University of Kent

Kent Academic Repository

Xie, Zhipeng, McLoughlin, Ian, Zhang, Haomin, Song, Yan and Xiao, Wei (2016) *A new variance-based approach for discriminative feature extraction in machine hearing classification using spectrogram features.* Digital Signal Processing, 54 . pp. 119-128. ISSN 1051-2004.

Downloaded from

https://kar.kent.ac.uk/55016/ The University of Kent's Academic Repository KAR

The version of record is available from

https://doi.org/10.1016/j.dsp.2016.04.005

This document version

Author's Accepted Manuscript

DOI for this version

Licence for this version

CC BY-NC-ND (Attribution-NonCommercial-NoDerivatives)

Additional information

Versions of research works

Versions of Record

If this version is the version of record, it is the same as the published version available on the publisher's web site. Cite as the published version.

Author Accepted Manuscripts

If this document is identified as the Author Accepted Manuscript it is the version after peer review but before type setting, copy editing or publisher branding. Cite as Surname, Initial. (Year) 'Title of article'. To be published in *Title of Journal* , Volume and issue numbers [peer-reviewed accepted version]. Available at: DOI or URL (Accessed: date).

Enquiries

If you have questions about this document contact ResearchSupport@kent.ac.uk. Please include the URL of the record in KAR. If you believe that your, or a third party's rights have been compromised through this document please see our Take Down policy (available from https://www.kent.ac.uk/guides/kar-the-kent-academic-repository#policies).

© 2016. Licensed under the Creative Commons Licence https://creativecommons.org/licenses/by-nc-nd/4.0/

A new variance-based approach for discriminative feature extraction in machine hearing classification using spectrogram features

Zhipeng Xie[a], Ian McLoughlin[a,b,1], Haomin Zhang[a], Yan Song[a], Wei Xiao[c]

[a]*National Engineering Laboratory of Speech and Language Information Processing, The University of Science and Technology of China, Hefei, PRC.*
[b]*School of Computing, The University of Kent, Medway, UK.*
[c]*European Research Center, Huawei Technologies Duesseldorf GmbH, Munich, Germany.*

Abstract

Machine hearing is an emerging research field that is analogous to machine vision in that it aims to equip computers with the ability to hear and recognise a variety of sounds. It is a key enabler of natural human-computer speech interfacing, as well as in areas such as automated security surveillance, environmental monitoring, smart homes/buildings/cities. Recent advances in machine learning allow current systems to accurately recognise a diverse range of sounds under controlled conditions. However doing so in real-world noisy conditions remains a challenging task. Several front-end feature extraction methods have been used for machine hearing, employing speech recognition features like MFCC and PLP, as well as image-like features such as AIM and SIF. The best choice of feature is found to be dependent upon the noise environment and machine learning techniques used. Machine learning methods such as deep neural networks have been shown capable of inferring discriminative classification rules from less structured front-end features in related domains. In the machine hearing field, spectrogram image features have recently shown good performance for noise-corrupted classification using deep neural networks. However there are many methods of extracting features from spectrograms. This paper explores a novel data-driven feature extraction method that uses variance-based criteria to define spectral pooling of features from spectrograms. The proposed method, based on maximising the pooled spectral variance of foreground and background sound models, is shown to achieve very good performance for robust classification.

Keywords:
Machine hearing, auditory event detection, robust auditory classification

1. Introduction

Machine hearing is an emerging research topic [1] for automated computer understand of sound environments, just as machine vision is concerned with the automated visual understanding of image scenes. Machine hearing is important for future natural audio interfacing between

[1]Corresponding author email address: ivm@kent.ac.uk

© 2016. Licensed under the Creative Commons Licence https://creativecommons.org/licenses/by-nc-nd/4.0/

humans and computers, including speech-based interaction. It also has applications in areas such as security monitoring of homes and offices, automated surveillance of public spaces, environmental noise pollution and activity monitoring, and in enabling smart homes, buildings and cities. In smart cities, for example, a computer is able to infer events from audible information using audio sensors that are lower cost, require less networking bandwidth, consume less power, are potentially more robust and less easily obscured by weather or dust compared to video sensors. They also have the ability to sense non-line-of-sight events and are likely to enjoy a lower computation burden for automated processing. When deployed in a future smart city, a network of audio sensors could be used for city monitoring and management. These advantages also hold true for security monitoring. In terms of human-computer interfacing, machine hearing can allow devices to react contextually to sound, and is particularly important during speech interaction, as recognised by the PASCAL CHiME speech separation and recognition challenge [2].

Machine hearing comprises several research areas [1] including auditory event detection, separation, monitoring and classification, which operate according to different criteria. The current paper is concerned with the classification of auditory events, in particular the task of robust classification of noise-obscured sounds.

In fact, a myriad of sound event classification techniques have been published, ranging from parametric signal processing-based approaches [3, 4, 5] to automatic speech recognition (ASR) inspired methods [6]. Many of these make use of mel-frequency cepstral coefficients (MFCCs) [7] and similar features such as perceptual linear prediction (PLP) that are common in ASR. Biologically inspired (bio-mimetic) techniques have also been developed, with good performance for noise-free audio retrieval [8, 9], but simple spectrogram image features (SIF) have been shown to work well for robust classification [10, 11, 12, 13].

Powerful machine learning methods such as deep neural networks (DNN), when used in related domains, have been shown capable of inferring discriminative classification rules from less structured front-end features than those developed for less capable classifiers [14, 15, 16]. This includes recent methods for sound classification [17]. In fact the author's recent paper [10], found that simple SIF features and a DNN classifier outperformed auditory image models (AIM) and stabilised auditory images (SAI) as well as MFCCs and many other features [13].

1.1. Contribution

Although DNNs have been shown to perform well when classifying SIF features, the precise method of feature extraction between the SIF formation and the DNN input layer is relatively unexplored to date. Due to the high dimensionality of the SIF representation, SIF-based classifiers published to date all reduce the feature size in both spectral and temporal dimensions, usually through average (mean) pooling or max pooling [18]. The best performing pooling factors depend on the classifier, the nature of the underlying data and background noise, and are thus not known a priori. In this paper, several methods of extracting features from SIF are evaluated. However analysis of the sound data and noise leads to a variance-based criteria for spectral feature pooling. The aim is to steer the DNN classifier towards more discriminative spectral regions. The method requires only approximate spectral shape models of the underlying data, rather than a detailed spectrum and can be performed in a single step as opposed to the existing trial-and-error approach of running multiple classification experiments with different pooling parameters on development set data. The technique will be shown to yield a significant performance improvement over competing methods for noise robust classification. The method could also be applied to other classifiers and other front end feature extractors in related domains. The

© 2016. Licensed under the Creative Commons Licence https://creativecommons.org/licenses/by-nc-nd/4.0/

remainder of this paper is organised as follows. Section 2 discusses current classification methods in more detail. Section 3 details the DNN classifier used for all experiments while Section 4 discusses the SIF feature extraction framework and analyses the variance characteristics of SIF data as the basis for Section 5 to propose a variance-based pooling technique that is exploited in Section 6. Section 7 then analyses the performance of the technique while Section 8 concludes the paper.

2. Related Work

Earlier research in the sound event recognition field tended to use features borrowed from ASR, such as zero crossing rate, frame power energy, pitch and so on. These were used in conjunction with simple classifiers such as k-NN and decision trees to classify a small number of sounds in relatively noise-free conditions. These systems were generally efficient and obtained good performance on such limited tests.

However, in the era of big data, with the availability of large amounts of training data and the need to recognise more sound classes, machine learning techniques are required. Systems were thus developed which used more complex features such as MFCCs [3] and perceptual linear prediction (PLP) coefficients [19], allied with more powerful classifiers [20]. Given sufficient training material, these systems are often able to learn the non-obvious relationships between input data and output classes to yield good performance [21]. Popular techniques include support vector machine (SVM) [19, 22], Gaussian mixture models (GMMs) [23] and multi-layer perceptrons (MLP) [24]. Again, many of these research methods were driven by the success of techniques used for ASR, and generally performed well on larger classification tasks, but when tested with noise-corrupted sound, were found to perform poorly [13].

