DATA AUGMENTATION AND PREPROCESSING TECHNIQUES FOR ENHANCED UNDERWATER DETECTION AND CLASSIFICATION

KT Hjelmervik	University of South-Eastern Norway (USN), Horten, Norway
César Ortiz-Toro	Universidad Politécnica de Madrid (UPM), Madrid, Spain
Alberto Belmonte-Hernández	Universidad Politécnica de Madrid (UPM), Madrid, Spain
Anaida Fernández García	Universidad Politécnica de Madrid (UPM), Madrid, Spain
Alvaro Gutiérrez	Universidad Politécnica de Madrid (UPM), Madrid, Spain

1 INTRODUCTION

The opaqueness of the ocean provides opportunities for illicit trade enterprises, acts of terrorism, and covert military operations. Compact submarines or divers can infiltrate crucial infrastructure, such as harbors, to either cause destruction or transport illegal goods within the secure expanse of the ocean[1,2,3]. The EU Horizon project, Smaug, develops an AI backed underwater surveillance system for harbours to counter these threats[4] using acoustic sensors. The soundscape in a harbour environment is dominated by man-made noise that mask the acoustic signal from desired targets. Additionally, the geometry of a harbour due to quays and breakwaters, offers a complex environment that strongly limits the acoustic propagation. Short sensor ranges are therefore expected, emphasizing the importance of either mobile sensors, or a large number of sensors for reliable monitoring of the underwater domain. Resulting in a sensor-intensive operation. Conventionally, acoustic sensors are monitored by highly trained sonar operators at a one-to-one basis. This is not economically viable for a harbour security system, so automatic solutions that detect, classify, and flag suspicious activity is required. The introduction of AI solutions in sonar operation[5] allows for this level of reduction in human interaction.

Here we explore the use of both conventional methods and deep learning (DL) algorithms for detecting the presence of targets close to the sensor. The deep learning algorithm leverages convolutional neural networks (CNNs) which are particularly effective in recognizing patterns and anomalies in complex sensory data like sonar signals. By automatically extracting features from raw data, CNNs can learn to identify subtle acoustic signatures of different types of objects, distinguishing between benign and potentially threatening targets with high accuracy[6,7]. Furthermore Deep Neural Networks (DNNs) can improve classification results using several extracted features from the acoustic signal[8]. The ShipsEAR dataset[9] (available at http://atlanttic.uvigo.es/underwaternoise/) is used to demonstrate the overall method. An augmentation technique that mix hydrophone recordings of present ships with ambient noise data to generate a rich and balanced data set for the training. In the augmentation datasets ranging from fairly clean (target and noise only) to complex (target, noise, and interfering vessel nearby) are generated, and the performance of the proposed algorithms are assessed for each dataset.

2 DATASET

The ShipsEar dataset[9] captures ship-generated noise from the Spanish Atlantic coast in Northwest Spain, specifically in the port of Vigo area, during autumn 2012 and summer 2013. The samples are collected at

a 52,734 Hz rate using digitalHyd SR-1 recorders. The dataset features 90 recordings with hydrophones configured in a vertical arrangement. The dataset encompasses eleven ship categories, ranging from fishing boats to ocean liners, along with coastal background noise. Here the *target* class consists of the recordings of motor-, pilot-, and sailboats, while the remaining ship classes are used as interfering vessels. The recordings were split into three different groups before data augmentation. The groups were made as balanced as possible in terms of recording minutes per class. The groups are used as test, training, and validation datasets in the machine learning. By splitting the datasets before the augmentation, we make sure that no single recording is both used as test and training data.

3 METHOD

3.1 Data augmentation

The data augmentation scheme mixes recordings of both target vessels and undesired, interfering vessels as well as ambient noise. The intention is both to increase the amount of recordings, but also to increase the complexity of the soundscape. K mixed recordings, $s[n]$, of length N are generated from random segments from three different random recordings; the *target* class, $s_t[n]$, the *noise* class, $s_n[n]$, and the an interfering vessel class, $s_{iv}[n]$, which represents an unwanted, interfering target in the vicinity of the target. Each segment starts at a random sample from the selected recording.

