PAPER

Acoustic Event Detection in Speech Overlapping Scenarios Based on High-Resolution Spectral Input and Deep Learning

Miquel ESPI[†a)], ***Nonmember***, **Masakiyo FUJIMOTO**[†b)], ***and*** **Tomohiro NAKATANI**[†c)], ***Members***

SUMMARY We present a method for recognition of acoustic events in conversation scenarios where speech usually overlaps with other acoustic events. While speech is usually considered the most informative acoustic event in a conversation scene, it does not always contain all the information. Non-speech events, such as a door knock, steps, or a keyboard typing can reveal aspects of the scene that speakers miss or avoid to mention. Moreover, being able to robustly detect these events could further support speech enhancement and recognition systems by providing useful information cues about the surrounding scenarios and noise. In acoustic event detection, state-of-the-art techniques are typically based on derived features (e.g. MFCC, or Mel-filter-banks) which have successfully parameterized the spectrogram of speech but reduce resolution and detail when we are targeting other kinds of events. In this paper, we propose a method that learns features in an unsupervised manner from high-resolution spectrogram patches (considering a patch as a certain number of consecutive frame features stacked together), and integrates within the deep neural network framework to detect and classify acoustic events. Superiority over both previous works in the field, and similar approaches based on derived features, has been assessed by statical measures and evaluation with CHIL2007 corpus, an annotated database of seminar recordings.

key words: *acoustic event detection/recognition, high-resolution feature, spectrogram patch, communication scene understanding*

1. Introduction

Most of the research efforts in conversation scene understanding have usually focused on speech. This is in part due to the assumption that speech is the most informative component of the acoustic signal. However, in real environments, complete understanding of the scene cannot be achieved only through speech. Non-speech acoustic signals can reveal aspects of the scene that would be ignored otherwise. Speakers typically assume the context is implied in the conversation and neglect or avoid mentioning certain information. This context could be the surroundings, the activity a speaker is undergoing, etc. Acoustic event detection (AED) aims at detecting and classifying these acoustic signals: actively or passively produced by humans (e.g. speech, laugh, steps, etc.) or other objects (air conditioning, machine sounds, etc.). The goal is to process a continuous acoustic signal and convert it into a sequence of event labels with associated start and end times. Rich transcription in speech communication [1], [2] and scene understanding [3], [4] benefit from it, but also informed speech enhancement and automatic speech recognition (ASR) systems could benefit from it as a source of information. Recent hands-free meeting analysis systems already include simple event detection components in order to differentiate speech from laughter [5], but achieving a richer acoustic event recognition could effectively support speech detection and informed speech enhancement [6] by providing detailed description of the surrounding noises, besides the obvious benefits of richer transcriptions. Moreover, AED can also be applied in a variety of areas, including surveillance [7], context-based indexing and retrieval in multimedia [8], [9], or health care [10]; and, at higher abstraction levels, in automatic tagging [11], and audio segmentation [12].

Conversation scenes are mainly populated with speech, which usually overlaps with other co-occurring sounds and that we want to detect and classify. AED has typically concentrated efforts in the recognition stage [13], [14], leaving the feature extraction stage to standard ASR features such as Mel-frequency cepstrum coefficients (MFCC) or Mel-filter bank features that tend to obtain broad characterizations but end up reducing detail that is critical to deal with overlapping signals. Mel-filter-bank's dimensional reduction and DCT compression make MFCCs and Mel-filter-banks dense and concentrated around specific channels, which ultimately causes the loss of detail. A high-resolution spectrogram has of course more detail and is more sparse, enabling the assumption that there is no sound overlap within time-frequency bins. This is also why high-resolution spectrograms are used in sparse-analysis based blind source separation applications [15], [16]. By looking at a simple example of a signal with two sounds overlapped (Fig. 1), one realizes that while most prominent properties of a sound in the Mel-filter-bank domain are diluted when overlapping with other sounds (Fig. 1.a), this is not the case with high-resolution spectrograms (Fig. 1.b), where properties are still identifiable even after mixing it with another sound.

Another factor that has to be taken into account is that different acoustic events have different temporal structures, and these are also different from speech. In a field where temporal structures have been traditionally modeled outside the feature domain, typically with an hidden Markov model (HMM) on top, frame-based features have dominated. However, a feature that also embeds temporal information can lead to better recognition, even if we still continue to use an HMM on top. By using a spectrogram patch, which

Manuscript received December 19, 2014.
Manuscript revised May 20, 2015.
Manuscript publicized June 23, 2015.

