Robust Acoustic Event Classification Using Deep Neural Networks

Roneel V Sharan* and Tom J Moir

School of Engineering, Auckland University of Technology, Private Bag 92006, Auckland 1142, New Zealand

Email: roneel.sharan@aut.ac.nz, tom.moir@aut.ac.nz

* Corresponding Author

Abstract

Support vector machines (SVMs) have seen an increased usage in applications of acoustic event classification since its rise to popularity about two decades ago. However, in recent years, deep learning methods, such as deep neural networks (DNNs), have shown to outperform a number of classification methods in various pattern recognition problems. This work starts by comparing the classification performance of DNNs against SVMs with a number of feature representations which fall into two categories: cepstral features and time-frequency image features. Unlike related work, the classification performance of the two classifiers is also compared with feature vector combination and the training and evaluation times of the classifiers and features are also compared. The performance is evaluated on an audio surveillance database containing 10 sound classes, each class having multiple subclasses, with the addition of noise at various signal-to-noise ratios (SNRs). The experimental results shows that DNNs have a better overall classification performance than SVMs with both individual and combined features and the classification accuracy with DNNs is particularly better at low SNRs. The evaluation time of the DNN classifier was also determined to be the fastest but with a slow training time.

Keywords: Sound event recognition, time-frequency image feature, deep neural networks, support vector machines

1.0 Introduction

Advancements in machine learning algorithms can significantly improve the classification performance in pattern recognition problems. As mentioned in [11], one such advancement for automatic speech recognition (ASR) applications was the introduction of expectation maximization (EM) algorithm [4] for representing the relationship between the hidden Markov model (HMM) states and the acoustic input using Gaussian mixture model (GMM). Such techniques have also been employed in sound event recognition (SER) applications as in [6]. Another such advancement was with support vector machines (SVMs) with the introduction of soft margin [3] for non-separable datasets and the kernel trick [2] for non-linear datasets.

While artificial neural networks (ANNs) trained using back propagating error derivatives also had the potential to learn more accurate models, limitations in hardware and learning algorithms for training neural networks with many hidden layers and large amounts of data restricted progress along these lines. However, this changed over the last few years with advancements in computer hardware and machine learning algorithms giving rise to a modified machine learning algorithm called deep neural networks (DNNs). DNNs have found usage in a number of applications such as speech recognition [30], classification of electrocardiogram (ECG) signals [23], and in face completion and reconstruction [34]. As summarized in [11], DNN has been shown to outperform GMM for acoustic modeling in ASR on many different datasets by a number of research groups.

Furthermore, in two recent works in acoustics event classification [7, 15], DNN was shown to

perform better than other classifiers. However, these works only compare the classification performance and with mel-frequency cepstral coefficients (MFCCs) only. In another recent work in SER [18], the classification performance of SVM and DNN classifiers are compared with a number of features. These include MFCCs, features extracted from a stabilized auditory image (SAI) [38], and the spectrogram image feature (SIF) [6] where central moment values are extracted as features from the spectrogram image of sound signals. The overall classification performance of the DNN classifier was determined to be better than SVM with greater noise robustness.

This work is a continuation of our earlier work in [28] where we used SVM classification and considered a number of cepstral and time-frequency image features in trying to achieve robust sound classification in the presence of environmental noise in an audio surveillance application. In this work, we propose the use of DNNs for robust sound classification on the same database. The approach taken in this work is similar to [18], that is, the classification performance of the DNN classifier is compared against the results using SVM at various signal-to-noise ratios (SNRs) with a number of individual features, outlined in [28]. SVMs have been shown to perform on par, and in some literature, even better than some more traditional classification methods. For example, SVMs outperformed GMM in audio classification in [17] and HMM in [22]. More such comparisons, justifying the use of SVMs as the baseline classifier in this work, can be found in [29].

In addition, we compare: the classification performance of the two classifiers with feature vector combination, the training and evaluation time of the two classifiers, and the training and evaluation time of the various individual features and feature vector combinations. As such, in comparison to [7, 15, 18], this paper evaluates DNN performance with a range of cepstral and time-frequency image features, feature combinations, and also compares the training and evaluation times of classifiers and features, which, to the best of our knowledge, hasn't been done before for the application considered here.

The rest of this paper is organized as follows. Section 2 summarizes related work and section 3 gives an overview of the DNN classifier. The experimental setup, experimental results, and related discussions are given in section 4 and conclusion and recommendations are given in section 5.

2.0 Previous Works

For evaluating the performance of the classifiers, this work considers the same features as in [28] which can be broadly categorized as cepstral features and time-frequency image features. The cepstral features considered are MFCCs and gammatone cepstral coefficients (GTCCs). MFCCs for the t^{th} frame are obtained as the discrete cosine transform (DCT) of the compressed filter bank energies given as

$$c(q,t) = \sqrt{\frac{2}{M_1}} \sum_{m=1}^{M_1} \log\big(E(m,t)\big) \cos\left(\frac{\pi q}{M_1}(m-0.5)\right) \tag{1}$$

which is evaluated from $q = 1, 2, \ldots, Q$, where Q is the order of the cepstrum, $E(m,t)$ represents the filter bank energy of the m^{th} filter, and M_1 is the total number of mel-filters.