2.1. Bio-mimetic machine hearing

Inspired by the ability of humans to recognise sounds in noise, another approach adopted bio-mimetic (biologically-inspired) models of the human auditory system (HAS), such as AIM and SAI features, along with brain-inspired classifiers. Researchers, including Dennis et. al. [11] presented new time-frequency based feature extraction methods. "Spectrogram reading" [25], which was popular in the 1980s, provided evidence that spectrograms contain discriminative human-recognisable information, and more recent research has shown that it is also machine-recognisable. Unlike continuous speech or music, isolated sound events tend to be of short duration with a distinctive time-frequency signature. This motivated the current authors to develop a spectrogram image feature (SIF) [10] which, when classified by DNN, achieved state-of-the art performance in noisy conditions. The DNN is a powerful classifier which is widely used among computer vision and pattern recognition research communities. When provided with representative features, and given sufficient training data, the recognition performance of DNNs is often good, even with a large number of classes. In fact, indications are that well-trained DNNs are powerful enough to extract discriminative information from less structured features like the SIF, rather than more structured features such as SAI, which may perform better with a smaller numbers of classes, a less powerful classifier and reduced size training set. SIF features with a DNN classifier currently achieve state-of-the-art performance for robust classification [10], and have thus been adopted as a baseline technique for this paper.

© 2016. Licensed under the Creative Commons Licence https://creativecommons.org/licenses/by-nc-nd/4.0/

2.2. Robust classification

The noise-robust classification task differs from that for clean sounds, and requires different techniques. For example, figures reproduced in [10] show that MFCC features can achieve a 99.4% classification accuracy for clean sounds in 50 classes, compared to 98.9% for SIF features (albeit with different classifiers). However the addition of even modest amounts of noise (20dB signal to noise ratio, SNR[2]) degrades the MFCC results to 71.9% whereas SIF features still achieve 96.13%. In high levels of noise (0dB SNR), MFCC accuracy degrades much further to 15.7% compared to 85.47% for SIF features. Similar trends have been reported in [11], [13] and [26], and provide good evidence for the use of spectrogram features for robust sound event classification. The main disadvantage of the SIF is that it is much higher dimensionality than alternatives such as MFCCs and must be reduced substantially in dimension before DNN classification. Previous SIF-based classification by DNN [10] applied a simple average pooling-based downsampling to 24 frequency bins, and achieved good performance. The current paper extends upon this by evaluating a number of alternative feature extraction methods to suit the DNN classifier, and in particular develops a variance-based spectral pooling method that will be shown to achieve excellent performance, in Section 7. The method is potentially suitable for designing feature extractors for classifiers in any other domain where representative models of foreground (wanted sounds in this case) and background (noise) data are available.

2.3. Evaluation and comparison

A standard evaluation task allows sound event detection methods from different authors to be readily compared. In this paper we adopt a widely cited task and scoring method defined by Dennis et. al. [11] in which 50 sound classes are chosen from the RWCP (real world computing partnership) Sound Scene Database in real acoustic environments [27] and corrupted by background noise from the NOISEX-92 database. Full details of the sound and noise data, and the definition of training, test and development data sets will be given in Section 6.1.

2.4. Comparison with existing approaches

Several other published systems are compared using the standard evaluation task, with results from [11, 10] reproduced in Table 1. These include MFCC features with SVM classifier and hidden Markov model (HMM) back end, ETSI Advanced Front End (ETSI-AFE) toolkit enhancement (which uses noise removal techniques to significantly improve performance in noisy conditions), MPEG-7 method (57 features per frame, reduced to a dimensionality of 12 through principal component analysis (PCA) [28], and then augmented with difference and acceleration features in conjunction with a 5 state HMM having 6 Gaussian mixtures), and the Gabor method (a feature-finding single-layer perceptron network to select the best 36 features), all as reported in [13]. In the GTCC system, gammatone cepstral coefficients were extracted by 36 gammatone filters, then reduced to 12 dimensions using PCA before being augmented in the same way as in the MPEG-7 method. The MP+MFCC system used matching pursuit (MP) [29] to find the top five Gabor bases from a decomposition of the signal window, yielding four mean and variance features from the Gabor bases scale and frequency parameters. These were concatenated with MFCC features then augmented with deltas and accelerations to form the final feature vector. Dennis's own SIF extraction method ('Dennis SIF') was the first system capable

[2]All signal to noise ratio (SNR) computations in his paper are the ratio between each original entire RWCP sound file and the corresponding randomly-chosen section of background noise being added to it.

© 2016. Licensed under the Creative Commons Licence https://creativecommons.org/licenses/by-nc-nd/4.0/

Table 1: Classification accuracy for several state-of-the-art methods (all figures are courtesy of [13] and [10])

System	clean	20dB	10dB	0dB	mean
MFCC-HMM	99.4%	71.9%	42.3%	15.7%	57.4%
MFCC-SVM	98.5%	28.1%	7.0%	2.7%	34.1%
ETSI-AFE	99.1%	89.4%	71.7%	35.4%	73.9%
MPEG-7	97.9%	25.4%	8.5%	2.8%	33.6%
Gabor	99.8%	41.9%	10.8%	3.5%	39.0%
GTCC	99.5%	46.6%	13.4%	3.8%	40.8%
MP+MFCC	99.4%	78.4%	45.4%	10.5%	58.4%
Dennis SIF	91.1%	91.1%	90.7%	80.0%	88.5%
SIF-DNN-DN-v	98.9%	95.3%	92.4%	78.9%	91.4%
SIF-DNN-DN-e	96.0%	94.4%	93.5%	85.1%	92.3%

of achieving good performance in noise (note the excellent 80% classification accuracy for 0dB SNR in Table 1), while the DNN-based voting and e-scaled systems from [10] (SIF-DNN-DN-v and SIF-DNN-DN-e) currently achieve the highest overall mean performance for mismatched training conditions (i.e. systems trained and operated without a-priori noise information).

3. The Classifier

This paper is concerned with the feature extraction stage of a sound classifier, in particular adapting the feature space of a DNN based on the observed variance in foreground (wanted) data and background data (noise). The baseline for comparison is a DNN classifier and spectrogram image feature (SIF) derived from the configuration used in [10]. The SIF features and their evolution into context-sensitive feature spaces will be discussed in Section 4. Meanwhile, the single DNN classifier used for all feature evaluations in the remainder of this paper will be described in this section to ensure it is repeatable by other authors.

3.1. DNN classifier

An L-layer DNN classifier is constructed with the output layer in a one-of-K configuration (i.e. K classes), and the input layer fed with the feature vectors. The DNN is constructed from individual pre-trained restricted Boltzmann machine (RBM) pairs, each of which comprise V visible and H hidden stochastic nodes, $\mathbf{v} = [v_1, v_2, ..., v_V]^\top$, and $\mathbf{h} = [h_1, h_2, ..., h_H]^\top$. Two different RBM structures are used in this paper. Intermediate and final layers are Bernoulli-Bernoulli, whereas the DNN input layer is formed from a Gaussian-Bernoulli RBM. In the former, nodes are assumed to be binary (i.e. $\mathbf{v}_{bb} \in \{0,1\}^V$ and $\mathbf{h}_{bb} \in \{0,1\}^H$), and the energy function of the state $E_{bb}(\mathbf{v}, \mathbf{h})$ is therefore:

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

w_{ji} represents the weight between the ith visible unit and the jth hidden unit and b_i^v and b_j^h are respective real-valued biases. Bernoulli-Bernoulli RBM model parameters are $\theta_{bb} = \{\mathbf{W}, \mathbf{b}^\mathbf{h}, \mathbf{b}^\mathbf{v}\}$, with weight matrix $\mathbf{W} = \{w_{ij}\}_{V\times H}$ and biases $\mathbf{b}^\mathbf{h} = [b_1^h, b_2^h, ..., b_H^h]^\top$ and $\mathbf{b}^\mathbf{v} = [b_1^v, b_2^v, ..., b_V^v]^\top$.