[†]The authors are with NTT Communication Science Laboratories, NTT Corporation, Kyoto-fu, 619–0237 Japan.
a) E-mail: espi.miquel@lab.ntt.co.jp
b) E-mail: fujimoto.masakiyo@lab.ntt.co.jp
c) E-mail: nakatani.tomohiro@lab.ntt.co.jp

DOI: 10.1587/transinf.2014EDP7430

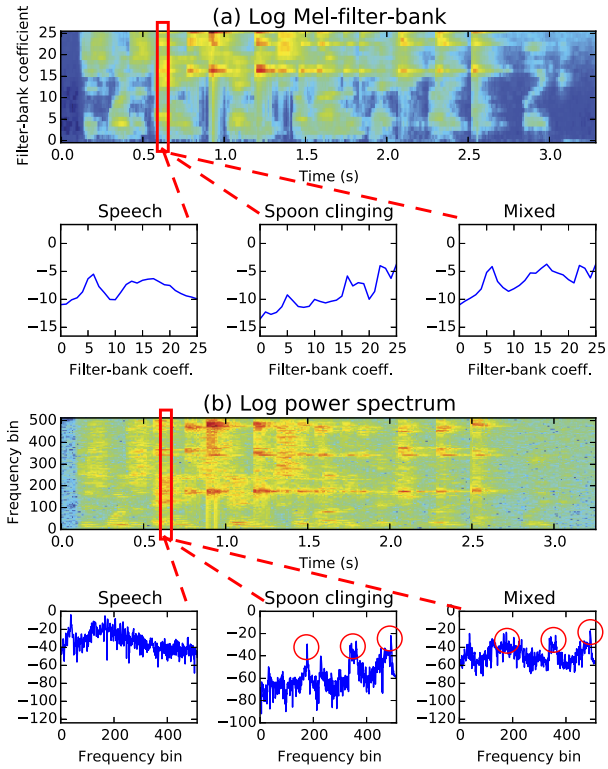

Fig. 1 Comparing a derived feature such as log Mel-filter-bank (a) and log-power spectrogram (b) for a the same signal, containing speech and a coffee spoon clinging and overlapped. Feature values for a specific instant where both sounds overlap are also shown for each of the acoustic events isolated and mixed to compare how informative each of the features is.

we define as a number of consecutive spectrogram frames stacked together, as the input feature we take a simple yet effective approach to embed short-time temporal structures in the feature. After all, a spectrogram patch can contain sufficient temporal structure detail if enough frames are included. Moreover, combined with the use of high-resolution spectrograms to generate the patches, we can assume that these would package enough time-frequency detail to model complex sounds.

Finally, to tackle the problem of modeling such a high-dimensional input feature, we need a model that can keep up with it. One of the reasons why high-resolution features have not been used until recently with state of the art classifiers was that before the arrival of deep learning, the standard classifier was Gaussian mixture model with hidden Markov models (GMM-HMM). GMMs excelled at modeling almost any distribution, however they are not as efficient when modeling high-dimensional data [17]. In GMMs, each component must generate all the features causing that as data has more dimensions, the number of patterns the GMM has to model on each feature segment grows too, and therefor the amount of parameters the model requires grows exponentially. That is why low resolution features (e.g. derived features) have dominated for so long. Deep neural networks (DNNs), on the other side, have flourished through recent years to become the standard discriminative

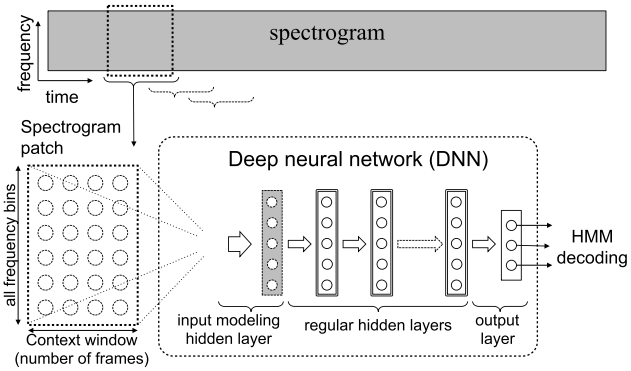

Fig. 2 Overview of the proposed spectrogram patch input model.

classifier also on AED related areas [18]. One of the advantages is that we have a model where the number of parameters needed grows linearly as data dimensionality grows. Mainly, because each component of the model is in charge a specific feature segment. Summarizing, it is because of DNNs that we are able to model the spectrogram directly as a feature. Restricted Boltzmann machines (RBMs) [19], unsupervised generative models with great high-dimensional modeling capabilities, allow us to learn features in an unsupervised manner from such a high dimensional input. Not only that, but given that RBMs are the base of current state-of-the-art DNNs [17] classification model. The learned feature extraction is integrated into the DNN framework seamlessly. Thus, resulting in a model that performs both feature extraction and classification at once, as we briefly introduced in a previous work [20]. Figure 2 shows an overview of the proposed model. In this paper we have considered two kinds of inputs: log-power-spectra patches, which can be dealt with current RBMs; and power-spectra patches, for which we also introduce exponential unit RBMs.

The remainder of this paper relates this research with other works in Sect. 2, introducing the spectrogram patch modeling and the acoustic modeling in Sect. 3, and a visual analysis of the training process in Sect. 4. Experimental evaluation, described in Sect. 5, confirmed that spectrogram patch models outperform those based on derived features, completing the paper with concluding remarks and future work in Sect. 6.