GTCCs are computed similar to MFCCs but utilize a gammatone filter bank instead of the triangular mel filter bank. The gammatone filter models the frequency selectivity property of the human cochlea and a commonly used cochlea model is that proposed by Patterson et. al. [21] which is a series of bandpass filters where the bandwidth is given by equivalent rectangular bandwidth (ERB). The three commonly used ERB filter models are given by Glasberg and Moore [8], Lyon's cochlea model as given in [31], and Greenwood [9]. An

efficient implementation of the gammatone filter bank provided in [32] has been utilized.

While conventional cepstral coefficients apply log compression to the filter bank energies before computing the DCT, linear cepstral coefficients, without any compression, were determined to be more noise robust and have a better overall classification performance for both MFCCs and GTCCs in [28]. Results using both log and linear compression will be presented here.

The use of time-frequency image derived features for sound classification has been seen in a number of literature [6, 14, 18, 28]. In [28], two types of time-frequency images were considered for feature extraction: spectrogram image and cochleagram image. The spectrogram derived features considered were: SIF, reduced spectrogram image feature (RSIF) [26], and the spectrogram image texture feature (SITF) [27]. With the SIF, the spectrogram image for each sound signal is divided into blocks, second and third central moments are computed in each block and concatenated to form the final feature vector. The v^{th} central moment for any given block of image is determined as

$$\mu_v = \frac{1}{S}\sum_{s=1}^{S}\left(I_s - \mu\right)^v \tag{2}$$

where S is the sample size or the number of pixels in the block, I_s is the intensity value of the s^{th} sample in the block, and μ is the mean intensity value of the block.

The RSIF is computed similar to SIF but has reduced feature dimensions, utilizing the mean and standard deviation values along the rows and columns of the blocks.

In addition, the SITF utilizes the image texture analysis technique of gray-level co-occurrence matrix (GLCM) [10], applied to the spectrogram image. GLCM is a matrix of frequencies where each element (a_x, a_y) is the number of times intensity value a_y is located at a certain distance and angle, given by the displacement vector $[d_k\ d_t]$, where d_k is the offset in the y direction and d_t is the offset in the x direction, from intensity value a_x in an $N_t \times N_k$ image I. Mathematically, this can be given as

$$p\left(a_x, a_y\right) = \sum_{k=1}^{N_k}\sum_{t=1}^{N_t}\begin{cases}1, & \text{if } I\left(k,t\right) = a_x \ \&\ I\left(k+d_k, t+d_t\right) = a_y \\ 0, & \text{otherwise}\end{cases} \tag{3}$$

where the size of the output matrix is $N_g \times N_g$, N_g is the number of quantized gray levels.

GLCM analysis is performed in subbands and the matrix values from each subband are concatenated to form the final feature vector for the SITF.

These same features were also extracted from the cochleagram image and now the SIF, RSIF, and SITF were referred as CIF, RCIF, and CITF, respectively. The cochleagram is a variation of the spectrogram image utilizing a gammatone filter. Cochleagram image derived features were determined to give a better overall classification performance in [28] and also much more noise robust, therefore, only cochleagram image derived features are considered in this work.

SVMs are used for classification in [28] and it is used as a baseline classifier in this work. SVM determines the optimal hyperplane to maximize the distance between any two given classes. The initial SVM was a linear classifier [35] which was later extended to nonlinear datasets [2]. Consider a set of l training samples belonging to two classes given as $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_l, y_l)\}$, where $\mathbf{x}_i \in R^d$ is a d-dimensional feature vector representing the i^{th} training sample, and $y_i \in \{-1, +1\}$ is the class label of $\mathbf{x}_i$. The optimal hyperplane can be

determined by minimizing $\frac{1}{2}\|\mathbf{w}\|^2$ subject to $y_i(\mathbf{w}\cdot\mathbf{x}_i + b) \geq 1$, where $\mathbf{w} \in R^d$ is a normal vector to the hyperplane and b is a constant. The optimization is solved under the given constraints by the saddle point of the Lagrange functional.

For linearly nonseparable problems, the optimization can be generalized by introducing the concept of *soft margin* [3]. Introducing non-negative slack variables ξ_i which measure the degree of misclassification of data $\mathbf{x}_i$ and a penalty function $\sum_i \xi_i$, the optimization is a trade-off between a large margin and a small error penalty.

In applications where linear SVM does not give satisfactory results, nonlinear SVM is suggested which aims to map the input vector $\mathbf{x}$ to a higher dimensional space $\mathbf{z}$ through some nonlinear mapping $\phi(\mathbf{x})$ chosen *a priori* to construct an optimal hyperplane. The *kernel trick* [2] is applied to create the nonlinear classifier where the dot product is replaced by a nonlinear kernel function $K(\mathbf{x}_i, \mathbf{x}_j)$ which computes the inner product of the vectors $\phi(\mathbf{x}_i)$ and $\phi(\mathbf{x}_j)$. A commonly used kernel function is Gaussian radial basis function (RBF), $K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / 2\sigma^2\right)$, where $\sigma > 0$ is the width of the Gaussian function.

The classifier for a given kernel function with the optimal separating hyperplane is then given as

$$f(\mathbf{x}) = \operatorname{sgn}\left(\sum_{i=1}^{l} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b\right). \tag{4}$$

with α_i being the Lagrange multipliers.

