International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-4, February 2020

Efficiency of Various Time-Frequency Representations in Deep Neural Network based Passive Sonar Target Classifiers

Suraj Kamal, Satheesh Chandran C., Supriya M.H.

Abstract*: Passive acoustic target classification is an exceptionally challenging problem due to the complex phenomena associated with the channel and the relatively low Signal to Noise Ratio (SNR) manifested by the pervasive ambient noise field. Inspired by the overwhelming success of Deep Neural Networks (DNNs) in many such hard problems, a carefully crafted network specifically for target recognition application has been employed in this work. Although deep neural networks can learn characteristic features or representations directly from the raw observations, domain specific intermediate representations can mitigate the computational requirements as well as the sample complexity required to achieve an acceptable error rate in prediction. As the sonar target records are essentially a time series, spectro-temporal representations can make the intricate relationship between time and spectral components more explicit. In a passive sonar target recognition scenario, since most of the defining spectral components reside at the lower part of the spectrum, a nonlinear dilated spectral scale having an emphasis on low frequencies is highly desirable. This can be easily achieved using a filterbank based time-frequency decomposition, which allows more filters to be positioned at the desired frequency ranges of interest. In this work, a rigorous analysis of the performance of time-frequency representations initialized at various frequency scales, is conducted independently as well as in combination. A convolutional neural network based spectro-temporal feature learner has been utilized as the initial layers, while a deep stack of Long Short Term Memories (LSTMs) with residual connections has been used for learning the intricate temporal relationships hidden in the intermediate representations. From the experimental results it can be observed that a linear scale spectrogram achieves an accuracy of 92.4% and 90.2% respectively for validation and test sets in the single feature configuration, whereas the gammatone spectrogram is capable of attaining an accuracy in the order of 96.7% and 96.1% respectively for the same. In a multifeatured setup however, the accuracy reaches up to 97.3% and 96.6% respectively, which reveals that a combination of properly initialized intermediate representations can improve the classification performance significantly.

***Keywords* : Deep Neural Network, Passive Sonar, Residual LSTM, Time-Frequency Representations.**

Revised Manuscript Received on February 28, 2020.
***** Correspondence Author
Suraj Kamal*, Department of Electronics, Cochin University of Science and Technology, Kochi, India. Email: surajkamal@cusat.ac.in
Satheesh Chandran C., Department of Electronics, Cochin University of Science and Technology, Kochi, India. Email: satheeshchandran@cusat.ac.in
Supriya M.H., Department of Electronics, Cochin University of Science and Technology, Kochi, India. Email: supriya@cusat.ac.in

I. INTRODUCTION

Passive sonar target recognition is an exceptionally hard pattern recognition problem rendered by the idiosyncrasies associated with the problem it is trying to solve as well as the unforgiving conditions it is operated in. Although the ocean medium is highly transparent to acoustic waves theoretically, many adversarial phenomena like severe time varying multipath propagation, extreme reverberations, anomalous scattering, frequency selective fading and multitude of competing noise sources make it really challenging in utilizing it for any practical purpose, target recognition being one of them [1], [2]. Due to its strategic importance, sonar target recognition has been extensively studied during the past few decades by various communities including statistical pattern recognition, signal processing and recently by the machine learning community, but remained rather unsolved or not robust enough to be utilized in a tactical scenario.

Recent advancements in the field of Artificial Neural Networks (ANNs) known as Deep Neural Networks or Deep Learning (DL) in general, have made significant breakthroughs in equally hard problems such as machine vision [3]-[5], speech recognition [6], [7], video games [8], [9], natural language processing [10] etc., a few to mention among many. The overwhelming success of deep learning is often attributed to the notion of feature learning as opposed to feature engineering in classical approaches, where a hierarchy of abstract characteristic features is learned along the progression of many successive layers [11]. The layered approach helps to form an invariant representation at higher layers that are resistant to the perturbations in the input signal. The robustness against domain variability is a desirable quality especially in the target recognition scenario, where sonar variability has profound impact on the reliability of the system. DNNs are often considered to be as end to end learning systems [12], where all the components of the system are part of the optimization loop or every component of the system is differentiable against the cost function used for optimization. This notion of end to end learning is quite true in image classification tasks, where the system learns the features directly from the raw pixels. However, in other areas like speech recognition, this is often not the case. Here, instead of taking raw waveform as the input, some hand engineered features such as Mel Frequency Cepstral Coefficients (MFCCs) that are non-differentiable,

are used as an intermediate representation before the actual learnable layers. Recent works have demonstrated that the requirement of such hard-coded intermediate representations can be avoided if there is enough data to characterize the domain and the facility for training sufficiently deep networks is available at hand [13]. However, getting such stupendously large data set in the sonar target recognition scenario is quite impractical as it demands massive investments in data collection operations, which are costly in terms of money, manpower as well as required time. Therefore, employing right intermediate representations can alleviate the dependency on the required amount of data to a significant extent. In this paper, efficiency of various Time-Frequency (TF) representations for passive sonar target recognition is explored in detail as well as a deep residual LSTM based network is proposed for the upstream target classification task.

II. TIME FREQUENCY REPRESENTATIONS

A passive sonar target record has dimensions comprising of time, frequency and quite often a spatial component if the sonar front end utilizes an array. The spatial component is used for localization and tracking of the target in general. It can be also used for spatial filtering in order to improve the SNR in multi-target scenarios. The spatially filtered signal can be further analyzed in both time and frequency. In a target classification scenario, the characteristic signature of a target is cumulatively delineated by time varying spectral components of noise from various internal machineries as well as hydrodynamic noise produced by the propellers and the hull. The overall noise made by a target measured at a distance is known as its acoustic signature. The harmonic structure of these signatures stays stationary for relatively short intervals. Hence estimating these subtle, time varying yet locally stationary structures embedded in the ambient noise field would help in identifying the target.

The time-frequency analysis offers a way to decompose a one-dimensional signal such as the sonar target record into a two-dimensional representation where the temporal evolution of the spectral structure becomes more explicit. Short Time Fourier Transform (STFT) is the most popular method employed for time-frequency analysis although many other methods such as wavelet transform (WT), Wigner-Ville distribution (WVD), Gabor Transform (GT), S transform etc. which offer various advantages over standard STFT exist. Standard STFT (spectrogram) has a linear scale in the frequency axis, i.e. having same resolution over the entire spectral range of analysis. However, studies in psycho-acoustics have identified various non-linear frequency scales at which human auditory perception occurs, often termed as perceptual scales. Generally, these scales have high resolution at lower frequencies and vice versa. In biological beings, these optimal perceptual scales have evolved over several millennia by the process of evolution through natural selection. This non-linear scale of perceptual importance also holds true in the passive sonar target recognition scenario as the low frequency components of the target signature contains most of the defining TF structures. Inspired from this, the scientists in various disciplines especially in signal processing and acoustics have developed mathematical formalism approximating these scales.