The received signal from the two vessels are modified to account for distance to the sensor. The distances are randomly selected and are assumed constant during the 3s recording. Both thermal absorption and geometrical loss are taken into account. The original recordings were typically made at 50 to 100 m range[9]. Since thermal absorption is both range and frequency dependent[10], the modifications are made in the frequency domain. The final mixed recording then becomes:

$$\widehat{s}[n] \quad = \quad s_n[n] + \widehat{s}_t[n] + \widehat{s}_{iv}[n], \tag{1}$$

where $\widehat{s}[n]$ is the augmented data, $s_n[n]$ is a random noise segment, and $\widehat{s}_t[n]$ and $\widehat{s}_{iv}[n]$ are the range-compensated versions of random target and interfering vessel segments, respectively. In order to assess the false alarm rate of the detection algorithms and to provide background noise for the conventional detectors, we generate additional two mixed segments, $s_{nt,j}[n]$, per target segment $\widehat{s}_t$. These extra segments represent recordings with no target present:

$$\widehat{s}_{nt,j}[n] \quad = \quad s_{n,j}[n] + \widehat{s}_{iv,j}[n]. \tag{2}$$

Both the noise segments $s_{n,j}[n]$ and the segments of the interfering target $s_{iv,j}$ are taken from the same files as their counterparts, $s_n[n]$ and s_{iv}, but the randomisation ensures that there are no overlaps neither between $s_{n,j}[n]$ and their counterparts, nor between any of the J different realizations of $s_{nt,j}[n]$.

3.2 Conventional detection

As a baseline comparison to the deep learning approach, we apply binary detection using Maximum Likelihood Estimation (MLE) and logistic regression, a form of the logit model. By formulating the probability of signal presence as a function of observed signal-to-noise ratios, our approach estimates the model parameters through MLE, providing robust detection performance. Three different conventional detectors are applied on the data; broadband, narrowband and DEMON. The signal-to-noise ratio (SNR) output of each detection algorithm is input to a maximum likelihood estimator (MLE).

For both broadband and narrowband detection of a signal with an unknown shape and strength the noise normalised energy detector (NNED) is a natural choice[11]:

$$\textbf{Energy in the received signal}: \quad E = \sum_{n=0}^{N-1} |\widehat{s}[n]|^2$$
$$\textbf{Estimated noise variance}: \quad \widehat{\sigma}_w^2 = \frac{1}{N} \sum_{n=0}^{N-1} |\widehat{s}[n]|^2$$
$$\textbf{Test statistic}: \quad \Lambda = \frac{E}{\widehat{\sigma}_w^2}$$

The noise variance must be estimated from portions of the data not contaminated by the signal we attempt to detect. We use the first of the two generated segments with no target present, $s_{nt,1}[n]$. We employ the same strategy for detection based on DEMON processing, as formulated by Trevorrow[12].

We run both the target signal, $\widehat{s}_t[n]$, and the second segment with no target present $\widehat{s}_{nt,2}$ through the detectors for all K segments. By applying thresholds to the test statistics we can estimate the probability of detection from the target signals, and probability of false alarm from the segments with no target. The performance of each detector is assessed by estimating a receiver operating characteristics (ROC) curve.

3.3 Deep learning detection

Deep neural networks (DNNs) are particularly suitable at processing feature vectors that encapsulate essential characteristics of acoustic data, extracted through advanced signal processing techniques. Before feeding data into the network, preprocessing steps can include the extraction of linear and power spectrum, Mel spectrograms, as well as Mel-Frequency Cepstral Coefficients (MFCCs), which succinctly capture the textural nuances of sound in a format highly amenable to neural analysis.

In this work, a DNN with two hidden layers with 512 neurons and ReLu activation is used with a final one neuron layer and Binary Cross Entropy Loss function for the classification. We employ the ResNet architecture, introduced by He et al.[13], specifically adapted for underwater detection of sounds emitted by vessels. The efficacy of ResNet for this purpose lies in its robustness and efficiency in processing complex sound representations, which are fundamental in capturing the distinctive features of underwater sounds. Linear and Power Spectrum Spectrograms, Mel spectrograms and Mel-Frequency Cepstral Coefficients (MFCCs) have been extracted from the 3s audios and features presented in the previous section have been used to compare results. Some examples are shown in Figure 1. This data has been obtained as follows. For linear spectrogram:

- **Fourier Transform:** The first step is to slice the continuous signal into overlapping segments and apply a Fourier Transform (specifically, the Short-Time Fourier Transform, or STFT) to each segment:
$$STFT\{x(t)\}(m,\omega) = \sum_{n=-\infty}^{\infty} x(n)w(n-m)e^{-j\omega n}$$
where $x(t)$ is the signal, $w(n)$ is the window function, m is the time index, and ω is the frequency.
- **Spectrogram Calculation:** It is possible to use linear or power spectrogram. For the second it is calculated by taking the magnitude squared of the STFT and converting to dB logarithmic scale:
$$S(m,\omega) = |STFT\{x(t)\}(m,\omega)|^2$$

For Mel spectrograms the frequencies are converted to the mel scale, approximating the human auditory system's response more closely than the linear frequency scale.