2. Prior Art and Contributions

Recent developments using spectrogram part decomposition approach the overlapping events problem by-passing features derived from acoustic spectra, and observing spectra itself instead (e.g. non-negative matrix factorization). Reference [21] proposes to exploit the acoustic spectrogram directly within the NMF framework. This work focuses in semi-supervised to unsupervised diarization of recordings. Besides tackling the problem of overlapping signals, the proposed method attempts to solve two main issues: detection of unknown acoustic events, and the lack of labeled AED training resources. This is done by defining a gen-

erative model in which the overlapping acoustic events are modeled based on non-negative factorization matrix (NMF), incorporating Bayesian modeling. NMF is in charge of decomposing signals into two components: a set of basis and their activations in time. By incorporating Bayesian modeling, the model can autonomously determine the appropriate number of basis. However, while these approaches succeed to some extent, computational cost and relevance of discovered information are still a significant trade-off.

2.1 Deep Learning and AED

Phone classification in ASR and detection of single non-speech acoustic events similarities led in [13], [14] to exploit the tandem connectionist model in AED too. The tandem connectionist model [22] consists in discriminatively trained posteriors from a neural network that are fed to a generative model (typically a GMM-HMM) as features themselves. The model benefits from the neural network posterior features, and the nonlinear mapping it provides, revealing underlying relationships between events [13]. This is further exploited in [14], which includes extended features related to non-speech acoustic events in the model. Reference [18] also shows a preliminary study on using DNNs for scene classification using tuned low resolution features, yet little has been done in the classification of single events, detection of start and end points, or more suitable characterizations of non-speech events using such models.

2.2 Contributions of This Work

The goal of this work is to take advantage of both, an advanced classification model such as DNNs, and using the spectrogram directly as the input by defining a deep model that learns hidden features from the spectrogram, and fine-tunes itself for the specific task of AED.

The major contributions of this work are: overcoming the limitations of low resolution features in non-speech acoustic event detection by using high-resolution spectrogram directly as a features; a model in which short-time temporal structure is directly embedded in the feature by using spectrogram patches as input, rather than single-frame features; providing a model that is able to deal with such a high-resolution input, deep neural networks, along with necessary and specific modifications to model power spectrogram too; and a detailed analysis of the learning process and results which show substantial improvements in recognition accuracy over systems based on derived features for a speech overlapping conditions task.

The presented approach tackles the issue of recognizing acoustic events that overlap with speech, and does not offer support for overlapping of multiple non-speech events as there is only a single recognition stream as the output. That means that only the following four scenarios are considered: silence, speech, acoustic events one at a time, or speech and single acoustic events. Polyphony approaches such as the one described in [23] could be considered, although this is out of the scope of this work.

3. High-Resolution Spectrogram Patch Modeling

RBM provide the means to obtain binary representations of spectrogram, a powerful functionality to characterize classes in data, but this is not the first time this has been applied. In speech coding, [24] introduced a deep encoder with a first layer pre-trained to obtain binary representations of log-power-spectra enabling it to capture greater detail from acoustic signals. In AED, exploiting this fact can allow the hidden layer to learn insights from non-speech signals. In this paper we propose two approaches: modeling log-power-spectra (Sect. 3.1) and, as in many spectrogram part decomposition approaches, raw power-spectra (Sect. 3.2), for which we introduce exponential unit RBMs to train a power-spectrogram patch modeling layer. The resulting layer can be effortlessly integrated in the DNN framework for classification, as it is the ultimate goal of this work. Further on DNN training and classification can be found in Sect. 3.3.

3.1 Log-Power-Spectrogram Patch

For log-power-spectra, log transformation allows us to model this with Gaussian (real-valued) input units, which are currently supported in RBMs. The spectrogram patch auto-encoder is trained as an RBM with Gaussian visible units to model acoustic features, and Bernoulli (binary) hidden units (see Fig. 3), just as it is done also with MFCC or Mel-filter-bank features. RBMs are trained by contrastive divergence [19], which repeatedly updates the parameters using the difference between the correlations of the training data, and the reconstruction sampled from that.

$$\Delta w_{ij} \propto \langle v_i h_j \rangle_{\text{data}} - \langle v_i h_j \rangle_{\text{reconstruction}} \tag{1}$$

Reconstructions are produced by sampling the required distribution from the hidden states computed from actual data. In the case of log-power-spectra, reconstructions

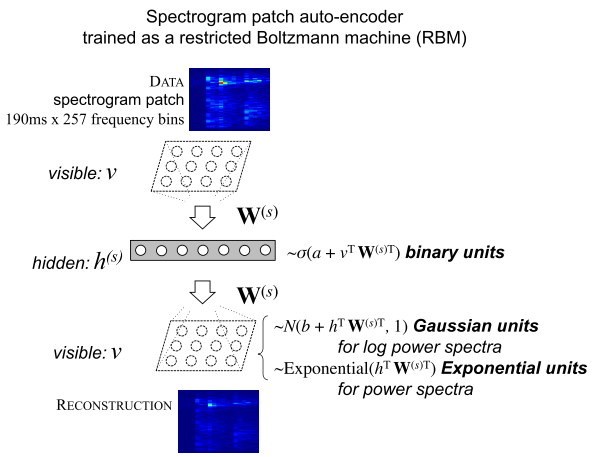

Fig. 3 Overview of the pre-training for *log-power-spectrogram* and *power-spectrogram* patch feature layer.

are sampled using a Gaussian distribution, hence Gaussian units. Gaussian units require a modified equation for the free energy as follows,

$$E(v,h) = \frac{1}{2}(v-b)^T(v-b) - a^T h - v^T Wh \tag{2}$$

Visible units are subject to a conditional distribution as follows

$$\begin{aligned} P(v|h) &= \mathcal{N}(\mu, 1) \\ \mu &= b + h^T \mathrm{W}^T \end{aligned} \tag{3}$$

where v and h refer to the array of values of visible and hidden units, respectively.