A. Filterbanks for TF Analysis

Filterbanks offer a common way to analyze a signal in TF domain. They are arrangements of low pass, high pass or band pass filters that are widely used in spectral decomposition and reconstruction of signals of interest [14]. The frequency spectrum of the signal is divided into finite sub-bands having a series of coefficients indexed in time denoting the average signal power centered in the band of the filter. Thus, an M-channel filterbank consists of M analysis filters characterized by different passbands, that decompose the input signal into several sub-band signals. Due to this discrete yet explicitly controllable selective band-limiting in the time-frequency surface, the filter banks can be used to mask out spurious artifacts that could otherwise hinder the performance of the upstream processing stages. This could also help in reducing the overall redundancies in the time frequency structure thereby improving the computational efficiency. Filterbanks can be used to create efficient TF representations analogous to utilizing STFT for the same [15]. The most widely used TF representations are in the form of spectrograms, which can be generated either using STFT or from filterbank outputs.

i) TF Representation using STFT

STFT is being widely used to generate TF representations from raw time domain target signal. The energy equivalent of STFT yields spectrogram [16], which is nothing but the squared magnitude of STFT. If $X(n,\omega)$ is the STFT of a target signal $x[n]$, then the corresponding spectrogram $S_x(n,\omega)$ is given by,

$$S_x(n,\omega) = |X(n,\omega)|^2 \tag{1}$$

In order to compute STFT, the target signal is divided into small segments of equal length using a windowing function and the Fourier transform of each segment is determined to yield the spectrum, which changes as a function of time. The signal $x_w(n,m)$ obtained by windowing the input signal $x[m]$ with a windowing function $w[m]$ shifted in time n, can be represented as,

$$x_w(n,m) = x[m].w[m-n] \tag{2}$$

where the subscript w is used to indicate the windowing function.

The Fourier transform $X_w(n,\omega)$ of the windowed signal can be written as,

$$X_w(n,\omega) = \sum_{m=-\infty}^{\infty} x_w(n,m).e^{-j\omega m} \tag{3}$$

Substituting (2) in (3), the Fourier transform can be rewritten as,

$$X_w(n,\omega) = \sum_{m=-\infty}^{\infty} x[m].w[m-n]e^{-j\omega m} \tag{4}$$

Since we are computing STFT over a finite set of frequencies only, it is more convenient and efficient to evaluate discrete STFT, as given by,

$$X_w(n,k) = X_w(n,\omega)|_{\omega=2\pi k/N} \tag{5}$$

where N is the sequence length and $k = 0,1, \dots, N-1$. Hence (4) can be modified as,

$$X_w(n,k) = \sum_{m=-\infty}^{\infty} x[m].w[m-n]e^{-j2\pi mk/N} \quad (6)$$

The corresponding spectrogram can be obtained by,

$$S_{xw}(n,k) = |X_w(n,k)|^2 \quad (7)$$

Substituting (6) in (7), the resulting spectrogram can be represented as,

$$S_{xw}(n,k) = \left| \sum_{m=-\infty}^{\infty} x[m].w[m-n]e^{-j2\pi mk/N} \right|^2 \quad (8)$$

It can be observed that the spectrogram is a TF representation as it is a function of both n and k components. In other words, this spectrogram can be thought of as a filtered output, where the filtering action is performed by the windowing function $w[m]$.

ii) TF Representation using Filterbanks

Instead of using STFT to create TF representations, filterbanks can be applied on the raw time domain signal to generate band-pass components, whose squared magnitude yields the spectrogram. The filterbank approach has the obvious advantage of placing the filters at desired locations along the frequency dimension. Let $X(\gamma)$ be the spectrum of the input signal $x[n]$ and $H(\gamma)$ be the transfer function of the filter with impulse response $h[n]$. In order to find the band-pass component of $x[n]$ at $\gamma = \omega$, the filter transfer function needs to be shifted such that it is centered around $\gamma = \omega$. The resulting band-pass component can be formulated as,

$$B_{xH}(n,\omega) = IFT_{\gamma}\{X(\gamma).H(\gamma-\omega)\} \quad (9)$$

where the subscript H is used to indicate the filter transfer function and IFT_{γ} is the inverse Fourier transform computed with respect to γ.

By the property of Fourier transform, any frequency domain multiplication is equivalent to convolution in time domain and frequency shift by ω in frequency domain corresponds to multiplication by $e^{j\omega n}$ in time domain. Applying these properties to (9), the equation can be rewritten as,

$$B_{xH}(n,\omega) = x[n] * h[n]e^{j\omega n} \quad (10)$$

Again applying $\omega = 2\pi k/N$ in (10), the band-pass component becomes,

$$B_{xH}(n,k) = x[n] * h[n]e^{j2\pi nk/N} \quad (11)$$

$$= \sum_{m=-\infty}^{\infty} x[m].h[n-m]e^{\frac{j2\pi k}{N}(n-m)} \quad (12)$$

$$= e^{\frac{j2\pi nk}{N}} \sum_{m=-\infty}^{\infty} x[m].h[n-m]e^{\frac{-j2\pi mk}{N}} \quad (13)$$

The corresponding spectrogram can be obtained by,

$$S_{xH}(n,k) = |B_{xH}(n,k)|^2 \quad (14)$$

Substituting (13) in (14), the spectrogram can be expressed as,

$$S_{xH}(n,k) = \left| \sum_{m=-\infty}^{\infty} x[m].h[n-m]e^{-j2\pi mk/N} \right|^2 \quad (15)$$

Here also, the spectrogram is a TF representation and can be thought of as a filtered output, where the filtering action is performed by the impulse response h. Comparing (8) and (15), it can be concluded that the spectrograms obtained by STFT as well as the filterbank methods are the same if $h[n] = w[-n]$. This can be achieved if the window $w[n]$ considered is an even function and its value is equal to the filter impulse response $h[n]$. In this work, the TF representations are generated using the filterbank approach, since it allows centre frequencies of the filters to be selected according to various scales.

B. Filterbanks at Various Scales

Due to the non-linear distribution of spectral components, where most of the interesting phenomena occur at the lower regions of the spectrum especially from 5 Hz to ~ 500 Hz and the relatively subtle features being distributed over the rest of the band, filterbanks need to be initialized at various perceptual scales in order to effectively capture the salient target features. The time-frequency analysis in perceptual scales is usually implemented in the form of an array of filters having their centre frequencies (f_c) distributed according to the chosen scale. The chosen scales can be classified into:

i) Linear Scale

The filterbanks initialized on a linear scale have their centre frequencies distributed uniformly in the range $\left[0, \frac{f_s}{2}\right]$, where f_s is the sampling frequency of the target signal. A typical time-frequency representation such as spectrogram has filterbanks initialized on this scale, with all the spectral regions of interest having equal representation and the bandwidth remains the same for all the filters in the filterbank.

ii) Logarithmic Scale

Here, the centre frequencies of the filterbanks are initialized on a logarithmic scale in the range $\left[0, \frac{f_s}{2}\right]$. The resulting time time-frequency representation is termed as logarithmic spectrogram, having filterbanks with centre frequency and bandwidth that increase logarithmically. Such a representation dilates the lower spectral regions of the target signature compared to a linear scale spectrogram. The relation between hertz (f) and logarithmic (l) scales can be obtained by,

$$l = \log_{10}(1+f) \quad (16)$$

where the constant '1' has been added to avoid $log(0)$ condition when $f = 0$.