© 2016. Licensed under the Creative Commons Licence https://creativecommons.org/licenses/by-nc-nd/4.0/

The Gaussian-Bernoulli RBM visible nodes are real (i.e. $\mathbf{v}_{gb} \in R^V$), while the hidden nodes are binary (i.e. $\mathbf{h}_{gb} \in \{0,1\}^H$). Thus, the energy function becomes:

$$E_{gb}(\mathbf{v},\mathbf{h}) = -\sum_{i=1}^{V}\sum_{j=1}^{H}\frac{v_i}{\sigma_i}h_j w_{ji} + \sum_{i=1}^{V}\frac{(v_i - b_i^v)^2}{2\sigma_i^2} - \sum_{j=1}^{H}h_j b_j^h \tag{2}$$

Every visible unit v_i adds a parabolic offset to the energy function, governed by σ_i. Gaussian-Bernoulli RBM model parameters thus contain an extra term, $\theta_{gb} = \{\mathbf{W}, \mathbf{b^h}, \mathbf{b^v}, \sigma^2\}$, with variance parameter σ_i^2 pre-determined rather than learnt from training data.

Given an energy function $E(\mathbf{v},\mathbf{h})$ defined as in either eqn. (1) or eqn. (2), the joint probability associated with configuration $(\mathbf{v},\mathbf{h})$ is defined as $p(\mathbf{v},\mathbf{h};\theta) = Z^{-1}exp\{-E(\mathbf{v},\mathbf{h};\theta)\}$ where Z is a partition function, $Z = \sum_{\mathbf{v}}\sum_{\mathbf{h}} e^{\{-E(\mathbf{v},\mathbf{h};\theta)\}}$.

3.1.1. Pre-training

Given a training set, RBM model parameters θ can be estimated by maximum likelihood learning using the contrastive divergence (CD) algorithm [30]. This runs through a limited number of steps in a Gibbs Markov chain to update hidden nodes $\mathbf{h}$ given visible nodes $\mathbf{v}$ and then update $\mathbf{v}$ given the previously updated $\mathbf{h}$. The input layer is trained first (i.e. the layer 1 $\mathbf{v}_{gb}$ input is the feature vector $\mathbf{v}$ from Section 4.1). After training, the inferred states of its hidden units $\mathbf{h_1}$ become the visible data for training the next RBM visible units $\mathbf{v_2}$. The process repeats to produce multiple trained layers of RBMs. Once complete, the RBMs are stacked to produce the DNN, as shown in Fig. 1.

3.1.2. Fine-tuning

A size K softmax output labelling layer is then added to the pre-trained stack of RBMs [31]. The function of the layer is to convert a number of Bernoulli distributed units in the final layer, $\mathbf{h_L}$, using a softmax function.

Back propagation (BP) is then used to train the stacked network, including the softmax class layer, based on minimising the cross entropy error between the true class label, c and the class predicted by the softmax layer. The cross-entropy cost function for class k, C, is easily computed as $-\sum_{k=1}^{K} c_k \log p(k|\mathbf{h};\theta_L)$ where θ_L represents the model parameters for the entire DNN.

During training, dropout was maintained at 0.1 (unless otherwise noted), mini-batch training size was set to 100, 1000 training epochs used and a sigmoid output function used. No significant effort is made to optimise these values until Section 7.2 since the aim of the experiment is to optimise features rather than classifier. All DNNs used in this paper were pre-trained and fine-tuned exclusively with noise-free sound features (i.e. so-called mismatched noise conditions).

4. Preliminary feature extraction

4.1. SIF features

A spectrogram is formed from a stack of fast Fourier transform (FFT) magnitude spectra. Given a sound vector s, a real valued spectral vector f is obtained from an FFT on frames of length w_s samples. For current frame F, spectral magnitude vector f_F is thus obtained as follows:

$$s_F(n) = s(F\delta + n)w(n) \qquad for\ n = 0 \ldots (w_s - 1) \tag{3}$$

© 2016. Licensed under the Creative Commons Licence https://creativecommons.org/licenses/by-nc-nd/4.0/

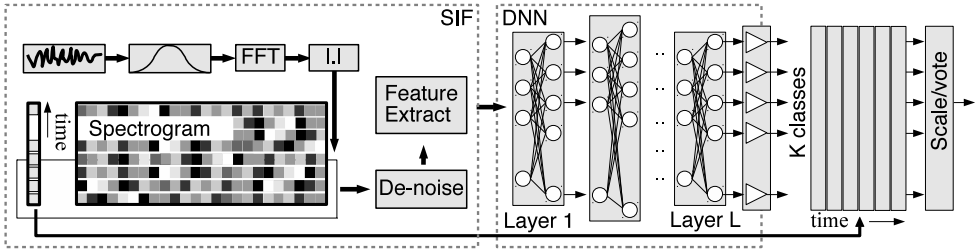

Figure 1: Diagram showing detail of SIF formation and extraction of DNN feature vector.

$$f_F(k) = \left| \sum_{n=0}^{w_s-1} s_F(n) e^{-j2\pi nk/w_s} \right| \qquad for\ k = 1 \ldots (w_s/2 - 1) \tag{4}$$

δ is the sample step between analysis frames and $w(n)$ is a w_s-point Hamming window. Average-pooling is performed in the frequency domain to downsample to a B bin resolution, as described in [10] and the resulting spectra stacked to form an overlapped spectrogram $\mathcal{S}$,

$$\mathcal{S}(l, m) = \frac{1}{B} \sum_{n=Bl}^{B(l+1)} f_{F-m}(n) \qquad for\ l = 0 \ldots B/\delta \tag{5}$$

In practice, the spectrogram $\mathcal{S}$ contains a history of up to D consecutive spectral lines (i.e. $m = 0 \ldots D-1$) which are concatenated to populate a $(BD + 1)$ dimension feature vector V which is augmented by a scalar energy metric. Feature vector $\mathbf{v}$ comprises elements $v(i) = \mathcal{S}(\lfloor i/B \rfloor, i - B\lfloor i/B \rfloor)$ for $i = 0 \ldots (BD - 1)$ with the energy metric computed as,

$$v(BD) = \sum_{l=0}^{D-1} \sum_{m=0}^{B-1} \mathcal{S}(l, m) \tag{6}$$

v is designed to capture information regarding frame energy because this is a significant factor in classification of even noisy sounds. This value is also used as a scaling for the DNN frame output classification used later, described as *energy scaling*. The feature vector $\mathbf{v}$, with a dimensionality of only $(BD + 1)$ constitutes the DNN input, and thus defines its input layer size.

When de-noising (DN) is applied to the SIF test features, this is achieved by computing a minimum energy frequency vector and subtracting it from all frequencies in the spectrogram prior to forming the frequency matrix. The de-noised spectrogram $\mathcal{S}_{dn}$ is thus,

$$\mathcal{S}_{dn}(l, m) = \mathcal{S}(l, m) - \min_l(\mathcal{S}(l, m)) \qquad for \quad m = 0 \ldots (B - 1) \tag{7}$$

Note that training, by contrast, is always performed using noise-free sounds. The initial BD elements of the final feature vector $\mathbf{v}$, are then formed from $\mathcal{S}_{dn}$, rather than $\mathcal{S}$, however the energy metric $v(BD)$ is computed from original spectrogram data as per Eqn. (6).