- **Mel Scale Conversion:** Convert the frequencies to the mel scale where f is the frequency in Hz.:

$$\text{Mel}(f) = 2595 \log_{10}\left(1 + \frac{f}{700}\right)$$

- **Filter Banks:** Apply triangular filters, typically 20-40 filters, spaced evenly on the mel scale to the power spectrum obtained from the STFT. Sum the energy in each filter to get the mel spectrogram.

MFCCs provide a compact representation of the mel spectrogram, making them useful features for audio processing tasks.

- **Compute Mel Spectrogram:** As described above.
- **Discrete Cosine Transform (DCT):** Apply a DCT to the log of the mel spectra to decorrelate the filter bank coefficients and yield a compressed representation:

$$MFCC_k = \sum_{n=1}^{N} \log(S_n) \cos\left[k \frac{(2n-1)\pi}{2N}\right] \quad \text{for } k = 1, \ldots, K$$

where S_n are the mel spectral coefficients, and K is the number of cepstral coefficients to retain.

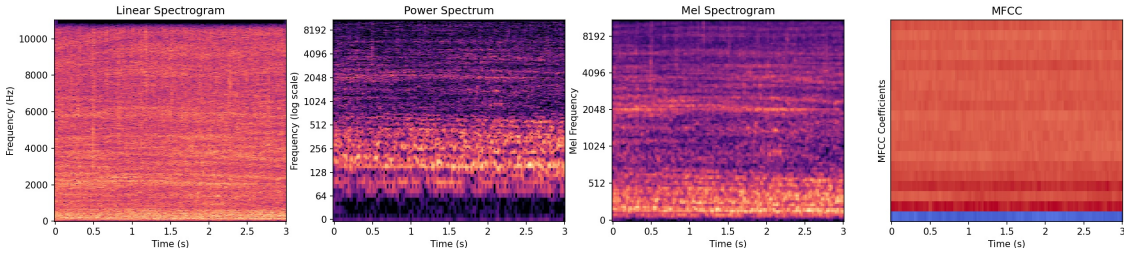

Figure 1: From left to right: Linear Spectrogram, Power Spectrum Spectrogram, Mel spectrogram, Mel-Frequency Cepstral Coefficients spectrograms. Example of data processing and image extraction from audio files to be the input of the neural networks of one audio from Class B.

4 RESULTS

When mixing data we randomly select an interfering vessel class and range-compensate both the target signal and the interfering vessel signal with uniformly distributed random ranges. 1000 segments are mixed for each of five different cases. Tab. 1 lists the cases and the minimum and maximum range of both the target and interfering vessel.

For the DNN we extracted features directly from the entire 3-second audio clip without using segmented windows, then, a simplified and compact set of features is generated. This method involves calculating a single feature vector that summarizes the acoustic properties of the entire audio clip. For this purpose, 13 Mel-frequency cepstral coefficients (MFCCs), 6 spectral contrast values, 12 chroma vectors, one spectral centroid value, one spectral roll-off value, and one zero-crossing rate are extracted, resulting in a total of 34 values per audio clip. This vector of 34 features represents a global view of the spectrum and tonal qualities of the clip, sacrificing temporal details for a more general and reduced description of the sound. The features utilized to train the ResNet50 prediction models are generated employing a STFT window length of 1024. In the case of linear spectrograms, the spacing between adjacent columns consists of

