	80.8	76.5
Window length: 1.48 s (75 frames)	65.0	94.5	83.6	64.1	81.7	78.3
Window length: 1.97 s (100 frames)	68.2	100	83.0	56.3	62.5	72.1

Considering the small and large vessel classes as a single one (Table VIII), it may be concluded that there was a reliable detection of the vessel noise presence, with a vessel recall above 93.7%. This shows that most of the wrong classifications of small and large vessel classes were due to small vessels being misclassified as large vessels and vice-versa.

TABLE VIII
CLASSIFIER RECALL OF THE TWO VESSEL CLASSES

	Vessel recall (%)
window length: 0.22 s (10 frames)	93.7
window length: 0.61 s (30 frames)	96.4
window length: 1.00 s (50 frames)	97.2
window length: 1.48 s (75 frames)	97.6
window length: 1.97 s (100 frames)	95.2

All models, except the one with a window duration of 1.97 s, correctly identified the absence of human activity or marine animals in most of the windows, where recall value for the background noise ranged between 76.1% and 81.7%.

V. CONCLUSIONS

This work consisted in developing a model capable of recognizing ocean acoustic sources present in hydroacoustic "raw" data. The task was conducted as a machine learning (ML) problem, where a series of hydroacoustic signals were used to find common patterns in the data. The methodology behind the model consisted in converting the acoustic data into more representative features by applying signal processing techniques. These features were classified by a state-of-the-art ML: CNN.

Two different feature extractions were studied, based on previous works in the field of ASR. Both involved converting the acoustic signal to a mel spectrogram representation. However, an extra step was added to one of the feature extraction processes, which consisted in computing the first and second derivatives of the mel spectrogram.

The designed recognition framework was applied to three different datasets. The first corresponds to the UrbanSound8k dataset, which was used solely as a testing of the recognition framework. Five different models were constructed with different window lengths. The accuracy of the five models ranged between 69.3% and 72.2%, which are very close to the level of other works using a similar methodology.

ShipsEar was the second dataset used in this work, which consists of underwater acoustic data of various types of vessels and background noise. The model using the feature combination between the mel spectrogram and the first and second derivatives performed better than only using the mel spectrogram as features, with a corresponding average accuracy of 88.8% and 83.2%, respectively. In addition, the results showed a good performance for both models in the perception of the presence and absence of vessel noise.

The last dataset aimed to spread the spectrum of marine sound sources. Thus, the data from the ShipsEar dataset was complemented with a set of marine animal (dolphin and humpback whale) recordings. Three data augmentation techniques were studied. These were applied individually and simultaneously to training data of dolphin and humpback whale classes. The model with the three techniques applied simultaneously outperformed the model without data augmentation, with a corresponding accuracy rate of 76.5%. The influence of the window length in the classification accuracy was analysed. The results indicated that the best two models used a window length of 1.00 s and 1.48 s, which achieved an accuracy rate of 76.5% and 78.3%, respectively. In this way, the devised methodology achieved the intended objective, which indicated that it could be a reliable tool in a future real-time application, since the two best models make predictions for windows with less than 1.50 s.

This work may be a springboard for further and more detailed research in this area. Acquired more data for all classes is important to study the capacities of the recognition framework in more detail. A review of the labeling process could be made by individualizing aspects that characterize each class. In order to avoid some classification mistakes for the dolphin data, partitioning this class based on the types of vocalizations (whistles and echolocation clicks) would be an improvement. The same should take place with the background noise class by distinguishing the recordings in which the presence of atmosphere elements (such as rain or wind) is more evident. Moreover, for vessels, performing an analysis based on the type of vessel rather than its size might be more suitable.

This model was not designed to distinguish overlapping sounds, suggesting this could be an interesting area to explore in order to devise a model which is better prepared to address the different scenarios which may occur in a marine environment.

ACKNOWLEDGMENT

I would like to give my thanks to Professor João Sousa for his help and guidance. I am equally grateful to Dr. Eng. Guilherme Vaz and Eng. Miguel Vicente from the WavEC company for their unconditional whole collaboration and support in this work. In addition, I would like to thank to Msc. Érica Cruz for the insight provided in field of marine bioacoustic.

REFERENCES

[1] Y. Xu, Y. Zhang, H. Wang, and X. Liu, "Underwater image classification using deep convolutional neural networks and data augmentation," *2017 IEEE International Conference on Signal Processing, Communications and Computing, ICSPCC 2017*, vol. 2017-January, pp. 1–5, 2017.

[2] S. Marini, E. Fanelli, V. Sbragaglia, E. Azzurro, J. Del Rio Fernandez, and J. Aguzzi, "Tracking fish abundance by underwater image recognition," *Scientific Reports*, vol. 8, no. 1, pp. 1–12, 2018.

[3] F. Han, J. Yao, H. Zhu, and C. Wang, "Underwater image processing and object detection based on deep cnn method," *Journal of Sensors*, vol. 2020, 2020.

[4] N. Wawrzyniak, T. Hyla, and A. Popik, "Vessel detection and tracking method based on video surveillance," *Sensors (Switzerland)*, vol. 19, no. 23, 2019.

[5] I. Jeon, S. Ham, J. Cheon, A. M. Klimkowska, H. Kim, K. Choi, and I. Lee, "A real-time drone mapping platform for marine surveillance," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, vol. 42, no. 2/W13, pp. 385–391, 2019.