3.2 Power-Spectrogram Patch

As literature shows, power-spectra is better handled with exponential distributions [25], rather than Gaussian distributions. Therefore, we require the visible units to be exponential in this case. Note that with exponential units we do not normalize the data, and we ignore the bias,

$$E(v,h) = -v^T Wh \tag{4}$$

and the visible units are defined by the conditional distribution,

$$\begin{aligned} P(v|h) &= \text{Exponential}(\lambda) \\ \lambda &= h^T \mathrm{W}^T \end{aligned} \tag{5}$$

which can be sampled as,

$$\frac{-\ln U}{\lambda} \tag{6}$$

where U is a random variate drawn from a uniform distribution, and λ is the distribution parameter [26].

3.3 DNN and Acoustic Modeling

DNNs training strategy is not original of this work, and while we summarize the keypoinrts here, further details can be found in [17], [19]. Training of DNNs consists of two main stages: generative *pre-training* of each hidden layer as an RBM [19], and discriminative *fine-tuning* of the whole network as a multi-layer perceptron (MLP) using the back propagation algorithm [17]. This is roughly described in Fig. 4.

1. Training data is sampled through the spectrogram patch hidden layer described in Sects. 3.1 and 3.2. Then, sampled data is considered the feature (i.e. visible units) to a new RBM. In thin new RBM both layers will be considered as binary, and after training this will become the first regular hidden layer $h^{(1)}$.
2. As in previous step, training data is sampled through all previous hidden layers to obtain a representation of the input and then used as data for the following layer $h^{(2)}$. This step repeats until reaching the desired number of hidden layers, resulting in a deep network with each layer having learned a good representation of the data in its predecessor.
3. Finally, the entire network is fine-tuned using the back-propagation algorithm. The network is trained so that an output "softmax" layer placed at the output of the network estimates label units representing HMM states. The discriminative training learns the weights from the last layer to the label units, and re-trains the detectors from unsupervised pre-training, using labeled data.

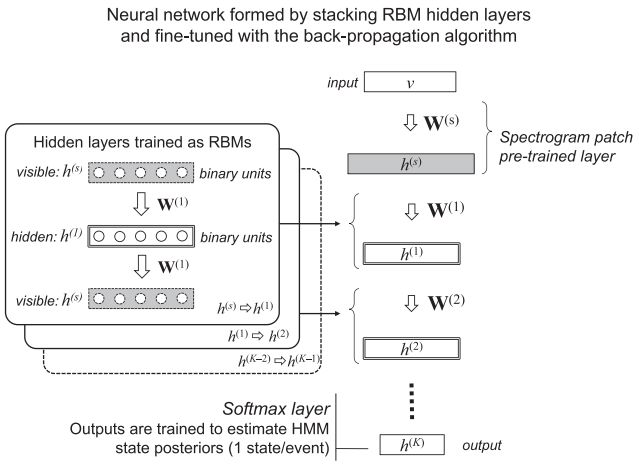

Fig. 4 Overview of the DNN layer pre-training and stacking.

Once we have pre-trained the hidden layers and fine-tuned the entire network to output label probability distributions over the central frame of the window, the most likely sequence of acoustic events $\hat{u}$ associated with the input sequence of features X is determined as,

$$\hat{u} = \underset{u}{\text{argmax}}\, P(u|X) = \underset{u}{\text{argmax}}\, P(X|u)P(u) \tag{7}$$

given, a sequence of observations $X = \{x_1, x_2, \cdots x_t | x_t \in \mathbb{R}^D\}$ where D denotes the number of features.

Then a sequence of acoustic events can be represented as a particular sequence of states s, leaving $P(X|u)$ as follows,

$$P(X|u) = \sum_{x_1,x_2,\cdots} \prod_t P(x_t|s_t)P(s_t|s_{t-1},u) \tag{8}$$

$$P(x_t|s_t) = \frac{P(s_t|x_t)}{P(s_t)}P(x_t) \tag{9}$$

acting as a conventional GMM-HMM, only that the observation probabilities $P(s_t|x_t)$ are estimated by the DNN instead of the GMM, considering x_t as the input and $P(s_t|x_t)$ as the output of the DNN. The decoding network consists of an ergodic HMM with equal transition probabilities.