In [26], the performance of four commonly used multiclass SVM classification methods were compared and the classification performance of the one-against-all (OAA) multiclass classification method was generally found to be the best. As such, only the OAA multiclass classification method is considered in this work. OAA is probably the earliest of the multiclass SVM classification techniques [36]. For a P-class problem, P binary SVM classifiers are constructed and evaluated where the i^{th} classifier is trained with all the training samples from the i^{th} class as positive labels and all the remaining samples as negatives labels. During classification, a sample $\mathbf{x}$ is classified in the class with the largest value of the decision function which can be given as

$$f(\mathbf{x}) = \arg\max_{i=1,2,\ldots,P} \left(\mathbf{w}^i \cdot \phi(\mathbf{x}) + b^i\right). \tag{5}$$

3.0 Deep Neural Networks

The methods for DNNs are now available in a number of literature, such as [11, 18, 20], and is summarized here. A DNN, as defined in [11], is a feed-forward ANN with more than one layer of hidden units between the inputs and outputs. The training data in a DNN can be modeled using a two-layer network known as a restricted Boltzmann machine (RBM). RBMs were invented by Smolensky in 1986 [33] but only gained attention in early 2000s after development of fast learning algorithms by Hinton [12]. A RBM is a generative energy based model that consists of a layer of stochastic binary visible units with undirected connections to a layer of binary hidden units but no visible-visible or hidden-hidden connections.

The DNN classifier [20] has L-layers with the feature vectors on the input layer and the output layer in a one-of-P configuration. The DNN is constructed using individual pre-trained RBM pairs with each pair comprising V visible and H hidden stochastic nodes, $\mathbf{v} = [v_1, v_2, \ldots, v_V]^T$ and $\mathbf{h} = [h_1, h_2, \ldots, h_H]^T$. This work uses Bernoulli-Bernoulli RBM (BBRBM) structures, however, the input layer can also be formed using Gaussian-Bernoulli

RBM (GBRBM) structures as in [18]. Assuming binary nodes for the BBRBM structure, that is, $\mathbf{v}_{bb} \in \{0,1\}^V$ and $\mathbf{h}_{bb} \in \{0,1\}^H$, the energy function of the state $E_{bb}(\mathbf{v},\mathbf{h})$ can be given as

$$E_{bb}(\mathbf{v},\mathbf{h}) = -\sum_{i=1}^{V}\sum_{j=1}^{H} v_i h_j w_{ji} - \sum_{i=1}^{V} v_i b_i^v - \sum_{j=1}^{H} h_j b_j^h \tag{6}$$

where w_{ji} is the weight between the i^{th} visible unit and the j^{th} hidden unit and b_i^v and b_j^h are the real valued biases, respectively. The BBRBM model parameters are $\theta_{bb} = \{\mathbf{W}, \mathbf{b^h}, \mathbf{b^v}\}$ where the weight matrix is given as $\mathbf{W} = \{w_{ij}\}_{V\times H}$ with biases $\mathbf{b^h} = [b_1^h, b_2^h, \dots, b_H^h]^T$ and $\mathbf{b^v} = [b_1^v, b_2^v, \dots, b_V^v]^T$.

The joint probability associated with configuration $(\mathbf{v},\mathbf{h})$ can then be given as

$$p(\mathbf{v},\mathbf{h};\theta) = \frac{1}{Y} e^{\{-E(\mathbf{v},\mathbf{h};\theta)\}} \tag{7}$$

where Y is a partition function given as $Y = \sum_v \sum_h e^{\{-E(\mathbf{v},\mathbf{h};\theta)\}}$.

During pre-training, the training data is used to estimate the RBM model parameter θ with maximum likelihood learning using the contrastive divergence (CD) algorithm [13]. CD gives a simple approximation of the gradient of the log probability of the training data. A better generative model is learned through a limited number of steps of alternating Gibbs sampling by updating the hidden nodes $\mathbf{h}$ given the visible nodes $\mathbf{v}$ and then using the updated $\mathbf{h}$ to update $\mathbf{v}$. The training starts at the input layer, which is fed with the feature vectors, and form the visible nodes. The hidden units determined after the training process form the visible units for training the next RBM visible units. Multiple layers of RBMs are trained by repeating this process as many times as desired and, in the end, the RBMs are stacked to form a DNN as a single, multilayer generative model.

In fine-tuning, a *softmax* output labeling layer of size P is added which aims to convert a number of units in the final layer into a multinomial distribution using the *softmax* function

$$p(r \mid \mathbf{h_L}; \theta_\mathbf{L}) = \frac{\phi(r, \theta_L)}{\sum_{p=1}^{P} \phi(p, \theta_L)} \tag{8}$$

where r is an index over all classes, θ_L are the model parameters for the DNN, $\phi(r,\theta_L) = e^{\{\sum_{i=1}^{H} w_{ki} h_i + b_r\}}$, and $p(r|\mathbf{h};\theta_L)$ is the probability of the input being classified into class r.

Back propagation derivatives of a cost function, which measures the discrepancy between the predicted outputs and the actual outputs c for each training case [25], can then be used to discriminatively train the DNN. With the *softmax* output function, the cross entropy is the natural choice of cost function C between the desired and actual distributions given as

$$C = -\sum_{r=1}^{P} c_r \log p(r \mid \mathbf{h}; \theta_L). \tag{9}$$

More on the setting for the various DNN parameters and the DNN structure for the various features considered in this work can be found in the experimental setup, section 4.1.

4.0 Experimental Evaluation

An overview of the experimental setup is given first followed by the classification

performance using individual features and then feature combination. Finally, the training and evaluation time of the classifiers and features are compared.