[6] J. Huang, J. Zhao, and Y. Xie, "Source classification using pole method of ar model," in *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 1. IEEE, 1997, pp. 567–570.

[7] R. C. Bennett *et al.*, "Classification of underwater signals using a back-propagation neural network," Ph.D. dissertation, Naval Postgraduate School, 1997.

[8] A. Kundu, G. C. Chen, and C. E. Persons, "Transient sonar signal classification using hidden markov models and neural nets," *IEEE Journal of Oceanic Engineering*, vol. 19, no. 1, pp. 87–99, 1994.

[9] B. Howell and S. Wood, "Passive sonar recognition and analysis using hybrid neural networks," in *Oceans 2003. Celebrating the Past... Teaming Toward the Future (IEEE Cat. No. 03CH37492)*, vol. 4. IEEE, 2003, pp. 1917–1924.

[10] C. Kang, X. Zhang, A. Zhang, and H. Lin, "Underwater acoustic targets classification using welch spectrum estimation and neural networks," in *International Symposium on Neural Networks*. Springer, 2004, pp. 930–935.

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[12] H. A. Garcia, T. Couture, A. Galor, J. M. Topple, W. Huang, D. Tiwari, and P. Ratilal, "Comparing performances of five distinct automatic classifiers for fin whale vocalizations in beamformed spectrograms of coherent hydrophone array," *Remote Sensing*, vol. 12, no. 2, pp. 1–25, 2020.

[13] M. Harvey *et al.*, "Acoustic detection of humpback whales using a convolutional neural network," *Google AI Blog*, 2018.

[14] E. H. Belghith, F. Rioult, and M. Bouzidi, "Acoustic diversity classifier for automated marine big data analysis," *Proceedings - International Conference on Tools with Artificial Intelligence, ICTAI*, vol. 2018-Novem, pp. 130–136, 2018.

[15] J. Salamon, C. Jacoby, and J. P. Bello, "A dataset and taxonomy for urban sound research," *MM 2014 - Proceedings of the 2014 ACM Conference on Multimedia*, no. 1, pp. 1041–1044, 2014.

[16] J. Nordby, "Environmental sound classification on microcontrollers using convolutional neural networks," Master's thesis, Norwegian University of Life Sciences, 5 2019. [Online]. Available: http://hdl.handle.net/11250/2611624

[17] J. Salamon and J. P. Bello, "Deep convolutional neural networks and data augmentation for environmental sound classification," *IEEE Signal Processing Letters*, vol. 24, no. 3, pp. 279–283, 2017.

[18] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery in databases," *AI magazine*, vol. 17, no. 3, pp. 37–37, 1996.

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

[20] I. Guyon, S. Gunn, M. Nikravesh, and L. A. Zadeh, *Feature extraction: foundations and applications*. Springer, 2008, vol. 207.

[21] B. McFee, V. Lostanlen, A. Metsai, M. McVicar, S. Balke, C. Thomé, C. Raffel, F. Zalkow, A. Malek, Dana, K. Lee, O. Nieto, J. Mason, D. Ellis, E. Battenberg, S. Seyfarth, R. Yamamoto, K. Choi, viktorandreevichmorozov, J. Moore, R. Bittner, S. Hidaka, Z. Wei, nullmightybofo, D. Hereñú, F.-R. Stöter, P. Friesch, A. Weiss, M. Vollrath, and T. Kim, "librosa/librosa: 0.8.0," July 2020. [Online]. Available: https://doi.org/10.5281/zenodo.3955228

[22] P. Peso Parada and A. Cardenal-López, "Using gaussian mixture models to detect and classify dolphin whistles and pulses," *The Journal of the Acoustical Society of America*, vol. 135, no. 6, pp. 3371–3380, 2014.

[23] M. Grandini, E. Bagli, and G. Visani, "Metrics for multi-class classification: An overview," *arXiv*, pp. 1–17, 2020.

[24] C. Li, Z. Liu, J. Ren, W. Wang, and J. Xu, "A feature optimization approach based on inter-class and intra-class distance for ship type classification," *Sensors (Switzerland)*, vol. 20, no. 18, pp. 1–12, 2020.

[25] X. Ke, F. Yuan, and E. Cheng, "Integrated optimization of underwater acoustic ship-radiated noise recognition based on two-dimensional feature fusion," *Applied Acoustics*, vol. 159, p. 107057, 2020. [Online]. Available: https://doi.org/10.1016/j.apacoust.2019.107057

[26] M. Kuhn and K. Johnson, *Applied predictive modeling*, 2013, vol. 26. [Online]. Available: http://appliedpredictivemodeling.com/s/Applied{_}Predictive{_}Modeling{_}in{_}R.pdf

[27] G. Maguolo, M. Paci, L. Nanni, and L. Bonan, "Audiogmenter: A matlab toolbox for audio data augmentation," *arXiv*, 2019.

[28] S. Wei, S. Zou, F. Liao, and W. Lang, "A comparison on data augmentation methods based on deep learning for audio classification," *Journal of Physics: Conference Series*, vol. 1453, no. 1, 2020.

[29] D. Berrar, "Cross-validation," *Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics*, vol. 1-3, no. April, pp. 542–545, 2018.