4.2. Frequency selective SIF (FSN-SIF)

According to Dennis et. al. [11], the noises selected for testing are typical environmental noises characterised by predominantly low frequency energy. As an example, Fig. 2 plots the cumulative absolute sum of 1024 frequency bins across the entire RWCP "Ring001" sound

© 2016. Licensed under the Creative Commons Licence https://creativecommons.org/licenses/by-nc-nd/4.0/

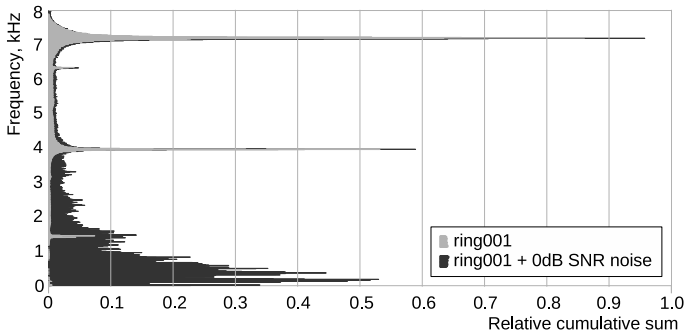

Figure 2: Cumulative absolute sum of frequency bins across a typical noise-free RWCP sound recording overlaid on the same sound corrupted by noise.

Table 2: Classification accuracy for different FSN-SIF frequency ranges using a DNN classifier.

s_f	e_f	0dB score	rel. to baseline
Baseline system			
0 Hz	*8000 Hz*	*83.60%*	–
0 Hz	7608 Hz	78.67%	-5.90%
390 Hz	7608 Hz	82.07%	-1.83%
390 Hz	8000 Hz	85.40%	2.15%
546 Hz	8000 Hz	83.67%	0.08%
999 Hz	8000 Hz	83.00%	-0.71%

recording overlaid on a plot of the same sound corrupted by NOISEX-92 "Destroyer Control Room" noise. Clearly, low frequency regions contain predominantly noise whilst the discriminating features of the sound predominantly lie at higher frequencies. In fact this holds true for many of the RWCP sounds, and may equally be true of spurious environmental sounds in general. Given these two observations relating to the noise and the sound distributions, a reasonable hypothesis is that the higher frequency regions provide more discriminative information for classification of noise-corrupted sounds than do the low frequency regions.

4.3. Frequency selectivity performance

To provide a simple test of this hypothesis, we construct an experiment in which several differently sized and located frequency regions are selected from the spectrogram and evaluated for classification performance. Frequency-pairs s_f and e_f, denoting start and end boundaries, are chosen arbitrarily to provide snapshots of the effect of including or excluding different frequency regions in the classification. Downsampling is performed as normal using the trimmed regions into the fixed B bin frequency resolution of Eqn. 5, with a DNN input dimension of $BD + 1 = 24{\times}30 + 1 = 721$ and 512 nodes in each hidden layer. A shift of $\delta = 16$ is used at sample rate $f_s = 16\,kHz$. Since the inspiration for this experiment is whether it is possible to weaken the effect of noise on the classification, the test is performed at 0dB SNR. Several indicative results are shown in Table 2 and compared to the baseline performance (i.e. using the full frequency range). The results show that a small improvement is achievable at 0dB against baseline SIF-DNN results simply by excluding lower frequencies, but excluding a similarly sized

© 2016. Licensed under the Creative Commons Licence https://creativecommons.org/licenses/by-nc-nd/4.0/

Table 3: Classification accuracy for different FSN-SIF frequency resolutions.

$(B \times D)$	0dB score	rel. to baseline
24×30	85.40%	–
30×30	88.47%	3.59%
40×30	81.40%	-4.68%
64×30	75.13%	-12.03%

high frequency region is detrimental. The best performing 390 to 8000Hz frequency pair with this downsampling arrangement will be retained (referred to as FSN-SIF) for future comparison against the proposed non-linear frequency mapping approach.

4.4. Frequency bins

The interaction between DNN classification performance and frequency range may also be explored in an alternative way by adjusting the DNN frequency resolution. We thus test different size frequency bins, selecting four different values of B from Eqn. 5 for FSN-SIF. Results are shown in Table 3. It is clear that a further slight improvement has been unlocked by the exclusion of lower frequencies from the input spectrogram (a 30×30 window does not improve performance when the full frequency span is used). Taken together, these results demonstrate that different frequency domain regions yield unequal contributions to class discrimination – even considering the fact that the frequency resolution at the DNN is very low (for example ranging from just 24 to 64 bins in these experiments). This evidence provides a degree of confidence to inspire further analysis and experimentation on frequency selectivity for robust machine hearing in the following section – particularly in designing a data-driven method for deciding upon span and resolution without requiring trial and error evaluations. In the remainder of this paper, unless where specifically noted, the SIF resolution is fixed at (30×30).

5. Proposed feature selection method

In general, to reduce the dimensionality of input features for a DNN, approaches such as downsampling or pooling are applied. These are common in computer vision and pattern recognition fields. Both Dennis et. al. [12] and the authors' [10] used downsampling to reduce the dimensions of spectrogram features in order to increase DNN efficiency and improve performance, although in different ways. Dennis used 9×9 blocks to pool the spectrogram into smaller regions. Meanwhile, linear average pooling was used in [10] to reduce the number of frequency bins (as discussed in Section 4.1), as well as explore different context sizes or time spans.

In fact endless adjustments and refinements are possible in the downsampling process, and these are evidently sensitive to the deployment scenario (i.e. the type of sounds and noises being used). This paper, rather than defining fixed frequency regions and performing linear downsampling, presents a non-linear downsampling method determined by models of the underlying noise and sound characteristics. Allowing a-priori knowledge, these characteristics can be extracted from the test scenario, however this paper disallows direct a-priori knowledge and instead uses a development data set to build low-order models of average sound and noise characteristics. These models are then used to exploit the performance gains achieved though both the frequency region selection technique and the frequency bin resolution adjustment technique explored in Section

© 2016. Licensed under the Creative Commons Licence https://creativecommons.org/licenses/by-nc-nd/4.0/

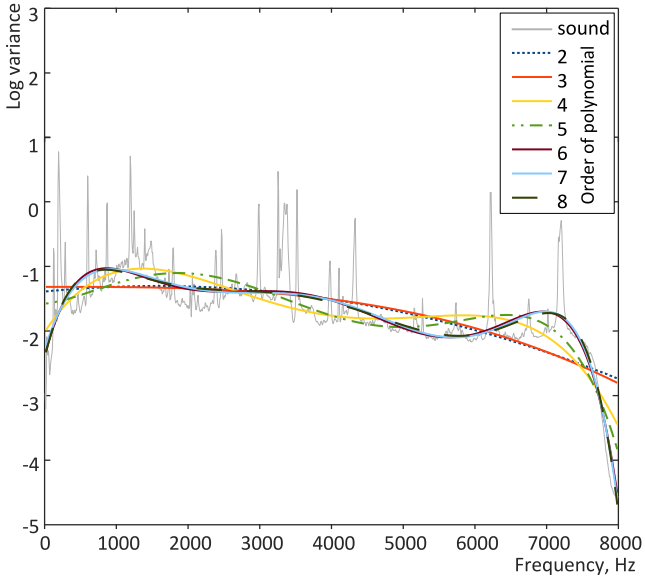

Figure 3: Per-frequency variance of all sounds along with several orders of polynomial fit to the data.

4. The difference is that we will accomplish this in a single step without requiring multiple experimental runs of the classifier with different parameters.

The basic idea is to estimate the frequency-selective discriminative power of sound features, as well as the frequency-selective masking effect of background noise, in terms of spectral power variance over time. Noise frequencies with larger power variance are likely to be greater sources of confusion than those frequencies that exhibit smaller variance. Conversely, sound frequencies with larger power variance (measured across all classes) are likely to be more discriminative than those with smaller variance. Then frequency regions with lower noise variance and higher sound variance are probably most discriminative, by contrast with those having high noise variance and low sound variance which will probably be less discriminative or masked. In general, the former should be emphasised at the expense of the latter.

5.1. Sound and noise variance

Disallowing a-priori information, variance models are constructed from polynomials fit to the development set data. Firstly we calculate the cumulative sum in the frequency domain of all files, normalising the results by dividing by the length of each file. Then obtain the frequency distribution by summing the files in the same class, again normalising by dividing by the number of files per class, 50. Working in the frequency domain, the result is thus a matrix of size $50 \times B$.