4.1 Experimental Setup

The sound database has a total of 1143 files belonging to 10 classes: *alarms*, *children voices*, *construction*, *dog barking*, *footsteps*, *glass breaking*, *gunshots*, *horn*, *machines*, and *phone rings*. Each sound class contains multiple subclasses with interclass similarity and intraclass diversity as demonstrated in [26]. The sound files are largely obtained from the Real World Computing Partnership (RWCP) Sound Scene database in Real Acoustic Environment [19] and the BBC Sound Effects library. All signals in the database have 16-bit resolution and a sampling frequency of 44100 Hz. More details about the sound database and its comparison with that used in other similar work can be found in [26].

The classification performance is evaluated under three different noise environments taken from the NOISEX-92 database [37]: *speech babble*, *factory floor 1*, and *destroyer control room*. The performance is evaluated in clean conditions and at 20dB, 10dB, 5dB, and 0dB SNRs. Illustrations of cochleagram image of a sample sound signal, mapped to the HSV color space for better visualization, under clean conditions and with the addition of noise at 0dB SNR can be seen in Fig. 1.

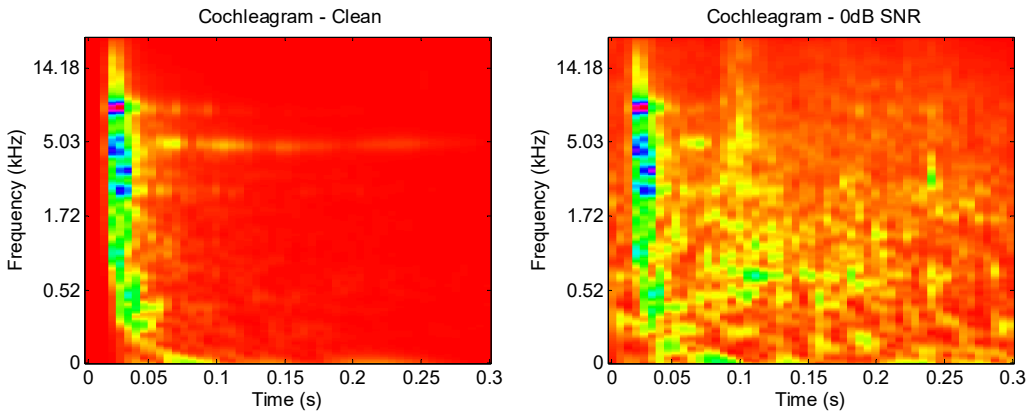

Figure 1: Cochleagram images for a sample sound signal from *construction* sound class. (a) Cochleagram image under clean conditions and (b) cochleagram image at 0dB SNR with *factory* noise [28].

For all experiments, signal processing is carried out using a Hamming window of 512 points (11.61 ms) with 50% overlap. For features utilizing the gammatone filter, only results using the best performing ERB model, as determined in [28], are reported. The classification accuracy is given in percentage as *number of correctly classified test samples* divided by the *total number of test samples*. Nonlinear SVM with a Gaussian RBF kernel is used in all cases as it was found to give the best results. The classifier parameters were tuned using cross validation. In tuning the parameters, one set of parameters which gave the best average classification accuracy were selected rather than determining the optimal parameters for each noise condition. For all experimentations, the classifier is trained with two-third of the clean samples with the remaining one-third of the samples used for testing under clean and noisy conditions.

For the DNN classifier, the number of hidden layers and their dimensions were determined through experimentation in each case, following a similar procedure to [18]. That is, a step-wise search of hidden layer widths between 10 and 400 was performed. The resolution in each case was set to 10 and the internal layers were constrained to equal size. Similar to [18], results are presented using only two hidden layers for all the features since the addition of more hidden layers was only seen to give a marginal improvement in classification performance but with significant increase in computation time. The final DNN structures for all features are given in Table I where the input and output layers are equal to the feature dimension and number of classes, respectively. In addition, for all experiments, the batch training size was set to 127, one-sixth of the number of training samples, and using 1000 training epochs.

Table I: Final DNN structures for all features

Feature	DNN Structure			
	Input Layer	Internal Layer 1	Internal Layer 2	Output Layer
MFCCs and GTCCs	72	50	50	10
SIF and CIF	162	60	60	10
RSIF and RCIF	72	50	50	10
SITF and CITF	256	60	60	10
Linear GTCC + CIF	234	160	160	10
Linear GTCC + RCIF	144	100	100	10
Linear GTCC + CITF	328	160	160	10

4.2 Results Using Individual Features

The classification performance of the SVM and DNN classifiers is compared using various individual features in this subsection which are grouped into cepstral features and time-frequency image features.

4.2.1 Cepstral Features

The classification accuracy values for MFCCs and GTCCs, for both log and linear compression, using SVM and DNN classification methods are given in Table II. The results were obtained using the optimal parameter settings for MFCCs and GTCCs as reported in [28]. For both the features, the feature vector for each frame in the sound signal is 36 dimensional which includes the first 12 cepstral coefficients and the first and second derivatives. The final feature vector is 72 dimensional, a concatenation of the mean and standard deviation values from each dimension.