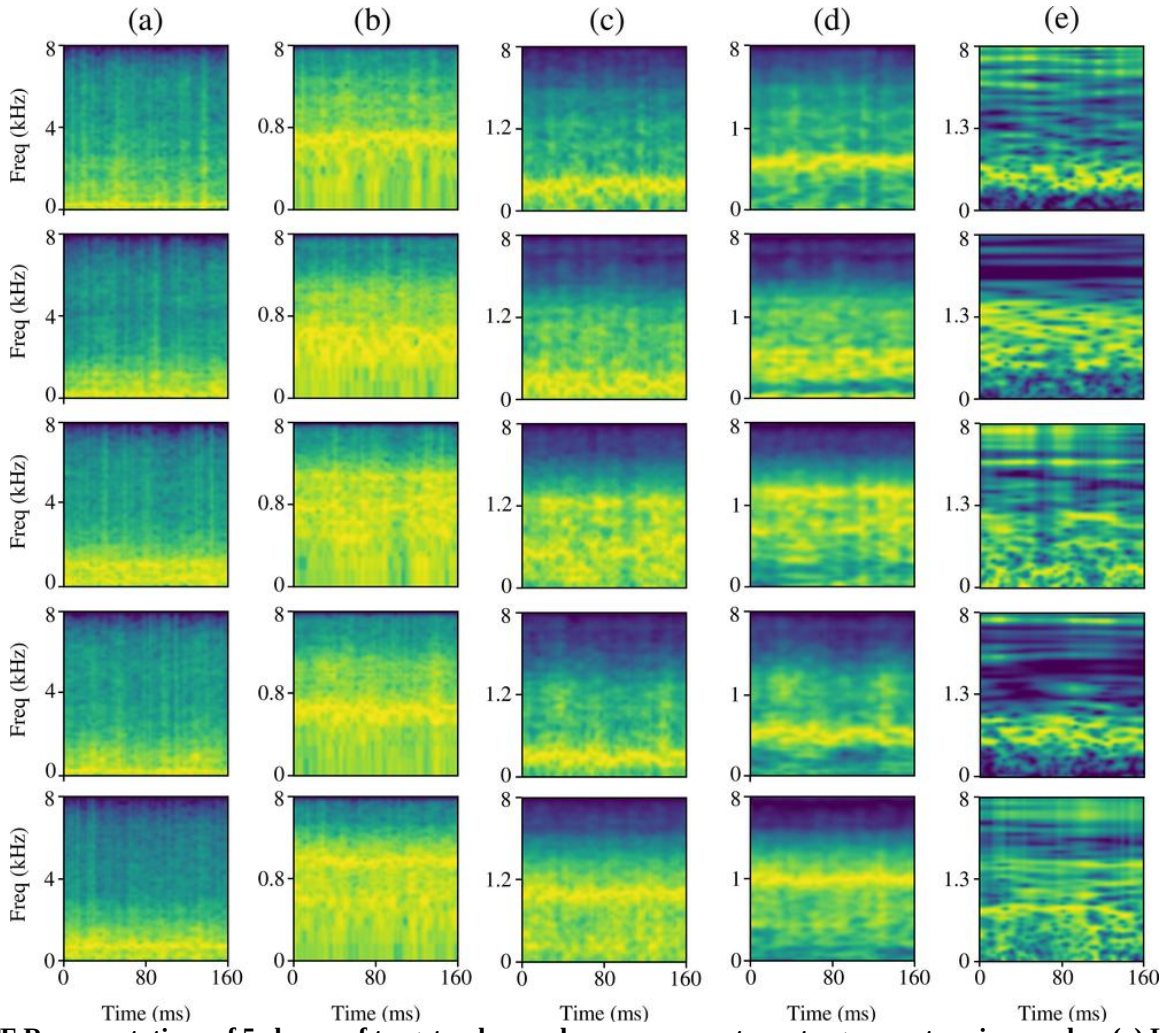


Fig. 1. TF Representations of 5 classes of targets where columns represent spectrograms at various scales: (a) Linear spectrogram, (b) Logarithmic spectrogram, (c) Mel spectrogram, (d) Gammatone spectrogram, (e) Constant Q spectrogram.

iii) Mel Scale

The Mel scale [17] involves a non-linear transformation of the frequency scale such that tones separated equally on Mel scale are also perceived by humans at equal distances from each other. It is observed that the mapping from actual frequency to Mel scale is almost linear below 1 kHz, while logarithmic above this frequency. In this type of initialization, the centre frequencies of the filterbanks are distributed on a Mel scale in the range $\left[0, \frac{f_s}{2}\right]$. The mapping from hertz (f) to Mel (m) scale is given by,

$$m = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \tag{17}$$

The spectrogram obtained with Mel scale initialization of filterbanks is referred to as Mel spectrogram, which emphasizes on the lower spectral region of the target signal, thereby revealing the salient cues.

iv) Equivalent Rectangular Bandwidth (ERB) Scale

In one of the most commonly used psychoacoustic scales known as ERB [18], [19], the filterbanks are organized such that their centre frequencies are distributed on an ERB scale in the range $\left[0, \frac{f_s}{2}\right]$. The mapping from hertz (f) to ERB scale can be achieved by the following transformation.

$$ERB = 24.7\left[\frac{4.37f}{1000} + 1\right] \tag{18}$$

Gammatone filterbank [20], a filterbank based on ERB scale, can be used to model the basilar membrane and the frequency response of the membrane at each position corresponds to the output of each filter in the filterbank. The spectrogram obtained using gammatone filterbank is known as gammatone spectrogram and the bandwidth of a gammatone filter is approximately 1.019 ERB for a filter of order 4. ERB scale also allocates more filters on the lower spectral region of the acoustic signal, which helps to concentrate on the characteristic subtle tonal components of the target signature.

v) Constant Q Scale

The filterbanks based on constant Q scale [21] have their centre frequencies geometrically spaced i.e. the ratio of the centre frequency to bandwidth of each filter remains constant throughout the entire frequency range of interest. Constant Q filterbanks yield constant Q spectrogram, where the bandwidth (Δf) of m^{th} filter can be represented as:

$$\Delta f_m = \left(2^{1/n}\right)^m \Delta f_0 \tag{19}$$

where n is the number of filters in one octave and f_0 is the centre frequency of the lowest filter in the filterbank.

In order to implement these filterbanks, the centre frequencies of all the filters are computed in accordance with the various scales such as linear,

logarithmic, Mel, ERB or constant Q. Using these centre frequencies, the filter coefficients are computed, which in turn determines the filter transfer function that completely characterizes the filterbank. The filterbanks initialized at various scales are then applied on the raw time domain target signals to obtain the band-pass components, as expressed in (11)-(13). The corresponding spectrograms can be obtained using (14)-(15), which constitute the TF representations for the initial layer of DNN. Fig. 1 depicts the TF representations corresponding to 5 different targets, obtained using filterbanks initialized at linear, logarithmic, Mel, ERB as well as constant Q scales. Each spectrogram has a time duration of 160 ms and is generated using a filterbank consisting of 64 filters, spanning a frequency range of approximately 8 kHz.

III. PROPOSED NETWORK ARCHITECTURE

Very deep networks have the obvious advantage of representing very complex functions as a composition of a hierarchy of functions in order to yield better performance [22]. However, they are notoriously hard to train due to the vanishing gradients problem [23]. The recent success of the decades old paradigm of neural networks can be attributed to various methods that have enabled the training of such deep networks by alleviating the vanishing gradients problem up to a certain extent. Batch normalization [24], Layer Normalization [25], Rectified Linear Units (ReLU) [26] and Residual learning [27] are some recent advancements that have enabled the successful training of very deep networks. Among these techniques, residual connections network is especially interesting as it enables training of networks in the order of hundreds of layers [28]. They are known to be good at creating very deep networks, which can effectively reduce the problem of vanishing gradients by creating bypass paths that offer a better way for gradients to flow backwards through several layers towards the initial layers.