4. Visualizing the Learning Process

Besides the goal of achieving robust performance in AED, we also want to observe how the proposed model learns new

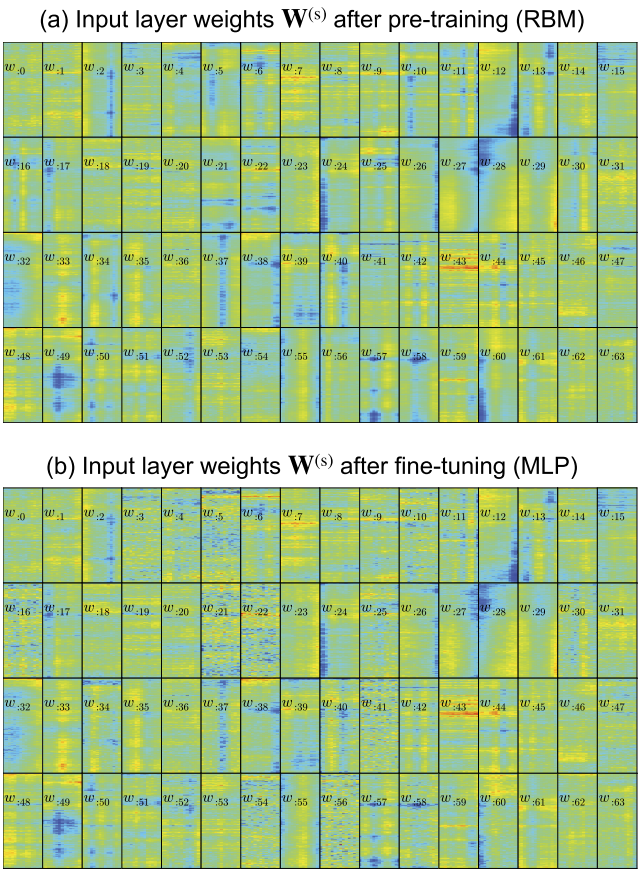

Fig. 5 A simple example of evolution of the weights in a small DNN between the input (spectrogram patch) and the first hidden layers throughout after pre-training as an RBM (a), and after fine-tuning the entire network as an MLP (b).

features. To do so we have visualized the features that the proposed model learns in an unsupervised manner. As we have defined in the previous section, the model consists of a series of layers with hidden nodes, which are fully connected to the previous and posterior layers with weights between each of the nodes, and a bias. Basically, by propagating the input through the neural network, each layer consists of a dot product of the input and the weights matrix between these layers, and adding the bias. Then the activation function, sigmoid for hidden layers and *softmax* for the output layer, is applied.

While the entire model is in charge of feature extraction and classification, it is the first layer the one that will be dealing directly with the input and the one that will show more insights. That is why we have obtained instant values of the weights matrix between the input, and first hidden layer to observe its evolution. Figure 5 shows 2D plots of this weight matrix, which can be analyzed as follows:

- each of the boxes $w_{:i}$ accounts for the weights connecting all the input nodes (spectrogram patch) to node i
- since the weights are directly connected to the input we have reshaped them in to 2D patches which vertical and horizontal axes account for frequency bin and time frame, respectively.
- the first matrix plot (a) refers to the weights after the pre-training as an RBM model, while the second plot (b) refers to the final weight matrix after the fine-tuning stage where the entire network is trained as an MLP.

Models trained in the experimental evaluation are larger having many more parameters, however, for ease of observation and analysis we show here a rather small sized model (a DNN with 4 hidden layers and 64 nodes per layer) which allows plots to be analyzed by simply looking at them. Bear in mind that the models trained during evaluation have an input layer with 1024 nodes, which makes the weights matrix 16 times bigger. The models here have been trained using the isolated acoustic events database (seminar room sounds), and further description of this and other data can be found in Sect. 5.

Figure 5 (a) show the weights after pre-training the input layer as an RBM for 200 epochs (one epoch refers to one pass over the entire training dataset). We have confirmed that after randomly initializing the weights, most of learning happens within the first 100 epochs. RBMs are expected to learn a good representations of data, i.e. hidden layers that are able to explain the data they have been trained with. Two-dimensional features after pre-training show energy concentrating in specific spectro-temporal regions. These spectro-temporal shapes do not account for any specific acoustic event, but we can assume they characterize components that together with other components would form spectro-temporal objects for acoustic events.

Fine-tuning of the entire network does not change much as it can be seen in Fig. 5 (b). This is consistent with the assumption that in the pre-training process the network parameters are left in a close-to-optimal state, and therefor, much of the change during fine-tuning happens in the upper layers, considering that the output layer has randomly initialized before fine-tuning.

Another interesting aspect of these fine-tuned features is how some of them get certain randomness in their spectro-temporal shapes (e.g. $w_{:3}$, $w_{:5}$, $w_{:41}$, etc.). Conceptually, the goals of RBM pre-training and back-propagation fine-tuning go in different directions: while RBMs are trained reduce the gap between the original signal and the signal reconstructed using the RBM, back-propagation training aims at reducing class discrimination error. In this way, after fine-tuning, some spectro-temporal features are modified to obtain better discrimination at the output of the DNN, while others might remain almost the same as pre-training is expected to achieve exactly that. The randomness in some of these features after fine-tuning could indicate that certain features are being tuned to better detect "silence" as it is one of targeted classes at the ouptut of the DNN. In most cases, the absence of sound is not sparse in the frequency domain, and such random-like features might allow discrimination of silence from other sounds.