Table II: Classification accuracy values for MFCCs and GTCCs using SVMs and DNNs

Feature	SVM						DNN					
	Clean	20dB	10dB	5dB	0dB	Average	Clean	20dB	10dB	5dB	0dB	Average
Log MFCC	97.11	92.21	73.32	60.54	47.77	74.19	96.85	90.03	81.36	66.05	50.48	76.96
Log GTCC	96.33	94.58	77.78	70.43	55.03	78.83	96.85	95.19	81.98	68.24	57.04	79.86
Linear MFCC	96.06	93.70	84.25	74.98	60.72	81.94	95.28	95.10	88.19	78.65	65.18	84.48
Linear GTCC	96.85	93.96	87.75	80.93	61.77	84.25	95.80	95.63	88.80	81.80	66.49	85.70

For both log and linear cepstrums, the average classification accuracy values for GTCCs are higher than MFCCs using both the classification methods. Also, the best average classification accuracy for both features is achieved using linear compression. As far as the

performance of the two classifiers is concerned, DNN is seen to give the highest average classification accuracy in all cases. The improvement in average classification accuracy is 2.77%, 1.03%, 2.54%, and 1.45% for log MFCCs, log GTCCs, linear MFCCs, and linear GTCCs, respectively. As such, the most improved overall results are with log MFCCs but linear GTCCs are the best performing cepstral feature. In general, there isn't a significant change in the classification accuracy under clean conditions and at 20dB SNR. However, the classification accuracy values are generally significantly improved as the SNR decreases. For example, at 0dB SNR, the improvement is 2.71%, 2.01%, 4.46, and 4.72% for log MFCCs, log GTCCs, linear MFCCs, and linear GTCCs, respectively.

4.2.2 Time-Frequency Image Features

Next, the classification accuracy values for the time-frequency image features are presented using SVM and DNN classifiers. Only results using cochleagram image derived features are reported here since it has been determined to be more effective than spectrogram image derived features in [28]. The classification accuracy values for the three cochleagram image features, CIF, RCIF, and CITF, using SVM and DNN classifiers are given in Table III. For CIF and RCIF, the cochleagram image is divided into 9×9 blocks and second and third central moments are computed in each block. For CIF, these values are concatenated to form a 162 dimensional final feature vector for each sound signal. For RCIF, the mean and standard deviation of the two central moment values along the rows and columns of the blocks are concatenated to form a 72 dimensional final feature vector. For the CITF, the GLCM image texture analysis technique is applied to the cochleagram images. GLCM analysis is carried out in subbands, with 64 determined to be the optimum number of subbands, at an angle of $45°$ with $N_g = 2$ and $d = 1$, as determined to give the optimal results in [27]. The final feature vector dimension for the CITF is, therefore, equal to 256 ($N_g^2 \times 64$, where 64 refers to the number of subbands).

Table III: Classification accuracy values for CIF, RCIF, and CITF using SVM and DNN

Feature	SVM						DNN					
	Clean	20dB	10dB	5dB	0dB	Average	Clean	20dB	10dB	5dB	0dB	Average
CIF	93.18	93.09	92.21	89.06	63.95	86.30	94.75	94.66	93.79	90.55	70.87	88.92
RCIF	94.75	94.75	94.58	91.69	69.38	89.03	96.06	95.54	95.19	92.39	72.70	90.38
CITF	92.65	92.65	92.21	90.38	78.30	89.24	95.80	95.63	95.45	95.19	88.54	**94.12**

For all three cochleagram image derived features, the average classification accuracy value using the DNN classifier are determined to be higher than the SVM classifier, as with cepstral features. The improvement in the average classification accuracy value over SVM is 2.62%, 1.35%, and 4.88% for CIF, RCIF, and CITF, respectively. While there is improvement in classification performance for all three features using DNN classification, the improvement in classification performance is seen to increase with the feature vector dimension. This suggests that while the DNN classifier always gives a better overall classification performance than the SVM classifier, the DNN classifier is much more suitable with higher feature dimensions compared to the SVM classifier.

The CITF produces the highest average classification accuracy using both classifiers and also the most improved classification performance from SVM to DNN classification. Unlike cepstral features, the classification accuracy under each noise condition has improved using DNN classification for all three cochleagram features. For the CITF, the improvement in classification accuracy is 3.15%, 2.98%, 3.24%, 4.81%, and 10.24% under clean conditions and at 20dB, 10dB, 5dB, and 0dB SNRs, respectively. As such, in general, the improvement

in classification performance increases as the SNR decreases.

4.3 Results Using Feature Combination

This section compares the classification performance of SVM and DNN classifiers using a combination of cepstral and time-frequency image features. For the cepstral features, only linear GTCCs are considered here being the best performing cepstral feature. The classification accuracy values using a combination of linear GTCCs and cochleagram image derived features are given in Table IV using SVM and DNN classification methods.

Table IV: Classification accuracy values for linear GTCCs in combination with cochleagram image derived features using SVM and DNN

Linear GTCCs +	SVM						DNN					
	Clean	20dB	10dB	5dB	0dB	Average	Clean	20dB	10dB	5dB	0dB	Average
CIF	96.06	95.98	95.28	93.35	76.99	91.53	97.38	97.38	96.59	95.28	85.39	94.40
RCIF	97.64	97.38	96.59	91.51	79.79	92.58	98.16	98.16	97.81	95.71	87.23	95.42
CITF	96.59	96.59	95.36	94.23	83.73	93.30	97.90	97.81	97.64	95.71	91.25	**96.06**

With both SVM and DNN classification methods, the average classification accuracy values for all cochleagram image derived features show improvement when combined with linear GTCCs. Compared to the average classification accuracy of CIF, RCIF, and CITF, when combined with linear GTCCs, the improvement is 5.23%, 3.55%, and 4.06%, respectively, for SVM classification. Similarly, with DNN classification, the improvement is 5.48%, 4.62%, and 1.94% when compared to the results for CIF, RCIF, and CITF, respectively.