Table 1: Parameters and performances of the five cases

	Case 1	Case 2	Case 3	Case 4	Case 5
Parameters					
Target min. range, r_t	50 m	100 m	100 m	100 m	100 m
Target max. range, r_t	50 m	200 m	200 m	200 m	200 m
Interfering vessel min. range, r_t	N/A	N/A	700 m	400 m	200 m
Interfering vessel max. range, r_t	N/A	N/A	1000 m	700 m	400 m
Number of interfering vessels	0	0	1	1	1
Performances (AUC)					
Broadband detector	1	1	0.81	0.75	0.67
Narrowband detector	1	0.98	0.80	0.75	0.65
DEMON detector	0.84	0.79	0.75	0.72	0.65
MLE	1	0.98	0.90	0.88	0.81
ResNet50 (STFL)	0.93	0.95	0.91	0.93	0.65
ResNet50 (MEL)	0.94	0.99	0.94	0.81	0.75
ResNet50 (MFCC)	1	0.99	0.87	0.67	0.73
ResNet18 (Power Spectrum)	1	0.99	0.95	0.93	0.83
Deep Neural Network (34 features vector)	0.78	0.77	0.68	0.61	0.63

512 points and the resulting 2D feature matrix is reshaped to dimensions of 128x102. Mel spectrograms are estimated utilizing 128 Mel bandpass filters, yielding a 2D matrix of size 128x102. The MFCCs are computed using 40 MFCC coefficients, with the resulting 2D matrix resized to dimensions of 128x64. For the Power Spectrum Spectrogram, same parameters as with STFT were used but image was resized to 256x256 and ResNet18 was employed as classification network.

In Case 1 the data are unmodified. Segments from the unmodified original target and ambient noise recordings are used. In Case 2 the target segments are range-compensated, while the noise recordings are unmodified. In cases 3 to 5 the target segments are range-compensated and a range-compensated interfering vessel segment is added to both the target segment and the noise segment. Receiver operating characteristic curves and the distributions of the test statistics for each detector are shown in Fig. 2. The resulting ROC curve for the MLE and DL algorithms were also included. The MLE detector was fed with the test statistics of all three conventional detectors as well as the frequency of the strongest peak in the DEMON and narrowband processing. The DL detectors were fed the full segments as detailed in section 3.3. The machine learning algorithms were trained on a training dataset containing 1000 realizations of *targets* and *non-targets*. The ROC curves shown are determined from a test dataset containing 1000 independent realizations. The original recorded files were separated to ensure that segments of data from a single recording is not used in both the test and training dataset. Tab. 2 shows the AUC for each combination of training (rows) and test (columns) dataset for the MLE, ResNet50 using the linear spectrogram and ResNet18 with power spectrogram. Result of ROC curves are collected in Fig. 2

5 DISCUSSION

The broadband and narrowband detectors have comparable performance. For cases 1 and 2, the target is consistently detected. When an interfering vessel is introduced (cases 3 to 5) the performance quickly falls off for decreasing distances to the interfering target. This is partly because the interfering target

Table 2: Area under curve for the ROC curve with MLE detector, ResNet50/Linear and ResNet18/power spectrum for case 5.

	MLE			**ResNet50/Linear**			**ResNet18/Power Spectrum**		
	Group 1	Group 2	Group 3	Group 1	Group 2	Group 3	Group 1	Group 2	Group 3
Group 1	N/A	0.58	0.81	N/A	0.66	0.65	N/A	0.71	0.73
Group 2	0.79	N/A	0.71	0.91	N/A	0.89	0.94	N/A	0.9
Group 3	0.84	0.58	N/A	0.52	0.58	N/A	0.66	0.68	N/A

increases the background estimate used in the NNED. As can be seen by the general reduction of the test statistics for the *target* class. But also because the interfering vessel may raise the noise levels in the *non-target* class, as observed from the increased width of the *non-target* distributions. The performance of the DEMON detector is significantly worse than the broadband and narrowband detectors. However, its performance is less influenced by the introduction of the interfering target.

The MLE detector combines the information, including the frequencies of the frequency peaks in the DEMON and narrowband processing. The combination of several detectors significantly improves the performance for the cases with an interfering target. However, the MLE detector is sensitive to what groups are used for test and training, see Table 2. When group 2 is used as a test group, the performance of the MLE detector falls significantly. This is possibly related to the drop in performance of the DEMON detector (AUC slightly above 0.5). Closer examination shows that the DEMON signatures of the targets allocated to group 2 were less pronounced than the recordings allocated to groups 1 and 3. This is a consequence of the relatively low number of different recordings in each group.