The standard deviation σ_s is then computed for each frequency bin across all classes, to obtain the intra-class variance:

$$\sigma_s = \left[\frac{1}{N} \sum_{i=1}^{N} \left(B_i^d - \bar{B}^d \right)^2 \right]^{\frac{1}{2}} \qquad for \quad d = 1 \ldots B \tag{8}$$

where N refers to the whole number of classes in the sound development set, d denotes the dth spectral index (frequency bin). $\bar{B}^d$ is then the mean for all classes of sound frequency d,

© 2016. Licensed under the Creative Commons Licence https://creativecommons.org/licenses/by-nc-nd/4.0/

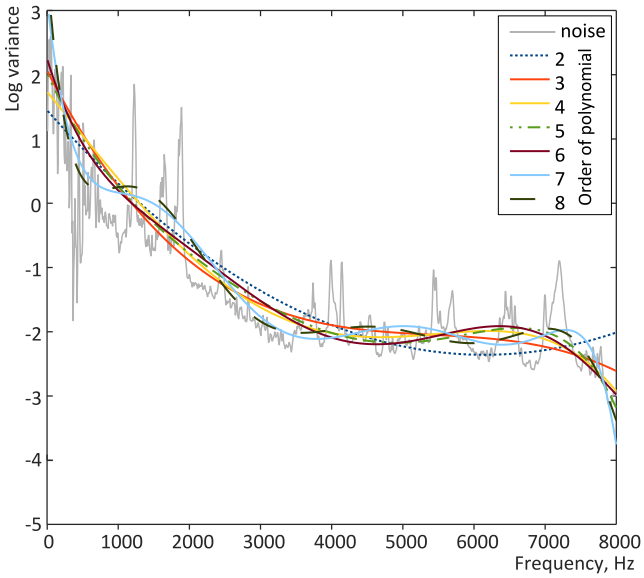

Figure 4: Per-frequency variance of all noise recordings in the training, test and development sets along with several orders of polynomial fit.

calculated as:

$$\bar{B}^d = \frac{1}{N}\sum_{i=1}^{N} B_i^d \qquad for \quad d = 1 \ldots B \tag{9}$$

However, σ_s is a high frequency resolution response that is specific to the development set data. To remove the possibility of a tight dependency between actual frequency shape and pooling performance, the frequency response is instead represented by a low-order model. Spectral shape models can be derived in a variety of ways from underlying data, for example though cepstral or linear prediction approaches [32], or from physical models of the underlying processes, this paper will use the most basic approach of a polynomial $p(x)$ of degree n, fit to the spectral response in a least squares sense:

$$p(x) = p_1 x^n + p_2 x^{n-1} + \ldots + p_n x + p_{n+1} \tag{10}$$

The result, p is a row vector of length $n+1$ containing the polynomial coefficients in descending powers. To visualise this process, Fig. 3 plots the per-frequency variance of all sounds, along with several orders of polynomial variance models, p_s, constructed from the data. The precise choice of model order for p_s is discussed in Section 6.2.

This process is repeated to obtain the frequency variance of noise, obtained from the noise development set. First we calculate the frequency distribution of each noise and normalise it by dividing by the length of the noise file. Then we repeat Eqns. 9 and 8 to obtain the noise standard deviation (σ_n). This is visualised in Fig. 4, again with a number of polynomial fits to the data. As expected the noise variance is much higher at low frequencies than the sound variance was.

© 2016. Licensed under the Creative Commons Licence https://creativecommons.org/licenses/by-nc-nd/4.0/

5.2. Overlap and region selection

As mentioned previously, the ideal regions which promise good discriminative performance with lower noise influence should be those that have large sound variance and low noise variance. To achieve this, we introduce a non-linear mapping method pools the spectral features based on (σ_n) and (σ_s).

Once the sound and noise variance models have been determined, we calculate the positive area that is the intersection between the model curves, and then divide it by slicing vertically into B equal area slices (i.e. where slice defines the pooling used to feed one DNN input node). The slices are partitioned by $B+1$ frequency boundaries (i.e. to include upper and lower boundaries). All of the FFT frequency bins lying between two neighbouring boundaries are averaged to form a single element of the resulting length B feature vector, without overlap. In this way, the area of each slice is calculated as:

$$\beta = \frac{(\mathcal{M}_S - \mathcal{M}_N) \times [(\mathcal{M}_S - \mathcal{M}_N) > 0]}{B} \tag{11}$$

where $\mathcal{M}_S$ is the model curve generated from the polynomial fit of σ_s, representing the variance the sound development set with frequency. $\mathcal{M}_N$ is similarly the model of the noise variance.

In practice, the variance of sound is actually much lower on average than that for noise. We therefore introduce an offset, κ to adjust the gap between them, which serves as a method to vary the region-selection emphasis given to $\mathcal{M}_S$ compared to that given to $\mathcal{M}_N$. The revised function is shown below:

$$\beta = \frac{[(\mathcal{M}_S + \kappa) - \mathcal{M}_N] \times [((\mathcal{M}_S + \kappa) - \mathcal{M}_N) > 0]}{B} \tag{12}$$

Fig. 5 shows a block diagram of this process, where the variance of noise and sound development sets are calculated, modelled by polynomial, the intersection computed and then split into equal-area regions. When constructing the DNN input feature vector, each of the equal-area regions contributes its own frequency bins to one DNN input node. The method aims to ensure that regions of interest (i.e. those with sound variance greater than noise variance) are rewarded with a higher frequency resolution. In fact, this allows the feature vector elements to concentrate on areas where the variance difference between sound and noise is larger. The effect is that every DNN input node carries approximately the same degree of contribution to the overall sound variance, with respect to noise variance. If variance can predict the discriminative power of the DNN, then that predictive power is maximised, as well as spread much more evenly across the DNN input layer. It also avoids the situation where some DNN input nodes contribute much more to the overall discriminative power of the network than other nodes do. We call this non-linear mapping technique applied to the SIF the NLM-SIF feature extraction method. This process combines both the nonlinear mapping and region selection approaches, both now being driven by models of the data.

6. System Design

After detailing the evaluation task and methodology, this section discusses the criteria for NLM-SIF parameters, before outlining the set-up of the DNN. Finally, the structures, sizes and parameters of the proposed systems will be presented.

© 2016. Licensed under the Creative Commons Licence https://creativecommons.org/licenses/by-nc-nd/4.0/

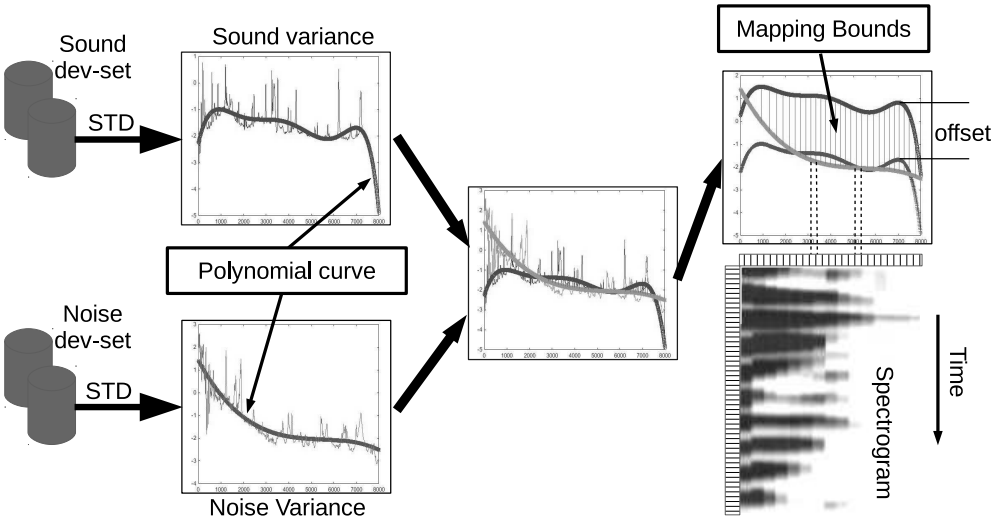

Figure 5: Block diagram of the non-linear mapping SIF method (NLM-SIF)