5. Experimental Evaluation

The performance and comparison of the proposed approach has been evaluated over the acoustic event recognition task in CHIL2007 [1], a database of seminar recordings where 12 acoustic event classes appear besides speech (around 60% of the acoustic events are reportedly overlapped with speech). The experimental conditions are summarized in Table 1. All models were trained using the training dataset provided by CHIL2007, a dataset of isolated acoustic events which we split in *train* and *dev* sets for pre-training and training of the networks, respectively. Evaluation was completed with the *test* dataset which contains 20 live seminar recordings involving 4 to 5 participants in a seminar where there are presentations, discussion, break-times, etc.

5.1 Evaluation Metrics

Two metrics have been obtained to analyze the performance of our models and compare it with existing approaches: frame-score, and AED-accuracy. The first, frame-score, accounts for the percentage of correctly recognized frames at the output of the system including "silence." AED-accuracy, originally defined in the CHIL challenge [27], refers to the F-measure between precision and recall, considering correctly detected acoustic events as those where either the detected central frame falls within a ground-truth event period, or a ground-truth event central frame falls within the detected period. Such a "correctly detected event" definition comes from the fact that unlike speech, we just need to know when, and if, an acoustic event happened with no major worries on obtaining exact start and end times. Both measures allow us to evaluate the models in terms of frame-wise and event-wise performances.

5.2 Evaluated Methods

The results include our proposed approaches using log-spectrogram patches (referred as LOGSPEC), and power-spectrogram patches (referred as SPEC), both described in Sect. 3. Accordingly, we have compared these with state-of-the-art DNN models using derived features: Mel-frequency cepstrum coefficients (referred as MFCC), and 26 coefficients Mel-filter-bank (referred as FBANK), both replicating the DNN architectures used in the proposed model with Gaussian input RBM pre-training. Additionally, we also included reference AED-accuracy results in the same CHIL2007 task for [13] and [14] which are based in the tandem connectionist model and use custom features based in MFCC and Mel-filter-bank. Best scores for all models are summarized in Table 4.

Note that the number of frames stacked in the spectrogram patch is larger than the context window with derived features. While MFCC and Mel-filter-bank models have only 11 frames context windows, they include two-frames-wide deltas and acceleration coefficients, enlarging the information in those context windows with four frames before and after. Spectrogram patches have no delta or acceleration coefficients and therefore width has been set to 19 frames (4+11+4) so they contain equivalent information. As frames are 25 ms long, each of the spectrograms patches contain 205 ms of consecutive frames. As in many of the events targeted in these experiments are usually shorter (e.g. a knock, a clap, a door slam, etc.), we can assume the patches package enough spectro-temporal detail. Bear in mind this assumption is task-specific, and tasks with more complex sounds might require longer patches, or more sophisticated HMM architectures.

Table 1 Experimental conditions.

Acoustic events (12 events)	applause, spoon/cup jingle, chair moving, cough, door slam, key jingle, door knock, keyboard typing, laugh, phone ring, paper wrapping, and steps
Datasets	*train*: 10 sessions, 303 events × 9 overlapping speech conditions (3.5h) → pre-training (RBM) *dev*: 10 sessions, 306 events × 9 overlapping speech conditions (3.5h) → fine-tuning (back-propagation) *test*: 20 sessions (1.5h)
Features	Re-sampled to 16 kHz Frame: 25 ms, shift: 10 ms MFCC (39 features): MFCC (12) + energy (1) including deltas and accelerations FBANK (78 features): Mel-filter-bank (26) including deltas and accelerations **LOGSPEC (257 features): log-power-spectrum** Gaussian-binary RBM (1024 hidden units) **SPEC (257 features): power-spectrum** Exponential-binary RBM (1024 hidden units)
Context window	MFCC, FBANK: 11 frames (current ±5) LOGSPEC, SPEC: 19 frames (current ±9)
Nodes per layer	256, 512
Hidden layers	1, 2, 3, 4, 5, 6, 7, 8, 9, 10

5.3 Robust Training Setup

As seen in many recent DNN-based ASR works [28], DNNs perform reasonably well when they are trained with large datasets, while the opposite happens with datasets that are small. This is usually assumed to be caused by DNN models being weak to unseen data. Acoustic events are short and sparse in real environments, and this is no exception in the CHIL dataset. Since the target is to achieve robust performance in the presence of speech, we have introduced a step previous to the pre-training and training in which we augment the dataset by mixing it with speech signals. This is done by adding a random chunk of speech to the signal with a specific signal-to-noise ratio, where the original sound is the signal, and the random chunk of speech is the noise.

In order to obtain robust models, acoustic conditions of both *train* and *dev* datasets have been enlarged by adding overlapping speech publicly available from AURORA-4 [29], under several signal-to-noise ratio conditions: −9, −6, −3, 0, 3, 6, 9, 12, 15, and 18 dB.