A. Residual Learning

In residual learning, instead of approximating the underlying mapping function $\mathcal{H}(x)$, a residual function $\mathcal{H}(x) - x$ is learned by making the stacked layer output y as,

$$y = \mathcal{F}(x, W) + x \tag{20}$$

where $\mathcal{F}(x)$, parametrized by W, is the output of the layers to which the input x is added. So, if $y = \mathcal{H}(x)$ is still the underlying function to be learned, then setting $\mathcal{F}(x) = \mathcal{H}(x) - x$ results in $y = \mathcal{F}(x, W) + x = \mathcal{H}(x)$. This can be realized by providing skip connections between one or more layers in a network, performing an identity mapping between the inputs and the outputs. This formulation forces the network to learn the residual mapping function $\mathcal{F}(x, W)$. These identity mappings provide a shortcut path for gradients during backpropagation so that the gradients will reach deep down the network without significant degradation. This might eventually help the initial layers to learn better representations optimized on the training objective. In this work, a stacked layer of two-dimensional convolution layers is used for spectro-temporal feature learning, followed by the upstream sequence learning layers with residual connections.

B. Spectro-temporal Feature Learning

Convolutional Neural Networks (CNNs) are a specific variant of ANNs, introducing the concept of receptive fields, inspired by the cortical topographical organization of neurons in the mammalian neocortex [29], [30]. Two other important aspects of CNNs are the concept of weight reuse and local pooling of features. The concept of shared weight matrix offers many advantages such as a significantly smaller number of parameters compared to that of a fully connected network, which makes it less prone to overfitting and the property of shift invariance along the convolution dimension. There may be many of these convolutional layers stacked to form a deep network of nonlinear feature transforms, each of which can be written as:

$$a_f^l = \sum_{u,v} x_{u,v}^{l-1} * w_f^l + b \tag{21}$$

where l is the current layer, a_f^l is the output of the current layer for filter f known as the pre-activation, w_f^l is the shared weight matrix $u \times v$, b is the shared bias and $x_{u,v}^{l-1}$ is the input at the current layer, which is the area of the activation map coming under the receptive field. The variables u, v define the filter size that in effect determines the spectro-temporal area covered at once.

Convolutional networks offer an obvious advantage of taking multiple input features together in the form of input channels. The various spectro-temporal representations can hence be easily integrated together at the input for upstream feature learning. The intuition behind employing multiple features simultaneously at the input layer is that it would help in compensating the subtle yet defining characteristics missing in one or the other representation. Hence the features synergistically produce a better representation at the input of the network. The redundancy, if any, introduced by the combination of features however will be eliminated in the higher order layers during the error backpropagation step of gradient descent process.

C. Learning Temporal Structures

The acoustic signature of a target is essentially a time series having intricate sequential dependencies embedded within its temporal structure across many time frames, caused by the very mechanism which produces it, such as the rhythmic slamming of pistons in a reciprocating engine, propeller beats, flow noise and their temporal modulation as well. Although a deep feed forward network or a convolutional network can generalize temporal structures well beyond its context window from the basis time-frequency distribution on its higher layers, the architecture of the network itself does not offer any specific advantage in learning fine sequential dependencies hidden in the observations. Recurrent Neural Network (RNN) on the other hand provides a way to learn sequential dependencies in the signal by virtue of its structure.

However, despite their inherent advantages in modeling temporal sequences, vanilla RNNs also suffer from the vanishing gradients problem along with the progression of temporal steps, hence it is really hard to learn sequential dependencies between arbitrarily long sequences.

LSTMs [31] tackle this problem with differentiable gating mechanisms that bypass recurrent connections whenever the information is not relevant in

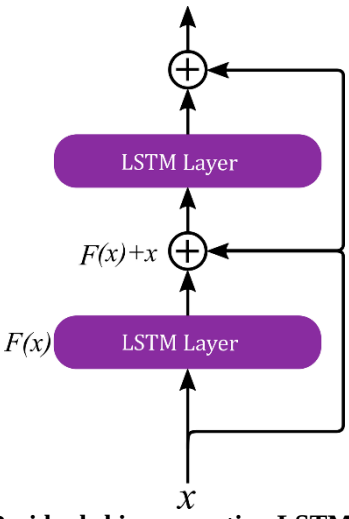


Fig. 2. Residual skip connection LSTM layers.

the sequential context. LSTMs are known to be good at learning very long dependencies in time series signals. Although stacking LSTM layers vertically can improve the performance of the system through hierarchical feature learning, compared to a shallower network, the recurring problem of vanishing gradients returns again as a roadblock.

The concept of residual learning might come as a rescue here. Combining LSTMs with residual learning helps in creating very deep networks capable of learning long temporal structures, while at the same time identifying the complex nonlinear relationships in a passive sonar target signature. A residual LSTM network has hence been utilized in this paper to classify the target signatures.

D. Residual LSTM Network

Residual learning over a deeply stacked LSTM can effectively mitigate the problem of vanishing gradients up to a significant extent. The input of layer l is added to the hidden state h_l of the network, which in turn is fed to the input of the following layer. The non-parametric addition operation does not contribute to the total number of learnable parameters.

The architecture of a vanilla LSTM network can be expressed as,

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \quad (22)$$

$$f_t = \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f\right) \quad (23)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (24)$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o) \quad (25)$$

$$h_t = o_t \odot \tanh(c_t) \quad (26)$$

where t is the time step, i_t is the input gate, f_t is the forget gate, c_t is the memory cell, o_t is the output gate, W, b are the parameters of the network, σ is the activation function, x_t is the input, h_t is the output of the memory network, called the hidden state and $\odot$ is the elementwise (Hadamard) multiplication. Thus, for a vertically stacked LSTM, the hidden state h_t^l of layer l can be computed as,

$$h_t^l = LSTM(x_t, h_{t-1}^l, c_{t-1}^l) \text{ if } l = 1 \quad (27)$$

$$h_t^l = LSTM(h_t^{l-1}, h_{t-1}^l, c_{t-1}^l) \text{ if } l > 1 \quad (28)$$

In a residual LSTM network, the hidden state $\hat{h}_t^l$ of layer l can be represented as,

$$\hat{h}_t^l = LSTM(h_t^{l-1}, h_{t-1}^l, c_{t-1}^l) \oplus x_t^{l-n} \quad (29)$$

where $\oplus$ indicates matrix addition and n denotes the number of LSTM layers present in between a residual connection.