Models trained using ResNet50, especially those utilizing a linear spectrogram as a feature, exhibit similar performance to MLE, at least in the first four cases. It should be noted that the performance of CNN models also depend on the characteristics of both the test and, in particular, training sets. As shown in Table 2, significant variations in performance are evident depending on the test and training set employed, especially notable are the results achieved when test is performed using the group 2. Similar to MLE, this approach would be better suited for a bigger, more diverse dataset.

Ultimately, the dataset was trained using a ResNet18 architecture, however, employing logarithmic-scale power spectrograms and a DNN configured with 34 distinct features derived through various extraction techniques. In this scenario, the DNN utilizing the proposed feature set fails to deliver optimal performance when additional noise is introduced into the signal. This limitation is evident from the type of analysis conducted, which does not account for the temporal dependencies of the analyzed audio segments. Regarding the ResNet18-based detector, it consistently demonstrates superior results across all evaluated scenarios. This outcome is primarily due to the employed representation, as the power spectrogram adeptly captures variations across different frequencies, highlighting the most significant ones and thereby facilitating the learning process. Despite this, it is observed that as the signal degradation intensifies, the detection accuracy begins to plummet significantly. This decline is attributed to the network's lack of mechanisms for finer feature extraction that could enable more accurate classification.

6 CONCLUSION

Different target detection schemes employing both conventional signal processing techniques and machine learning have been demonstrated for an augmented version of the ShipsEar[9] dataset. The pre-

sented augmentation technique allowed the generation of more complex soundscape containing both the desired target and interfering noise from undesired targets. The detection schemes were evaluated for different cases of increasing complexity. The conventional algorithms were capable of reliable detection of the target in the less complex cases, but exhibited high false alarm rates in the presence of interfering vessels. The machine learning alternatives outperform the conventional algorithms. It can be argued that DNN detector necessitates a more thorough preliminary analysis to determine which features are most critical and to possibly incorporate some form of noise filtering or reduction. Consequently, it is evident that deep networks with more complex architectures can more effectively address this issue, particularly considering the presence of noise in the signal. Hence, deep convolutional network architectures represent a robust choice for detectors in these scenarios when working with spectrograms, yet the noise within the signal must be properly managed to enhance the overall outcomes.

ACKNOWLEDGMENTS

This research has been supported by the European Commission within the context of the project SMAUG (Smart Maritime and Underwater Guardian), funded under EU Horizon Europe Grant Agreement 101121129. We would also like to acknowledge the University of Vigo for collecting the ship noise dataset, ShipsEar[9].

REFERENCES

1. B. Ramirez and R. Bunker, “Narco-Submarines. Specially Fabricated Vessels Used For Drug Smuggling Purposes,” *CGU Faculty Publications and Research*, Jan. 2015.
2. C. Rainsford, “Peru, Ecuador Police Divers Target Drugs Hidden on Ship Hulls,” Sept. 2019.
3. M. Voytenko, “Cocaine found in parasite tube, attached to US tanker hull,” Aug. 2020. Section: Maritime Security.
4. “Smaug homepage.”
5. L. C. F. Domingos, P. E. Santos, P. S. M. Skelton, R. S. A. Brinkworth, and K. Sammut, “A Survey of Underwater Acoustic Data Classification Methods Using Deep Learning for Shoreline Surveillance,” *Sensors*, vol. 22, p. 2181, Jan. 2022. Number: 6 Publisher: Multidisciplinary Digital Publishing Institute.
6. X. Wang, J. Jiao, J. Yin, W. Zhao, X. Han, and B. Sun, “Underwater sonar image classification using adaptive weights convolutional neural network,” *Applied Acoustics*, 2019.
7. Y. Xu, X. Wang, K. Wang, J. Shi, and W. Sun, “Underwater sonar image classification using generative adversarial network and convolutional neural network,” *IET Image Process.*, vol. 14, pp. 2819–2825, 2020.
8. M. Chen, “Underwater acoustic signal recognition based on salient feature,” *ArXiv*, vol. abs/2312.13143, 2023.
9. 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, Dec. 2016.
10. M. Ainslie, *Principles of Sonar Performance Modelling*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010.
11. D. A. Abraham, *Underwater Acoustic Signal Processing: Modeling, Detection, and Estimation*. Modern Acoustics and Signal Processing, Cham: Springer International Publishing, 2019.
12. M. Trevorrow, “Examination of Statistics and Modulation of Underwater Acoustic Ship Signatures,” 2021.
13. K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