Table 2 AED evaluation results: frame-score (%) by input feature and nodes/layer, and number of hidden layers.

input	**MFCC**		**FBANK**		**LOGSPEC**		**SPEC**	
nodes/layer	256	512	256	512	256	512	256	512
1	68.9	69.9	59.0	60.2	75.6	75.4	71.6	70.9
2	69.9	70.1	60.5	60.3	75.7	75.7	70.6	71.8
3	70.5	71.0	61.3	60.9	75.3	75.5	72.3	72.3
4	70.3	**72.3**	59.8	61.4	**75.9**	76.0	72.4	72.0
5	70.1	69.5	61.8	60.9	75.8	76.1	**72.8**	72.8
6	**71.8**	69.4	62.7	60.2	75.8	76.0	72.2	72.5
7	69.9	69.9	61.6	**62.3**	75.8	75.9	72.7	72.7
8	70.7	70.0	**65.3**	61.8	75.9	**76.1**	72.8	**73.0**
9	69.6	70.3	61.9	61.3	76.2	75.8	72.4	72.4
10	70.4	70.0	61.3	59.9	75.7	75.3	72.8	72.4

Table 3 AED evaluation results: AED-accuracy (%) by input feature and nodes/layer, and number of hidden layers.

input	**MFCC**		**FBANK**		**LOGSPEC**		**SPEC**	
nodes/layer	256	512	256	512	256	512	256	512
1	**55.1**	56.6	31.7	32.8	62.2	60.4	60.6	55.5
2	53.0	55.4	31.8	33.3	60.9	59.4	59.1	56.2
3	53.0	53.9	37.2	36.5	59.7	59.3	60.3	60.3
4	52.5	**59.2**	33.2	36.8	62.8	61.2	59.9	60.4
5	52.5	50.5	40.2	34.7	60.8	61.0	**62.9**	**62.9**
6	57.0	51.4	40.9	33.9	60.5	61.0	58.4	59.2
7	49.8	52.5	37.4	**38.6**	62.0	**61.6**	59.6	59.6
8	52.8	53.9	**45.5**	36.3	**63.2**	60.4	59.7	60.0
9	50.7	51.4	39.0	34.8	61.5	60.8	61.7	61.7
10	54.2	50.8	36.7	32.3	60.0	58.7	58.0	60.0

5.4 Results and Analysis

Tables 2 and 3 contain the complete list of results for all settings as previously described, using spectrogram patches and derived features as inputs: MFCCs, Mel-filter-bank (FBANK), log-power-spectra (LOGSPEC), and power-spectra (SPEC); and 1 through 10 layer DNNs to observe the performance variations. The results obtained with narrow (256 nodes/layer) and wide (512 nodes/layer) regular hidden layer DNNs are also shown to understand how width of the network affects the performance.

Best scores are obtained by spectrogram patch-based models (LOGSPEC and SPEC) in terms of frame-accuracy as it can be seen in Table 2. While LOGSPEC scores appear to outperform other models, also notice that FBANK performs much worse than the other three models in both 256 and 512 nodes/layer. Results are consistent in event-wise accuracy (Table 3) where we can observe how LOGSPEC outperforms all other models. Here, top three models are much closer. In this table, we can also see scores reported in [13], [14], both outperformed by LOGSPEC, SPEC, and MFCC models.

It is worth as well noting the effects of width and depth of the DNN. This can be observed in both Table 2 and Table 3. While performance benefits from models with more layers, improvement slows down significantly after four layers. Nonetheless, best scores both in frame-based (frame-score) and event-based (AED-accuracy) evaluation, are not so different between four and eight layers.

Table 4 An AED-accuracy score summary with the best performing configurations in each model, and existing approaches [13], [14].

Method	AED-Acc.
Zhuang'2010 [13]	41.20 %
Espi'2012 [14]	49.83 %
DNN-HMM-MFCC (4 layers × 512 nodes)	57.03 %
DNN-HMM-FBANK (8 layers × 256 nodes)	45.50 %
DNN-HMM-LOGSPEC (4 layers × 256 nodes)	**63.20 %**
DNN-HMM-SPEC (8 layers × 512 nodes)	**62.90 %**

Additionally, the effects of width in the model are also interesting. While increasing the number of parameters provided better results with less layers, a much narrower network of 256 nodes per layer provided better results in deeper networks compared to a wider network of 512 nodes per layer. A direct advantage of narrower models is computational efficiency, both during training and classification.

6. Conclusion and Future Work

We have described a high-resolution spectral input model based on deep learning for acoustic event recognition that achieves significantly better results than existing works on a challenging seminar recording dataset, where acoustic events overlap constantly with speech. Although our experiments in AED-accuracy show undeniable improvements when using the proposed spectrogram patch input models, it is the log-power-spectra input model that provides better performance. That being said, power-spectra input models come close in performance, out-performing in fact similar models based on derived features, rather than on spectrogram patches. This is, as we have claimed, because a high-resolution input such as spectrogram patches provide comparatively more time-frequency detail than low resolution features like MFCCs or Mel-filter-bank features. More importantly, the proposed deep neural network model, along with pre-training and training are able to model such a high dimensional input and learn discriminative features during pre-training. While pre-training and training are quite expensive steps, classification and later decoding are comparatively much light-weight, computationally speaking. This could eventually allow the presented approach to be performed in real-time.

From a broader point of view, the presented DNN approach looks over spectrogram patches as a whole, looking for "global" shapes that together form acoustic events and the actual spectrogram. However, this ignores "local" properties of sounds like stationarity, transiency, etc. Without leaving the deep learning approach there are models such as convolution layers that enable extraction of features in this domain [30], and we believe it is a path worth exploring in the future in combination with the current approach.