Using both classification methods, the feature combination of linear GTCCs and CITF gives the highest average classification accuracy. Also, once again, the average classification values using DNN classification are higher than SVM classification for all feature combinations with an improvement in classification accuracy under all noise conditions. For the best performing feature set of linear GTCCs + CITF, the improvement in classification accuracy is 1.31%, 1.22%, 2.28%, 1.48%, and 7.52% under clean conditions and at 20dB, 10dB, 5dB, and 0dB SNRs, respectively. As such, the improvement in classification performance from SVM to DNN is marginal and much more even when compared to the best performing individual features except at 0dB SNR where there is a significant improvement in classification accuracy.

4.4 Further Analysis

4.4.1 Classification Performance Comparison with [18]

For various individual and combined features, the DNN classifier has been seen to outperform the SVM classifier in terms of overall classification performance and noise robustness. The classification accuracy results for the individual cepstral and time-frequency image features also have some similarity to the results in [18]. For example, in [18], for MFCCs, the improvement in classification performance from SVM to DNN is -9.0%, 20.7%, 22.9%, and 8.5% under clean conditions and at 20dB, 10dB, and 0dB SNRs, respectively, with an improvement of 10.8% in the average classification performance. In our work, for linear GTCCs, the best performing cepstral feature, the improvement in classification accuracy over the baseline classifier is -1.05%, 1.67%, 1.05%, 0.87%%, 4.72% under clean conditions and at 20dB, 10dB, 5dB, and 0dB SNRs, respectively, with an improvement of 1.45% in the average classification accuracy. While our work does not achieve as much improvement in classification performance as in [18], it should be noted that the evaluation task in [18] was identical to [5] and the results for MFCC-SVM were taken from [5]. It is

understood that linear SVM is used in [5] using the OAO multiclass classification method. In our earlier work in [26], the classification performance using nonlinear SVM and OAA multiclass classification method were determined to be better than linear SVM and OAO multiclass classification method, respectively, which could explain the lesser improvements in classification performance in our work than in [18].

One possible explanation for the slight decline in the classification performance from SVM to DNN for cepstral features under clean conditions could lie with the approach taken in tuning the classifier parameters. One set of classifier parameters were determined through experimentation which produced the best average classification accuracy across all noise conditions. It is more than likely that better classification accuracy will be obtained under each noise condition if optimal classifier parameters were determined and utilized under each noise condition. However, this would be an unrealistic implementation since, in practice, the classifier would need to determine the noise level before deciding the classifier parameters to use. This is something that the developed system is currently not equipped with and, therefore, one set of classifier parameter settings is seen as the way forward for now. Besides, cepstral features are not the primary feature considered in this work so this wasn't investigated further.

For the SAI features, the improvement in classification performance from SVM to DNN in [18] is 1.87%, 1.80%, 8.07%, 9.40% under clean conditions and at 20dB, 10dB, and 0dB SNRs, respectively, with an improvement of 5.28% in the average classification performance. For the CITF, the best performing time-frequency image feature in our work, the improvement in classification performance is 3.15%, 2.98%, 3.24%, 4.81%, and 10.24% under clean conditions and at 20dB, 10dB, 5dB, and 0dB SNRs, respectively, with an improvement of 4.88% in the average classification performance. As such, the improvement in the classification performance for the CITF compares favorably with the SAI features in [18]. However, results for feature vector combination and the training and evaluation times have not been reported in [18].

The training and evaluation time of the classifiers and features considered in this work are compared next.

4.4.2 Training and Evaluation Time of the Classifiers

The training and evaluation time of the SVM and DNN classifiers are given in Table V for the best performing feature set of linear GTCC + CITF. All training and evaluation times were measured using Intel Core i7–3632QM CPU. The training time of the DNN classifier is considerably higher than the SVM classifier, about 200 times more, which can be a disadvantage if performing unsupervised training. However, the evaluation time of the DNN classifier is determined to be significantly faster, about 596 times faster than the SVM classifier. Also, the DNN classifier always produced the highest overall classification performance and also the most noise robust. Therefore, if using supervised training, as in this work, the DNN classifier can be considered the best choice due to its superior classification performance and faster evaluation time. Besides, techniques such as the use of GPUs over CPUs have been proposed for faster training time for DNNs [1, 24]. Also, accelerated decision making in SER using parallel processing techniques, implemented on a supercomputing cluster, has been proposed in [16].

Table V: Training and evaluation time of the SVM and DNN classifiers for the best performing combined feature vector (linear GTCC + CITF)

Classification Method	Training Time (s)	Testing Time (s)
SVM	0.4512	33.7504
DNN	89.8556	0.0566

4.4.3 Training and Evaluation Time of Features with DNN

Finally, the training and evaluation time of the different features are computed. These are plotted in Fig. 2 for DNN classification. The training and evaluation time in this instance are largely affected by two variables, the feature vector dimension and the internal layer dimensions of the DNN classifier, both of which are given in Table 1. For example, all cepstral features and the RCIF have the same feature dimension of 72 and DNN internal layer dimensions of 50. As such, the training and evaluation time of these features are approximately same. In general, a good correlation is observed between the training and evaluation times.