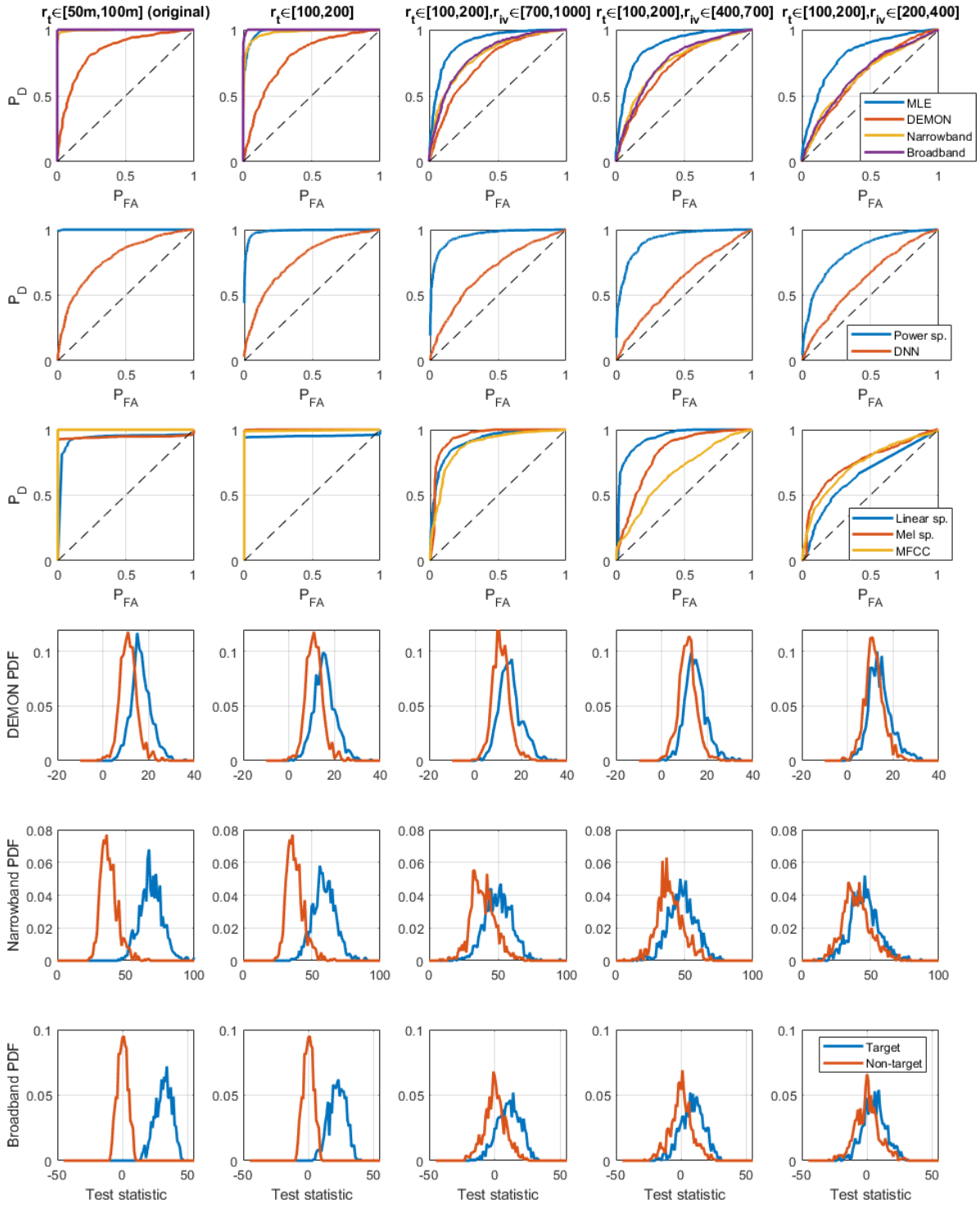

Figure 2: Each column of plots relates to the different cases 1 to 5 (from left to right). The top row shows the ROC curves for each detector. The three lower rows show the distributions for the test statistics of each conventional detector for both target realizations (blue) and non-target realizations (red). In all the plots date from group 3 are used as the test dataset. The ML detectors were trained using data from group 1.