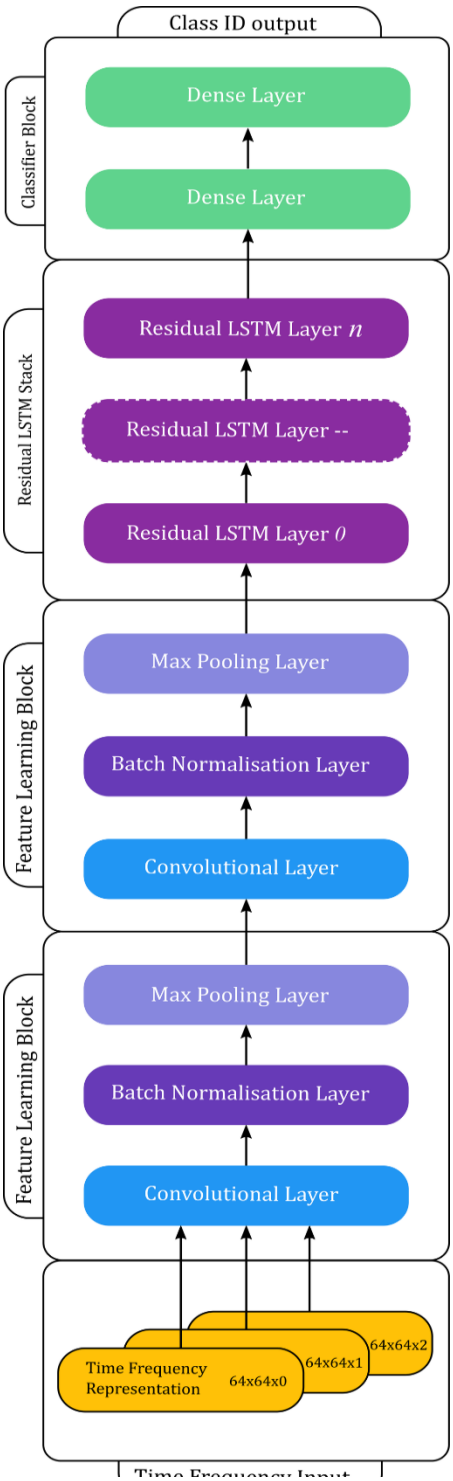
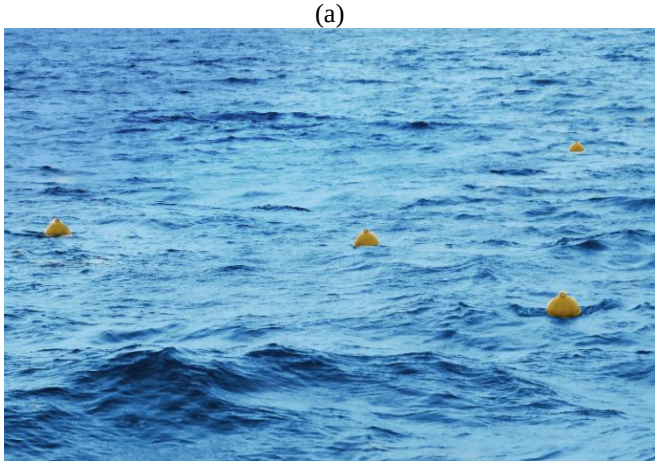


Fig. 3. The proposed network architecture.

Fig. 2 shows the residual connection between the LSTM layers. The LSTM memory block is followed by a set of fully connected feed forward network with a softmax transform at the output layer. Fig. 3 illustrates the complete network architecture including the CNN based spectro-temporal feature learners, residual temporal sequence learner as well as the fully connected feed forward classifier layers.

(a)



(b)

Fig. 4. Data collection buoys in the Arabian sea: (a) Deployment from vessel, (b) Spatially separated buoys.

IV. DATA COLLECTION

The field data required for the training and evaluation of the classifier system have been collected from the Indian Ocean, the shallows of Cochin in the vicinity of international transshipment terminal. The raw observations have been made at 96 ksps and 24 bps using up to five omni-directional hydrophones having input sensitivity of -180 dB re 1V/µPa with lateral as well as vertical separation of few meters in order to introduce limited but beneficial spatial variance. Recordings have been taken in different sea states and at various locations with diverse depth profiles ranging from 6 m to 50 m and the proximity of the shipping lane has helped in gathering many target records in different ambient noise conditions as well. Fig. 4 shows the deployment of the hydrophone array and marker floats from a vessel.

Most of these recordings have been made at the point of transit of the target platforms along the shipping lane at various cruising speeds ranging from 2 kn to 15 kn with the Closest Point of Approach (CPA) of about a few hundred meters. These recordings clearly demonstrate the challenging aspects of passive acoustic target recognition, such as shipping background noise, strong interference of broadband components due to severe multipath propagation, various marine biologics especially the constant crackling noise produced by the snapping shrimps, present throughout the recordings. Constant traffic of fishing boats is also considered as an unwanted interference. From the repertoire of the collected acoustic records, a set of 25 targets belonging to 10 broad categories with known parameters such as the engine type, propulsion system, hull size and speed has been chosen for evaluating the classifier.

The raw acoustic records have been partitioned into three records correspondingly. Each individual subset is subsequently converted into time-frequency representations, initialized on the scales described in section II. Each TF record is of 160 ms duration, with 64 frequency bins spanning from 5 Hz to 8.192 kHz. The entire dataset including the raw target records and the generated TF representations has been archived on a high-performance distributed data storage cluster in binary hdf format in order to achieve high read throughput during training. The corresponding size on disk is 8 GB approximately.

V. NETWORK TRAINING

The hierarchical spectro-temporal feature learning is performed in a supervised manner using the proposed deeply stacked network with error back propagation. A mini-batch gradient descent based on error deltas obtained from the cost function $L(\theta)$ has been used for updating the parameter (θ) space. Cross entropy or the negative log-likelihood between the prediction and the true target has been employed as the cost function, which can be expressed as,

$$\hat{h}_t^l = L(\theta) = -\frac{1}{n}\sum_{i=0}^{n-1}\sum_{j=0}^{C-1} y_{ij}\, logt_{ij} \tag{30}$$

where n is the total number of observations, y is the sparse encoded actual class labels and t is the probability matrix for each class in C number of classes. A softmax layer is used to express the network output as a multinomial distribution over all known classes such that,

$$\hat{y} = \frac{e^z}{\sum_{i=0}^{C-1} e_i^z} \tag{31}$$

where $\hat{y}$ is the conditional distribution $p(y = c|x, \theta)$ produced by the softmax layer, z is the vector of activation of the last fully connected layer and $argmax_i(\hat{y}_i)$ yields the most probable class. This aspect of softmax layer is particularly important in the context of the target classifier, since in an underwater contact detection and classification scenario, false positives and false negatives can have devastating effects. The classifier is optimized with a modern variant of the Stochastic Gradient Descent (SGD), known as the Adaptive Momentum (ADAM) optimizer [32]. ADAM optimizer adjusts the learning rate progressively by accumulating a decaying exponential moving average of the previous gradients as well as the squared gradients in the mini-batch gradient descent to provide better numerical stability.

The gradient is computed with adaptive estimates of the moments by,

$$\theta_t = \theta_{t-1} - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \varepsilon} \tag{32}$$

where $\hat{m}_t$ and $\hat{v}_t$ are the estimated bias corrected moments corresponding to biased first moment estimate (m_t) and second moment estimate (v_t) respectively. The smoothing term ε is added to avoid division by zero and is generally set to 10^{-8} in most scenarios.

Table- I: Performance scores of various TF representations sorted on validation accuracy