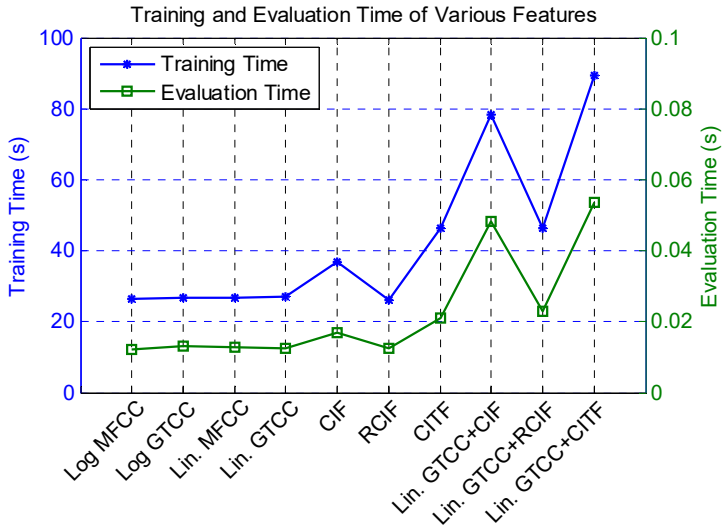

Figure 2: Training and evaluation time of various features and feature combinations with the DNN classifier

As far as the individual features are concerned, the cepstral features and the RCIF have the fastest training and evaluation times of about 26s and 12.5ms, respectively. With a training time of more than 40s and an evaluation time of approximately 20ms, the CITF, the best performing cochleagram feature, has the highest training and evaluation time of all the individual features. As such, the RCIF probably offers the best compromise between classification accuracy and the training and evaluation times.

The feature combination of linear GTCC with CIF and CITF coupled with 160 dimensional internal layers results in the highest training and evaluation times. However, due to relatively lower feature and layer dimensions, the training and evaluation time of linear GTCC + RCIF is relatively low, both at about half of linear GTCC + CITF. In addition, the average classification accuracy using this combination was determined to be 95.42%, only 0.64% lower than the feature combination of linear GTCC + CITF which gives the highest average classification performance. As such, the feature combination of linear GTCC + RCIF is a good alternative if lower computational costs are a priority.

5.0 Conclusion

This work proposes the use of DNNs over SVMs for robust sound classification in an audio surveillance application. When experimented with various individual features, the DNN classifier produced a better overall classification performance and was also seen to be more noise robust. With cepstral features, there wasn't any significant change in classification accuracy from SVM to DNN classification under clean conditions and at 20dB SNR. However, in general, marginal to significant increase in classification accuracy was observed at 10dB, 5dB, and 0dB SNRs. Only cochleagram based sound signal time-frequency representation was considered for feature extraction in this work and, with DNN classification, all cochleagram image derived features saw improvement in classification accuracy under all noise conditions with the most improved results at 0dB SNR.

The cochleagram image derived features were seen to be more noise robust than the cepstral features and with a better overall classification performance. The classification performance of the cochleagram image derived features was further improved with feature combination with linear GTCCs, the best performing cepstral feature. With feature combination, the classification performance of the DNN classifier was once again better than the SVM classifier. While the classification accuracy improved under all noise conditions, once again the most improved results were at 0dB SNR.

The DNN classifier was also seen to have a significantly faster evaluation time than the SVM classifier but the training time was observed to be significantly slower. As such, the only disadvantage of the DNN classifier over the SVM classifier was determined to be the slower training time. As future work, GPU could be used instead of CPU in a bid to reduce the training time. Also, feature vector dimension is one of the parameters affecting the training time and feature vector optimization could be considered for this purpose.

References

[1] J. Bergstra, O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, J. Turian, D. Warde-Farley, Y. Bengio, Theano: A CPU and GPU math expression compiler, in: Proceedings of the Python for Scientific Computing Conference (SciPy), Austin, TX, 2010, pp. 1-3.

[2] B.E. Boser, I.M. Guyon, V.N. Vapnik, A training algorithm for optimal margin classifiers, in: Proceedings of the fifth annual workshop on Computational learning theory, Pittsburgh, Pennsylvania, USA, 1992, pp. 144-152.

[3] C. Cortes, V. Vapnik, Support-Vector Networks, Machine Learning, 20 (3) (1995) 273-297.

[4] A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, Journal of the Royal Statistical Society, Series B (Methodological), 39 (1) (1977) 1-38.

[5] J. Dennis, Sound event recognition in unstructured environments using spectrogram image processing, Ph.D. Dissertation, Nanyang Technological University, 2014.

[6] J. Dennis, H.D. Tran, H. Li, Spectrogram image feature for sound event classification in mismatched conditions, IEEE Signal Processing Letters, 18 (2) (2011) 130-133.

[7] O. Gencoglu, T. Virtanen, H. Huttunen, Recognition of acoustic events using deep neural networks, in: Proceedings of the 22nd European Signal Processing Conference (EUSIPCO), 2014, pp. 506-510.

[8] B.R. Glasberg, B.C. Moore, Derivation of auditory filter shapes from notched-noise data, Hearing research, 47 (1-2) (1990) 103-138.

[9] D.D. Greenwood, A cochlear frequency-position function for several species - 29 years later, Journal of the Acoustical Society of America 87 (6) (1990) 2592-2605.