6.1. Standard evaluation task

In the RWCP database, every class contains more than 80 recordings, many have up to 100 recordings. Each recording contains a single example sound, captured with high SNR and having both lead-in and lead-out silence sections. As in [12], the training data set comprises 50 randomly-selected files from each of the 50 sound classes. A further 30 files are also randomly selected from each class and set aside for evaluation (testing set). Therefore, a total of 2500 files are available for training and 1500 for testing, per experimental run. All evaluations in this paper use classifiers trained with exclusively clean sounds, without pre-processing or noise removal. In all cases, evaluation is performed separately for original sound recordings ("clean"), as well as sounds corrupted by additive noise. The noise-corrupted tests use four background noise environments selected from the NOISEX-92 database (again, we confine the selection to those used in [12, 10], namely "Destroyer Control Room", "Speech Babble", "Factory Floor 1" and "Jet Cockpit 1"). These environments were chosen originally by Dennis [13] as realistic examples of non-stationary noise with predominantly low-frequency components. All sound files used in the experiments are stored at 16 kHz sample rate in 16-bit mono uncompressed PCM format, and range in duration from approximately 0.4 s to 1.9 s but are typically less than 1 s long. During evaluation under noisy conditions, noise is added to the test data set at levels of 20, 10 and 0 dB SNR. For each file in the test data set, one of the four NOISEX-92 recordings is randomly selected, a random starting point identified within the noise file, and then sample-wise added, at the given SNR, to the sound file. SNR is calculated over the entire noise and sound file in each case.

The current paper additionally gathers the remaining files in each class to form a sound development or validation set. This is used to tune the parameters of the system and obtain extra information about the sounds classes. 978 files in total are reserved for the sound development set, comprising 48 classes of 20 files with two classes having only 10 and 8 files in each, limited by lack of source data. Similarly, we define a noise development set, again used for system tuning and for noise modelling. These are selected from the unused files in the NOISEX-92 database,

© 2016. Licensed under the Creative Commons Licence https://creativecommons.org/licenses/by-nc-nd/4.0/

using the same criteria as used for the testing noise dataset: having energy mainly concentrated in the low frequency region. The development noise set comprises "Destroyer engine room noise", "Tank noise", "Factory floor noise 2" and "Military vehicle noise" recordings.

Five fold cross validation is used for all experiments (apart from the exploration of offset in Fig. 6), meaning that different combinations of files are used for training, testing and development – leading to five different models and five different sets of experimental results for each tested condition. Results are then tabulated in terms of the mean score and standard deviation.

Note that previous DNN classifiers achieved good performance with two different rules to determine final output classification from the multiple analysis frames spanning a single sound [10]. We similarly maintain these two rules during evaluation: **voting** (denoted by -v) means that the result of classification is based on vote share from the individual winning analysis frames across a (variable length) sound recording. Alternatively **e-scaled** voting (denoted by -e) weights the voting power of each analysis frame by the energy of the frame, to emphasise votes from higher energy regions against those from lower energy regions.

6.2. Parameters in modelling NLM-SIF

Polynomial curves are used to model the shape of the sound and noise variance described in Section 5. Higher order polynomial can fit the underlying variance response very well, including detail of peaks and valleys, but may be too specific (i.e. following the shape of the development set files too closely). Lower orders can describe the tendency of lines more generally, and are possibly better able to capture the generic frequency variance of the sounds and noise. To explore this further, Figs. 3 and 4 plotted the log variance of the sound and noise data respectively. The responses are quite spiky, but different underlying trends are visible. Polynomial models of order 2 to 8 are overlaid on the plots.

For noise variance, we prefer a common and representative shape for general classification. So we choose much lower order than for the sounds. A first order model represents the average variance, whereas second order can select either a central minima or central maxima whereas we can see from Fig. 4 that the underlying shape is more complex. Third order modelling allows a separate maxima/minima in high and low regions, and appears to be a reasonable initial choice. The sound model, by contrast would reasonably be expected to benefit from a more specific shape and thus we will adopt a degree of 7 for initial experiments. The performance of different polynomial degrees for both noise and sound will be explored fully in Section 7.1.

6.3. DNN size and structure

Given a time resolution of D, the DNN input feature vector thus has size $BD + 1$ (i.e. the down sampled spectrogram plus energy). This feeds a DNN with 2 hidden layers each of size 512, and a dimension 50 output layer, determined by the number of sound classes in the RWCP-based evaluation. The frequency resolution is increased slightly to 30 over [10], according to the experiments discussed above. The dimension of the input layer is thus $30 \times 30 + 1 = 901$, with the 1 representing the energy of the current frame. Table 4 presents performance figures for clean, 20, 10 and 0dB SNR and mean values for four different systems having 210 or 512 hidden nodes and frequency resolutions of 24 and 30. Values of 210 and 24 respectively were found optimal in [10], but we see here that a system with 512 hidden nodes and a frequency resolution of 30 performs slightly better in noise, and identically in clean conditions.

The final system parameters are listed in Table 5 along with the original baseline SIF-DNN system and the FSN-SIF system. As can be seen, the classifier for the final NLM-SIF system is $901 - 512 - 512 - 50$. All other parameters and evaluation settings are as in [10].

© 2016. Licensed under the Creative Commons Licence https://creativecommons.org/licenses/by-nc-nd/4.0/

Table 4: Performance of different frequency resolution and hidden nodes

System	clean	20dB	10dB	0dB	mean
210_24×30	96.00%	94.37%	93.53%	85.13%	92.26%
210_30×30	96.60%	95.80%	94.33%	85.33%	93.02%
512_24×30	95.47%	94.53%	94.00%	83.60%	91.90%
512_30×30	96.60%	95.53%	94.60%	87.13%	93.47%

Table 5: Final system parameters for evaluation

Classifier	DNN		
Features	SIF-DNN	FSN-SIF	NLM-SIF
Freq. resolution, B	24	30	30
Time resolution, D	30	30	30
Analysis window	128ms	128ms	128ms
Feature dimension	721	901	901
Hidden layers, L	2	2	2
Hidden nodes, H	210	512	512

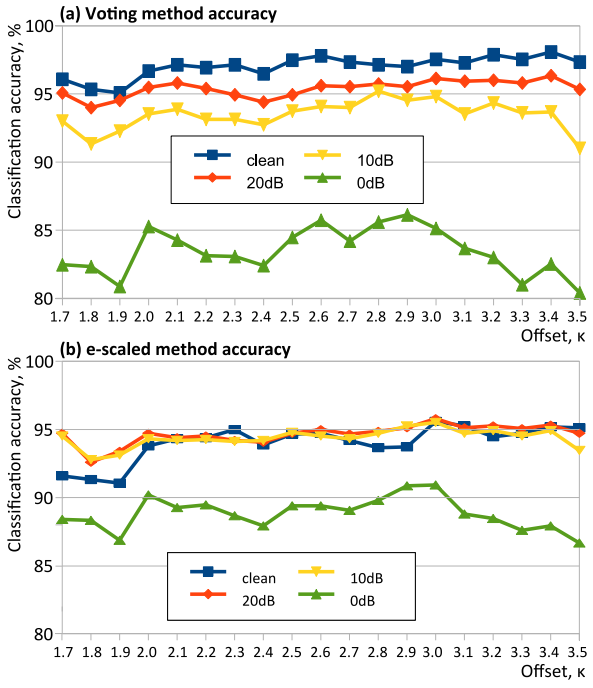

Figure 6: DNN classification accuracy for (a) voting and (b) e-scaled methods for NLM-SIF features against different degrees of offset.