No.	TF Representations					Validation				Test			
	A	B	C	D	E	Accuracy	Precision	Recall	F1Score	Accuracy	Precision	Recall	F1Score
1	✓		✓	✓	✓	0.973	0.98	0.967	0.973	0.966	0.976	0.958	0.967
2	✓	✓	✓	✓	✓	0.972	0.978	0.967	0.972	0.965	0.973	0.957	0.965
3		✓	✓	✓	✓	0.971	0.98	0.958	0.969	0.963	0.977	0.95	0.963
4			✓	✓	✓	0.97	0.976	0.963	0.97	0.964	0.973	0.954	0.963
5	✓			✓		0.97	0.978	0.962	0.97	0.963	0.972	0.954	0.963
6	✓			✓	✓	0.969	0.976	0.959	0.967	0.964	0.975	0.954	0.964
7	✓	✓		✓	✓	0.968	0.978	0.959	0.969	0.963	0.973	0.953	0.963
8				✓	✓	0.968	0.976	0.959	0.968	0.963	0.974	0.954	0.964
9		✓		✓		0.968	0.977	0.958	0.968	0.96	0.971	0.95	0.96
10	✓		✓		✓	0.967	0.976	0.958	0.967	0.958	0.968	0.949	0.959
11		✓		✓	✓	0.967	0.975	0.958	0.966	0.964	0.972	0.953	0.963
12	✓	✓		✓		0.967	0.978	0.957	0.967	0.959	0.972	0.947	0.959
13			✓	✓		0.967	0.977	0.956	0.967	0.959	0.97	0.948	0.959
14				✓		0.967	0.977	0.953	0.965	0.961	0.971	0.95	0.961
15	✓	✓	✓		✓	0.966	0.974	0.955	0.965	0.958	0.969	0.947	0.958
16	✓		✓	✓		0.966	0.975	0.959	0.967	0.963	0.973	0.954	0.963
17		✓	✓	✓		0.966	0.974	0.957	0.966	0.962	0.971	0.954	0.963
18	✓	✓			✓	0.963	0.971	0.955	0.963	0.953	0.963	0.943	0.953
19		✓	✓		✓	0.961	0.972	0.949	0.96	0.954	0.967	0.941	0.954
20	✓	✓	✓	✓		0.961	0.971	0.951	0.961	0.954	0.966	0.943	0.954
21			✓		✓	0.961	0.974	0.949	0.961	0.949	0.966	0.933	0.949
22	✓				✓	0.956	0.969	0.943	0.956	0.941	0.959	0.927	0.943
23		✓			✓	0.952	0.966	0.938	0.952	0.947	0.965	0.932	0.948
24	✓	✓	✓			0.95	0.962	0.938	0.95	0.95	0.963	0.937	0.95
25	✓		✓			0.948	0.961	0.932	0.946	0.944	0.96	0.93	0.945
26	✓	✓				0.947	0.963	0.93	0.947	0.933	0.953	0.915	0.934
27		✓	✓			0.944	0.956	0.933	0.944	0.947	0.959	0.938	0.948
28			✓			0.938	0.96	0.913	0.936	0.936	0.955	0.917	0.936
29		✓				0.933	0.955	0.902	0.928	0.917	0.947	0.885	0.915
30	✓					0.924	0.944	0.904	0.923	0.908	0.931	0.889	0.91
31					✓	0.873	0.912	0.833	0.871	0.85	0.899	0.809	0.852
w_n	0.6	0.4	0.5	0.9	0.8								
η_i	0.924	0.933	0.938	0.967	0.873								

A=Spectrogram, B=Logarithmic Spectrogram, C=Mel Spectrogram, D=Gammatone Spectrogram, E=Constant Q Spectrogram

VI. RESULTS AND DISCUSSIONS

The proposed passive acoustic target classifier has been trained on a repertoire of target records, collected during multiple field trials, using an NVIDIA GPU cluster incorporating 24 GPUs and having an aggregate computational capability of around 250 TFLOPS. The time-frequency representations of various scales have been created and used for training the network. All combinations of the members of the TF_{rep} set such that $TF_{rep} = \{A, B, C, D, E\}$ without repetition have been considered for training a set of independent classifiers, where A, B, C, D and E denote the spectrogram, logarithmic spectrogram, Mel spectrogram, gammatone spectrogram and constant Q Spectrogram respectively.

Since TF_{rep} is an n element set, for every $0 \leq k \leq n$, the binomial coefficients $\binom{n}{k}$ can be used to determine the number of k-combinations that can be chosen from n elements of TF_{rep}. Hence the number of k-combinations possible from the set TF_{rep} is given by,

$$\binom{n}{k} = \frac{n!}{k!\,(n-k)!} \tag{33}$$

Substituting the values of k in (33), the possible k-combinations get evaluated to $\{1,5,10,10,5,1\}$ for $0 \leq k \leq 5$. Since the 0-combination is invalid in the context of network training, the total number of valid k-combinations can be computed as,

$$N_{TF\_rep} = \sum_{i} \binom{n}{k_i} = \sum_{i=1}^{5} \frac{n!}{k_i!\,(n-k_i)!} \tag{34}$$

which results in $N_{TF\_rep} = 31$ i.e. there will be 31 combinations of TF representations.

The set N_{TF_rep} defines the number of classifiers that can be trained on the dataset. In the machine learning parlance, this can be interpreted as a hyperparameter, which needs to be optimized based on the scoring criteria. Besides N_{TF_rep}, the number of residual layers in the LSTM network also can be considered as a hyperparameter. In the current work, the number of residual layers chosen for evaluation is $\{3, 5, 7, 9\}$. For the cardinality N_{lstm} of the set, the total number of classifiers N_c that need to be evaluated can be expressed as,

$$N_c = N_{TF\_rep} \times N_{lstm} \tag{35}$$

Since $N_{lstm} = 4$, the number of classifiers N_c selected for evaluation becomes 124.

From the set of classifiers evaluated, a set of candidates can be chosen based on their relative merits on the criteria of selection. In a passive sonar target classification context, recall assumes prime importance compared to others because it expresses the ability of a classifier to identify the few but all the potentially hostile targets from a large set of acoustic sources that are either friendly or neutral. A correlation analysis on the performance

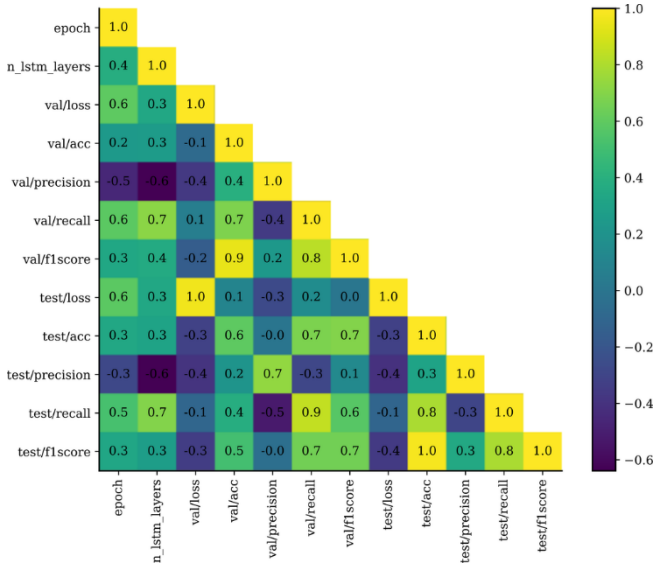


Fig. 5. Correlation plot of various scoring metrics.

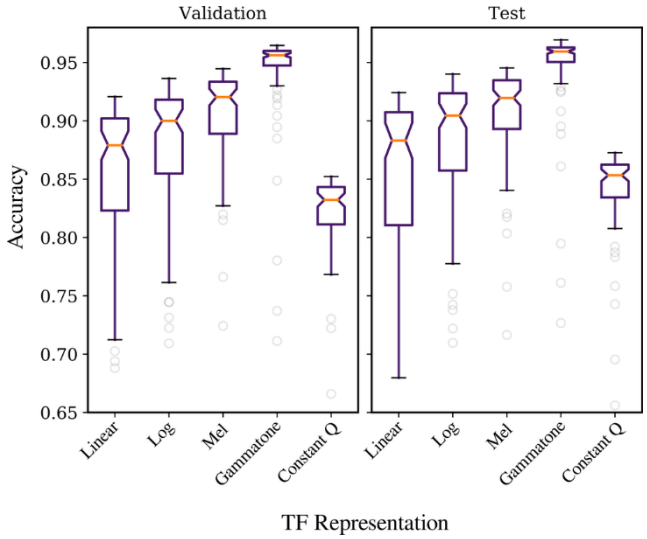


Fig. 6. Relative performance of individual TF representations on validation and test set.