[10] R.M. Haralick, K. Shanmugam, I. Dinstein, Textural features for image classification, IEEE Transactions on Systems, Man and Cybernetics, SMC-3 (6) (1973) 610-621.
[11] G. Hinton, L. Deng, D. Yu, G.E. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T.N. Sainath, B. Kingsbury, Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups, IEEE Signal Processing Magazine, 29 (6) (2012) 82-97.
[12] G.E. Hinton, Training products of experts by minimizing contrastive divergence, Neural Computation, 14 (8) (2002) 1771-1800.
[13] G.E. Hinton, S. Osindero, Y.-W. Teh, A fast learning algorithm for deep belief nets, Neural Computation, 18 (7) (2006) 1527-1554.
[14] P. Khunarsal, C. Lursinsap, T. Raicharoen, Very short time environmental sound classification based on spectrogram pattern matching, Information Sciences, 243 (2013) 57-74.
[15] Z. Kons, O. Toledo-Ronen, Audio event classification using deep neural networks, in: INTERSPEECH, 2013, pp. 1482-1486.
[16] K. Lopatka, A. Czyzewski, Acceleration of decision making in sound event recognition employing supercomputing cluster, Information Sciences, 285 (2014) 223-236.
[17] L. Lu, H.-J. Zhang, S.Z. Li, Content-based audio classification and segmentation by using support vector machines, Multimedia Systems, 8 (6) (2003) 482-492.
[18] I. McLoughlin, H. Zhang, Z. Xie, Y. Song, W. Xiao, Robust Sound Event Classification Using Deep Neural Networks, IEEE/ACM Transactions on Audio, Speech, and Language Processing, 23 (3) (2015) 540-552.
[19] S. Nakamura, K. Hiyane, F. Asano, T. Nishiura, T. Yamada, Acoustical sound database in real environments for sound scene understanding and hands-free speech recognition, in: Proceedings of the 2nd International Conference on Language Resources and Evaluation (LREC 2000), Athens, Greece, 2000, pp. 965–968.
[20] R.B. Palm, Prediction as a candidate for learning deep hierarchical models of data, Technical University of Denmark, 2012.
[21] R.D. Patterson, K. Robinson, J. Holdsworth, D. McKeown, C. Zhang, M. Allerhand, Complex sounds and auditory images, in: Y. Cazals, L. Demany, K. Horner (Eds.) Auditory physiology and perception, Pergamon, Oxford, 1992, pp. 429-446.
[22] A. Rabaoui, M. Davy, S. Rossignol, N. Ellouze, Using one-class SVMs and wavelets for audio surveillance, IEEE Transactions on Information Forensics and Security, 3 (4) (2008) 763-775.
[23] M.M.A. Rahhal, Y. Bazi, H. AlHichri, N. Alajlan, F. Melgani, R.R. Yager, Deep learning approach for active classification of electrocardiogram signals, Information Sciences, 345 (2016) 340-354.
[24] R. Raina, A. Madhavan, A.Y. Ng, Large-scale deep unsupervised learning using graphics processors, in: Proceedings of the 26th Annual International Conference on Machine Learning, Montreal, Quebec, Canada, 2009, pp. 873-880.
[25] D.E. Rumelhart, G.E. Hinton, R.J. Williams, Learning representations by back-propagating errors, Nature, 323 (6088) (1986) 533-536.
[26] R.V. Sharan, T.J. Moir, Noise robust audio surveillance using reduced spectrogram image feature and one-against-all SVM, Neurocomputing, 158 (2015) 90-99.
[27] R.V. Sharan, T.J. Moir, Robust audio surveillance using spectrogram image texture feature, in: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brisbane, Australia, 2015, pp. 1956-1960.

[28] R.V. Sharan, T.J. Moir, Subband time-frequency image texture features for robust audio surveillance, IEEE Transactions on Information Forensics and Security, 10 (12) (2015) 2605-2615.
[29] R.V. Sharan, T.J. Moir, An overview of applications and advancements in automatic sound recognition, Neurocomputing, 200 (2016) 22-34.
[30] S.M. Siniscalchi, D. Yu, L. Deng, C.-H. Lee, Exploiting deep neural networks for detection-based speech recognition, Neurocomputing, 106 (2013) 148-157.
[31] M. Slaney, Lyon's Cochlear Model, Apple Computer, Technical Report 13, 1988.
[32] M. Slaney, An efficient implementation of the Patterson-Holdsworth auditory filter bank, Apple Computer, Technical Report 35, 1993.
[33] P. Smolensky, Information processing in dynamical systems: Foundations of harmony theory, in: D.E. Rumelhart, J.L. McClelland (Eds.) Parallel distributed processing, MIT Press, Cambridge, 1986, pp. 194-281.
[34] D. Turcsany, A. Bargiela, T. Maul, Local receptive field constrained deep networks, Information Sciences, 349–350 (2016) 229-247.
[35] V. Vapnik, A. Lerner, Pattern recognition using generalized portrait method, Automation and Remote Control, 24 (6) (1963) 774-780.
[36] V.N. Vapnik, Statistical learning theory, Wiley, New York, 1998.
[37] A. Varga, H.J.M. Steeneken, Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems, Speech Communication, 12 (3) (1993) 247-251.
[38] T.C. Walters, Auditory-based processing of communication sounds, University of Cambridge, 2011.