© 2016. Licensed under the Creative Commons Licence https://creativecommons.org/licenses/by-nc-nd/4.0/

Table 6: Performance, in % (with standard deviation) of different order noise models and fixed sound order of 7.

order	clean	20dB	10dB	0dB	mean
Voting					
3	98.0 (0.4)	96.3 (1.9)	93.3 (2.4)	84.0 (2.2)	92.9 (1.7)
4	96.8 (0.6)	94.7 (1.5)	90.7 (2.8)	79.0 (2.7)	90.3 (1.9)
5	96.6 (1.0)	94.2 (2.1)	90.1 (3.6)	78.7 (3.0)	89.9 (2.4)
6	96.8 (0.9)	95.0 (1.6)	91.0 (2.4)	79.1 (3.9)	90.5 (2.2)
7	97.0 (0.3)	94.7 (1.4)	90.3 (3.3)	77.6 (3.2)	89.9 (2.1)
e-scaled					
3	95.0 (0.5)	94.3 (3.2)	93.3 (3.5)	88.6 (2.6)	92.8 (2.5)
4	94.6 (0.7)	93.2 (2.7)	91.3 (4.0)	84.5 (3.6)	90.9 (2.7)
5	94.0 (1.2)	92.8 (3.3)	90.9 (3.5)	83.8 (3.2)	90.4 (2.8)
6	94.6 (1.2)	93.4 (3.0)	91.5 (3.2)	84.5 (3.8)	91.0 (2.8)
7	94.6 (0.4)	93.7 (2.0)	91.9 (3.2)	84.2 (3.4)	91.1 (2.3)

6.4. Effect of offset

The region computation described in Eqn. 12 depends upon an offset which varies the relative contribution of the noise and sound variance models to the overall non-linear downsampling partitions. The noise variance is generally larger than the sound variance (this is visible in Figs. 3 and 4) and so the choice of optimal offset, κ is unclear. Thus we construct a series of experiments to investigate the effect of κ on overall classification performance. Order 3 and 7 variance models are again used for noise and sound respectively, but have been computed using the *test* data in this case to remove the issue of whether or not the development set is representative. For all other tests in this paper, only the development set data is used for modelling sound and noise. With this proviso, the results are shown in Fig. 6 for a range of offset, κ from 1.7 to 3.5. Both e-scaled and voting results are given, and results plotted for all noise conditions.

Although the result curves are not smooth, there is a shallow upward trend in performance from $\kappa = 1.7$, flattening out above about $\kappa = 3$.

7. Results and Discussion

Table 1 listed the classification accuracy of several machine hearing systems, including the baseline SIF-DNN-DN systems [10], which achieve 85.1% accuracy for the 0dB noise condition and 92.3% on average using an e-scaled voting criteria.

7.1. Exploring model complexity

This section explores the effect of model complexity for both the sound and noise models. With a sound model order of 7 and offset set to $\kappa = 3$, several orders of noise were investigated with the NLM-SIF system, with results listed in Table 6. Too high a noise order makes it become overly specific to the development set noise, and too low an order reduces the useful shape. Thus it is no surprise that a 3rd order noise model performs best - achieving 88.6% accuracy at 0dB, as well as a mean accuracy over all conditions of 92.8% (e-scaled results).

Without changing any other parameters, the noise order was then fixed to 3 and the experiment repeated to investigate the effect of several orders of sound model. Results are shown in

© 2016. Licensed under the Creative Commons Licence https://creativecommons.org/licenses/by-nc-nd/4.0/

Table 7: Performance, in % (with standard deviation) of different order sound models with fixed noise order of 3.

order	clean	20dB	10dB	0dB	mean
Voting					
3	97.1 (0.6)	94.6 (1.7)	90.3 (2.4)	77.9 (1.5)	90.0 (1.6)
4	97.4 (0.7)	95.1 (1.3)	90.9 (2.2)	80.3 (1.4)	90.9 (1.4)
5	97.3 (0.4)	94.7 (1.4)	91.6 (2.2)	81.3 (2.3)	91.2 (1.6)
6	96.7 (0.8)	94.1 (2.0)	90.3 (2.9)	78.6 (3.9)	89.9 (2.4)
7	97.1 (0.8)	95.0 (2.0)	91.5 (2.9)	81.8 (2.9)	91.3 (2.1)
8	96.7 (0.5)	94.3 (1.3)	90.0 (2.8)	77.7 (3.7)	89.7 (2.1)
e-scaled					
3	94.6 (0.9)	92.9 (3.4)	91.1 (3.3)	82.9 (2.6)	90.4 (2.6)
4	94.7 (0.4)	93.6 (2.4)	91.7 (3.1)	85.3 (2.0)	91.4 (2.0)
5	94.9 (0.6)	93.9 (2.0)	92.6 (2.7)	85.4 (2.6)	91.7 (2.0)
6	94.5 (0.7)	93.0 (2.8)	91.0 (3.7)	83.6 (4.4)	90.5 (2.9)
7	94.8 (0.8)	93.8 (2.6)	92.4 (3.4)	86.9 (3.0)	92.0 (2.5)
8	94.3 (0.8)	93.0 (2.5)	90.7 (3.8)	82.9 (3.8)	90.2 (2.7)

Table 8: Overall classification accuracy comparison of baseline (upper two) and proposed (lower two) methods, given as average percentage and (standard deviation).

system	clean	20dB	10dB	0dB	mean
SIF-DNN-v	97.8 (0.9)	94.8 (1.1)	90.8 (1.5)	75.1 (2.2)	89.6 (1.4)
SIF-DNN-e	95.5 (1.2)	93.9 (2.0)	92.8 (1.9)	82.9 (2.7)	91.3 (1.9)
NLM-SIF-v	98.0 (0.4)	96.3 (1.9)	93.3 (2.4)	84.0 (2.2)	92.9 (1.7)
NLM-SIF-e	95.0 (0.5)	94.3 (3.2)	93.3 (3.5)	88.6 (2.6)	92.8 (2.5)

Table 7. From this, it is clear that a 7th order sound model performs better using both the voting or e-scaled scoring, achieving up to 86.9% accuracy at 0dB and 92% overall.

Interestingly, it appears that the benefits achieved by optimising the noise model are slightly better than those achieved by optimising the sound model order.

7.2. Overall performance

From the experiments shown above, the proposed variance-based nonlinear mapping in NLM-SIF is capable of achieving good performance for noise corrupted sounds. Further experiments were conducted using the best performing NLM-SIF architecture, having sound and noise orders of 7 and 3 respectively, and $\kappa = 3$. The same DNN structure was used ($901 - 512 - 512 - 50$) with a nonlinearly down-sampled image size of 30×30. Other parameters were a minibatch of 100, dropout of 0.5, a softmax output function, and employing five fold cross-validation. Results are compared against the baseline SIF-DNN performance in Table 8.

In both SIF-DNN and NLM-SIF, there is a trade-off whereby voting criteria performs best with clean sounds, but energy scaling works better in the noise-corrupt case, confirming the findings noted in [10]. For the challenging 0dB noise evaluation, the proposed NLM-SIF method is able to achieve 88.6% accuracy – a significant improvement over the previous state-of-the-art performance. Overall accuracy has increased by about 1.6% to 92.9%.

© 2016. Licensed under the Creative Commons Licence https://creativecommons.org/licenses/by-nc-nd/4.0/

8. Conclusion

This paper has investigated the formation of DNN feature vectors for machine hearing classification from spectrograms. Working from a baseline of a state-of-the-art classifier which achieves the current best performance on a standard task of 50 RWCP sound classes in NOISEX-92 noise, we investigate the effect of frequency selectivity on the input spectrogram. This led into the formation of sound and noise models from a related sound and noise development dataset, and allowed the construction of equal-variance regions from these by pooling spectral features nonlinearly, with the motivation of ensuring that every DNN input node is able to contribute approximately equally to the discriminative task in noise.