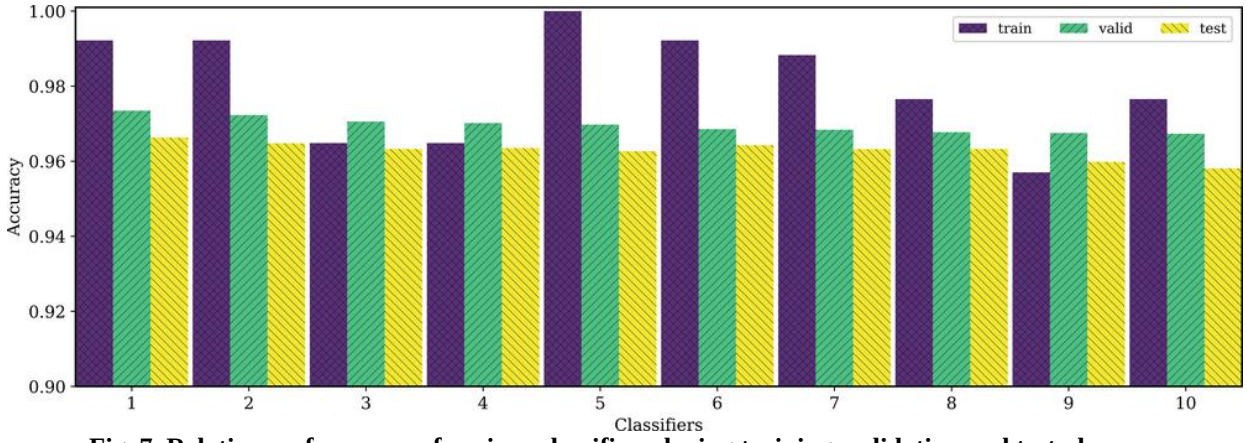


Fig. 7. Relative performance of various classifiers during training, validation and test phases.

metrics can be used to reveal its relationship to the hyperparameters under evaluation. A correlation plot of the classifiers is provided in Fig. 5, which shows a strong positive correlation between recall and the number of LSTM layers. Accuracy also has a positive correlation with the number of LSTM layers although it is much lower than that of recall. The corresponding correlation values for recall and accuracy have been found to be 0.7 and 0.3 respectively. Overall, the higher the number of LSTM layers, the better the performance in terms of recall and accuracy. Hence the classifier configurations having the maximum number of LSTM layers i.e. 9 in the current setup, have been chosen for final evaluation.

Table-I shows the comparative performance metrics of a set of classifiers sorted on the basis of validation accuracy. From the table, top N classifiers have been selected in order to estimate a relative weight w_n of each TF-representation such that,

$$w_n = n/N \tag{36}$$

where n is the number of occurrences of the feature under consideration within N members. A set of 10 classifiers has been chosen to estimate the relative weight, resulting in $w_n = n/10$. From this, it can be inferred that the gammatone spectrogram with a w_n of 0.9 contributes the most to classification performance, while logarithmic spectrogram with a w_n of 0.4 contributes the least. It can be observed that the individual classification accuracy η_i of gammatone spectrogram is well beyond the individual accuracies of all other spectrograms, i.e. almost 3% better than the nearest one.

It is also interesting to note that although the independent accuracy is rather low i.e. ~ 0.87 for constant Q spectrogram, its relative weight is much higher than others i.e. 0.8, which is only lower to that of gammatone spectrogram. This is because, while it is used in conjunction with other feature representations, it strongly helps in filling the latent defining attributes of the acoustic signature that aren't much obvious in others, which it can't do on its own.

The relative performance of individual TF representations on validation and test set is depicted in Fig. 6. It can be seen that the gammatone spectrogram has the highest average accuracy among all the representations, whereas constant Q exhibits the lowest accuracy. Gammatone is also found to have the least variance in accuracy, with linear spectrogram exhibiting the most. The training, validation and test accuracies of the top 10 classifiers mentioned in Table-I are plotted in Fig. 7. It can be observed that the classifier 1 having a TF representation combination of $\{A, C, D, E\}$, classifier 2 with a combination of $\{A, B, C, D, E\}$ and classifier 3 with $\{B, C, D, E\}$ combination yield a validation accuracy above 0.97. Hence it can be concluded that more features in combination results in better classification performance.

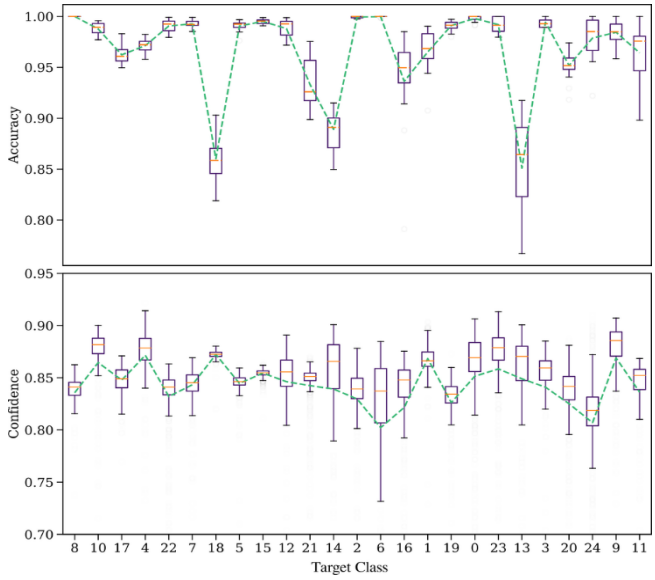


Fig. 8. Accuracy and confidence of classifiers with all hyperparameter configurations.

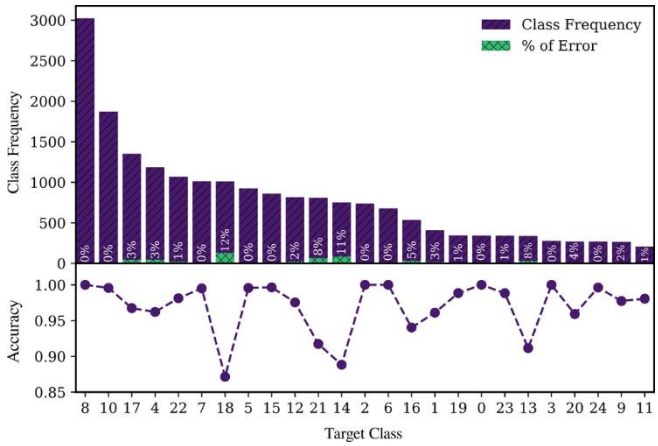


Fig. 9. Relative class frequency to error percentage for individual classes.

Fig. 8 shows the distribution of per class accuracy and confidence of prediction across all the 124 classifier configurations under evaluation. Although the accuracy for majority of the classes is clearly above 0.90, the classes 18, 14 and 13 have relatively lower accuracies and they also exhibit a high degree of variance. From Fig. 9, which illustrates the class frequency and respective error rate in percentage, it can be confirmed that these classes are not under represented. The lower accuracy might be due to ambiguities in the embedding space learned by the network, which leads to perplexities in the decision space.

The confidence of the network has a mean value around 0.85, and rarely crosses 0.95, i.e. the classifiers are not overconfident about their decision even if it is right in most of the scenarios. It can also be observed that the variance of the confidence of the classes having lower class frequency is generally high. From Fig. 9. it can be noted that class 18 has the maximum error which is about 12%, despite having a relatively good class representation frequency. Hence it can be concluded that class imbalance is not much of a problem for the proposed target classifier.

VII. CONCLUSION

Inspired from the immense success of deep neural networks in various classification problems, a specially crafted network for passive acoustic underwater targets has been employed in the current work. Despite the popular notion of end-to-end learning associated with deep learning, carefully tuned input representations have been found to improve the performance as well as reduce the overall resources needed in terms of both computation and data. In this work, a filterbank based time-frequency representation has been used as the input for the deep neural network, since a spectro-temporal decomposition can make the characteristic features of the acoustic emanations of a target more explicit, which is otherwise latent in the time domain records.

The filters have been initialized at various well-known perceptual scales such as Mel and ERB, as well as in a log scale, which generally emphasize the lower regions of spectrum where most of the defining tonal components of a target resides. In order to evaluate the efficiency of these representations, classifiers have been trained independently as well as with all their possible combinations. For better capturing the latent yet defining temporal dependencies, LSTM networks have been chosen, and to alleviate the problem of vanishing gradients in deep stacks, residual skip connections have been introduced.

The gammatone spectrogram initialized at ERB scale is observed to outperform the rest of the representations, with an accuracy of 0.967 in the single feature setup. In the multi-feature setup however, the maximum accuracy has been found to reach up to 0.973 for validation and 0.966 for test set. Even though the independent accuracy for the constant Q spectrogram is quite poor compared to others, it has remained throughout in the top 10 classifiers, when used in combination with others. From the correlation analysis on the performance metric for the entire combination of classifiers, it has been found that the residual layers have a positive correlation value of 0.7 to prediction recall and hence the classifier with 9 residual LSTM layers has been considered as a candidate for deployment.

A combination of these classifiers can be arranged as an ensemble to further improve the performance, which can be considered as a future work.

ACKNOWLEDGMENT

The authors gratefully acknowledge the Department of Electronics, Cochin University of Science and Technology, for extending all the necessary facilities to complete this work.

REFERENCES

1. A. Lee Swindlehurst, Brian D. Jeffs, Gonzalo Seco-Granados, Jian Li,. "Applications of Array Signal Processing." *Academic Press Library in Signal Processing: Volume 3*. Elsevier, 2014. 859-953
2. Knight, William C., Roger G. Pridham and Steven M. Kay. "Digital signal processing for sonar." *Proceedings of the IEEE 69.11*. IEEE, 1981.
3. Krizhevsky, Alex, Ilya Sutskever and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." *Advances in neural information processing systems* (2012): 1097-1105.
4. S, Ren, et al. "Faster r-cnn: Towards real-time object detection with region proposal networks." *Advances in neural information processing systems* (2015): 91-99.

5. Simonyan, Karen and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." *arXiv preprint* (2014): arXiv:1409.1556.
6. A, Hannun, et al. "Deep speech: Scaling up end-to-end speech recognition." *arXiv preprint* (2014).
7. Graves, Alex, Abdel-rahman Mohamed and Geoffrey Hinton. "Speech recognition with deep recurrent neural networks." *2013 IEEE international conference on acoustics, speech and signal processing*. IEEE, 2013.
8. D, Silver, et al. "Mastering the game of Go without human knowledge." *Nature* 550.7676 (2017): 354-359.
9. O, Vinyals, et al. "Grandmaster level in StarCraft II using multi-agent reinforcement learning." *Nature* 575.7782 (2019): 350-354.
10. J, Devlin, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." *arXiv* 1810.04805 (2018).
11. LeCun, Yann, Yoshua Bengio and Geoffrey Hinton. "Deep learning." *Nature* 521.7553 (2015): 436-444.
12. Glasmachers, Tobias. "Limits of end-to-end learning." *arXiv preprint* 1704.08305 (2017).
13. TN, Sainath, et al. "Factored spatial and spectral multichannel raw waveform CLDNNs." *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2016.
14. Mertins, A. and D. A. Mertins. *Signal analysis: wavelets, filter banks, time-frequency transforms and applications*. John Wiley & Sons, Inc., 1999.
15. Boashash, B. *Time-frequency signal analysis and processing: a comprehensive reference*. Academic Press, 2015.
16. Stankovic, L., M. Dakovic and T. Thayaparan. *Time-frequency signal analysis with applications*. Artech house, 2014.
17. Stevens, Stanley Smith, John Volkmann and Edwin B. Newman. "A scale for the measurement of the psychological magnitude pitch." *The Journal of the Acoustical Society of America* 8. 3 (1937): 185-190.
18. Moore, Brian CJ, and Brian R. Glasberg. "Suggested formulae for calculating auditory-filter bandwidths and excitation patterns." *The journal of the acoustical society of America* 74.3 (1983): 750-753.
19. Glasberg, Brian R., and Brian CJ Moore. "Derivation of auditory filter shapes from notched-noise data." *Hearing research* 47.1 (1990): 103-138.
20. J, Holdsworth, et al. "Implementing a gammatone filter bank." 1988.
21. Brown, Judith C. "Calculation of a constant Q spectral transform." *The Journal of the Acoustical Society of America* 89.1 (1991): 425-434.
22. Yoshua Bengio, Yann LeCun. "Scaling Learning Algorithms towards AI." Lin, C. J. Large-scale kernel machines. MIT press, 2007. 321-358.
23. Bengio, Yoshua, Patrice Simard and Paolo Frasconi. "Learning long-term dependencies with gradient descent is difficult." *IEEE transactions on neural networks* 5.2 (1994): 157-166.
24. Ioffe, Serge and Christian Szegedy. "Batch normalization: Accelerating deep network training by reducing internal covariate shift." *arXiv preprint* 1502.03167 (2015).
25. Ba, Jimmy Le, Jamie Ryan Kiro and and Geoffrey E. Hinton. "Layer Normalization." *arXiv preprint* 1607.06450 (2016).
26. Nair, Vinod and Geoffrey E. Hinton. "Rectified linear units improve restricted boltzmann machines." *Proceedings of the 27th international conference on machine learning (ICML-10)*. 2010.
27. K, He, et al. "Deep residual learning for image recognition." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.
28. He, K., et al. "Identity mappings in deep residual networks." *European conference on computer vision*. 2016. 630-645.
29. I, Kuzovkin, et al. "Activations of deep convolutional neural networks are aligned with gamma band activity of human visual cortex." *Communications biology* 1.1 (2018): 1-12.
30. Fukushima and K. "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position." *Biological cybernetics* 36.4 (1980): 193-202.
31. Hochreiter, Sepp, and Jürgen Schmidhuber. "Long short-term memory." *Neural computation* 9.8 (1997): 1735-1780.
32. Kingma, Diederik P and Jimmy Ba. "Adam: A method for stochastic optimization." *arXiv preprint* 1412.6980 (2014).

AUTHORS PROFILE

Suraj Kamal completed his Master of Science Programme in Electronics from College of Applied Science, Thodupuzha, affiliated to MG University, Kerala, India, in 2006. He is presently pursuing his Ph.D. in the Department of Electronics, Cochin University of Science and Technology, Kerala. His main areas of research are Applied Underwater Acoustics, Artificial Intelligence and Machine Learning.

Satheesh Chandran C. completed his Master of Technology in Electronics from Department of Electronics, Cochin University of Science and Technology, Kerala, India, in 2013 and is currently pursuing his Ph.D. in the same department. His current areas of research include Underwater Acoustics & Signal Processing, Machine Learning and Pattern Recognition.

Dr. Supriya M.H. obtained her Ph.D. from Department of Electronics, Cochin University of Science and Technology, Kerala, India, in 2008 and is currently working as Professor in the same department. She has more than 150 publications, 22 years of teaching experience and 4 years of Industrial experience. Her research and teaching areas of interest include Sonar Technology, Signal Processing as well as Underwater Target Recognition.