Results demonstrate that different regions of the spectrogram frequency domain contribute discriminative capabilities non-linearly. Thus, using a non-linear mapping allows us to focus on different frequency regions to cater for different types of sound, resulting in better performance, especially in high-SNR noise conditions. Results using different model orders for noise and sound frequency regions show that, for the chosen standard evaluation task, the beneficial effect of modelling noise is slightly greater than that achieved by modelling the sounds.

The proposed variance-based method has been shown to require only approximate spectral shape information; a 7th order model for sounds and a 3rd order noise model are sufficient, and this could be obtained either from a theoretical understanding of the environment or from a single step analysis of representative noise and data. The alternative approach in current systems would be to undertake an exhaustive brute-force evaluation of different spectral pooling parameters using a development data set.

Several system parameters were explored and investigated in the current paper using the variance-based spectral pooling method, and a final system constructed that is able to achieve 88.6% accuracy for the challenging 0dB SNR sound classification task. A good accuracy of 98% is achieved for clean sounds.

Acknowledgment

This work was partly supported by the Huawei Innovation Research Program under Machine Hearing and Perception Project Contract No. YB2012120147. Yan Song was supported by the Natural Science Foundation of China (NSFC), grant no. 61172158.

Reference

[1] R. F. Lyon, Machine hearing: an emerging field, IEEE Signal Processing Magazine 27 (5) (2010) 131–139.
[2] J. Barker, E. Vincent, N. Ma, H. Christensen, P. Green, The pascal chime speech separation and recognition challenge, Computer Speech & Language 27 (3) (2013) 621–633.
[3] G. Guo, S. Z. Li, Content-based audio classification and retrieval by support vector machines, Neural Networks, IEEE Transactions on 14 (1) (2003) 209–215.
[4] C.-C. Lin, S.-H. Chen, T.-K. Truong, Y. Chang, Audio classification and categorization based on wavelets and support vector machine, Speech and Audio Processing, IEEE Transactions on 13 (5) (2005) 644–651.
[5] T. Heittola, A. Mesaros, T. Virtanen, A. Eronen, Sound event detection in multisource environments using source separation, in: Workshop on machine listening in Multisource Environments, 2011, pp. 36–40.
[6] L.-H. Cai, L. Lu, A. Hanjalic, H.-J. Zhang, A flexible framework for key audio effects detection and auditory context inference, IEEE Transactions on Audio, Speech, and Language Processing 14 (3) (2006) 1026–1039.
[7] G. Chechik, E. Ie, M. Rehn, S. Bengio, R. F. Lyon, Large scale content-based audio retrieval from text queries, in: ACM International Conference on Multimedia Information Retrieval (MIR), 2008, pp. 105–112.
[8] T. C. Walters, Auditory-based processing of communication sounds, Ph.D. thesis, University of Cambridge, Cambridge, UK (2011).

© 2016. Licensed under the Creative Commons Licence https://creativecommons.org/licenses/by-nc-nd/4.0/

[9] R. F. Lyon, M. Rehn, T. Walters, S. Bengio, G. Chechik, Audio classification for information retrieval using sparse features, uS Patent 8,463,719 (Jun. 11 2013).
[10] I. McLoughlin, H.-M. Zhang, Z.-P. Xie, Y. Song, W. Xiao, Robust sound event classification using deep neural networks, IEEE Transactions on Audio, Speech, and Language Processing 23 (2015) 540–552.
[11] J. Dennis, H. D. Tran, H. Li, Spectrogram image feature for sound event classification in mismatched conditions, Signal Processing Letters, IEEE 18 (2) (2011) 130–133.
[12] J. Dennis, H. D. Tran, E. S. Chng, Image feature representation of the subband power distribution for robust sound event classification, IEEE Transactions on Audio, Speech, and Language Processing 21 (2) (2013) 367–377.
[13] J. W. Dennis, Sound event recognition in unstructured environments using spectrogram image processing, Ph.D. thesis, Nanyang Technological University, Singapore (2014).
[14] A.-r. Mohamed, G. E. Dahl, G. Hinton, Acoustic modeling using deep belief networks, IEEE Transactions on Audio, Speech, and Language Processing 20 (1) (2012) 14–22.
[15] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, et al., Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups, IEEE Signal Processing Mag 29 (6) (2012) 82–97.
[16] N. Morgan, Deep and wide: Multiple layers in automatic speech recognition, IEEE Trans Audio Speech Lang Processing 20 (1) (2012) 7–13.
[17] Z. Huang, C. Weng, K. Li, Y.-C. Cheng, C.-H. Lee, Deep learning vector quantization for acoustic information retreival, in: Acoustics, Speech and Signal Processing, 2014. ICASSP 2014 Proceedings. 2014 IEEE International Conference on, IEEE, 2014, pp. 1364–1368.
[18] Y.-L. Boureau, J. Ponce, Y. LeCun, A theoretical analysis of feature pooling in visual recognition, in: Proceedings of the 27th international conference on machine learning (ICML-10), 2010, pp. 111–118.
[19] J. Portelo, M. Bugalho, I. Trancoso, J. Neto, A. Abad, A. Serralheiro, Non-speech audio event detection, in: Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on, IEEE, 2009, pp. 1973–1976.
[20] A. Plinge, R. Grzeszick, G. A. Fink, A bag-of-features approach to acoustic event detection, in: Acoustics, Speech and Signal Processing, 2014. ICASSP 2014 Proceedings. 2014 IEEE International Conference on, IEEE, 2014, pp. 3732–3736.
[21] J. Maxime, X. Alameda-Pineda, L. Girin, R. Horaud, Sound representation and classification benchmark for domestic robots, in: Robotics and Automation (ICRA), 2014 IEEE International Conference on, IEEE, 2014, pp. 6285–6292.
[22] H. Phan, M. Maass, R. Mazur, A. Mertins, Acoustic event detection and localization with regression forests, in: 15th Annual Conference of the International Speech Communication Association (INTERSPEECH 2014), Singapore, 2014, pp. 1–5.
[23] P. K. Atrey, M. Maddage, M. S. Kankanhalli, Audio based event detection for multimedia surveillance, in: Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on, Vol. 5, IEEE, 2006, pp. V–V.
[24] P. Sidiropoulos, V. Mezaris, I. Kompatsiaris, H. Meinedo, M. Bugalho, I. Trancoso, On the use of audio events for improving video scene segmentation, in: N. Adami, A. Cavallaro, R. Leonardi, P. Migliorati (Eds.), Analysis, Retrieval and Delivery of Multimedia Content, Vol. 158 of Lecture Notes in Electrical Engineering, Springer New York, 2013, pp. 3–19.
[25] V. W. Zue, R. A. Cole, Experiments on spectrogram reading, in: Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'79., Vol. 4, IEEE, 1979, pp. 116–119.
[26] H. Zhang, I. McLoughlin, Y. Song, Robust sound event recognition using convolutional neural networks, in: Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on, no. 2635, IEEE, 2015, pp. 559–563.
[27] S. Nakamura, K. Hiyane, F. Asano, T. Yamada, T. Endo, Data collection in real acoustical environments for sound scene understanding and hands-free speech recognition, in: EUROSPEECH, 1999, pp. 2255–2258.
[28] M. Casey, Mpeg-7 sound-recognition tools, IEEE Transactions on circuits and Systems for video Technology 11 (6) (2001) 737–747.
[29] S. Chu, S. Narayanan, C.-C. Kuo, Environmental sound recognition with time–frequency audio features, IEEE Transactions on Audio, Speech, and Language Processing 17 (6) (2009) 1142–1158.
[30] G. E. Hinton, S. Osindero, Y.-W. Teh, A fast learning algorithm for deep belief nets, Neural computation 18 (7) (2006) 1527–1554.
[31] R. B. Palm, Prediction as a candidate for learning deep hierarchical models of data, Master's thesis, Technical University of Denmark (2012).
[32] I. V. McLoughlin, Applied Speech and Audio Processing, Cambridge University Press, 2009.