References

[1] D. Mostefa, N. Moreau, K. Choukri, G. Potamianos, S. Chu, A. Tyagi, J. Casas, J. Turmo, L. Cristoforetti, F. Tobia, A. Pnevmatikakis, V. Mylonakis, F. Talantzis, S. Burger, R.

Stiefelhagen, K. Bernardin, and C. Rochet, "The CHIL audiovisual corpus for lecture and meeting analysis inside smart rooms," Language Resources and Evaluation, vol.41, no.3-4, pp.389–407, 2007.

[2] D. Giannoulis, E. Benetos, D. Stowell, M. Rossignol, M. Lagrange, and M. Plumbley, "Detection and classification of acoustic scenes and events: An IEEE AASP challenge," 2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), pp.1–4, 2013.

[3] K. Imoto, S. Shimauchi, H. Uematsu, and H. Ohmuro, "User activity estimation method based on probabilistic generative model of acoustic event sequence with user activity and its subordinate categories," INTERSPEECH'2013, pp.2609–2613, 2013.

[4] C. Canton-Ferrer, T. Butko, C. Segura, X. Giro, C. Nadeu, J. Hernando, and J. Casas, "Audiovisual event detection towards scene understanding," IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR), pp.81–88, 2009.

[5] T. Hori, S. Araki, T. Yoshioka, M. Fujimoto, S. Watanabe, T. Oba, A. Ogawa, K. Otsuka, D. Mikami, K. Kinoshita, T. Nakatani, A. Nakamura, and J. Yamato, "Low-latency real-time meeting recognition and understanding using distant microphones and omni-directional camera," IEEE Trans. Audio, Speech, Language Process., vol.20, no.2, pp.499–513, 2012.

[6] A. Ozerov, A. Liutkus, R. Badeau, and G. Richard, "Informed source separation: source coding meets source separation," 2011 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), pp.257–260, IEEE, 2011.

[7] A. Harma, M. McKinney, and J. Skowronek, "Automatic surveillance of the acoustic activity in our living environment," IEEE International Conference on Multimedia and Expo (ICME), pp.634–637, 2005.

[8] R. Cai, L. Lu, A. Hanjalic, H.-J. Zhang, and L.-H. Cai, "A flexible framework for key audio effects detection and auditory con- text inference," IEEE Transactions on Audio, Speech and Language Processing, vol.14, no.3, pp.1026–1039, 2006.

[9] M. Xu, C. Xu, L. Duan, J. Jin, and S. Luo, "Audio key- words generation for sports video analysis," ACM Transactions on Multimedia Computing, Communications, and Applications, vol.4, no.2, pp.1–23, 2008.

[10] Y.-T. Peng, C.-Y. Lin, M.-T. Sun, and K.-C. Tsai, "Healthcare audio event classification using hidden Markov models and hierarchical hidden Markov models," IEEE International Conference on Multimedia and Expo (ICME), pp.1218–1221, 2009.

[11] M. Shah, B. Mears, C. Chakrabarti, and A. Spanias, "Lifelogging: Archival and retrieval of continuously recorded audio using wearable devices," Emerging Signal Processing Applications, pp.99–102, 2012.

[12] G. Wichern, J. Xue, H. Thornburg, B. Mechtley, and A. Spanias, "Segmentation, indexing, and retrieval for environmental and natural sounds," IEEE Trans. Audio, Speech, Language Process., vol.18, no.3, pp.688–707, 2010.

[13] X. Zhuang, X. Zhou, M. Hasegawa-Johnson, and T. Huang, "Real–world acoustic event detection," Pattern Recognition Letters, vol.31, no.12, pp.1543–1551, 2010.

[14] M. Espi, M. Fujimoto, D. Saito, N. Ono, and S. Sagayama, "A tandem connectionist model using combination of multi-scale spectro-temporal features for acoustic event detection," 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp.4293–4296, 2012.

[15] S. Araki, T. Nakatani, and H. Sawada, "Simultaneous clustering of mixing and spectral model parameters for blind sparse source separation," 2010 IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP), pp.5–8, 2010.

[16] T. Nakatani and S. Araki, "Single channel source separation based on sparse source observation model with harmonic constraint," 2010 IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP), pp.13–16, 2010.

[17] A.-R. Mohamed, G. Dahl, and G. Hinton, "Acoustic modeling using deep belief networks," IEEE Trans. Audio, Speech, Language Process., vol.20, no.1, pp.14–22, 2012.

[18] Z. Kons and O. Toledo-Ronen, "Audio event classification using deep neural networks," INTERSPEECH'2013, pp.1482–1486, 2013.

[19] G. Hinton, "A practical guide to training restricted boltzmann machines," Technical report 2010-003, Machine Learning Group – University of Toronto, 2010.

[20] M. Espi, M. Fujimoto, Y. Kubo, and T. Nakatani, "Spectrogram patch based acoustic event detection and classification in speech overlapping conditions," 2014 4th Joint Workshop on Hands-free Speech Communication and Microphone Arrays (HSCMA), pp.117–121, May 2014.

[21] Y. Ohishi, D. Mochihashi, T. Matsui, M. Nakano, H. Kameoka, T. Izumitani, and K. Kashino, "Bayesian semi-supervised audio event transcription based on Markov indian buffet process," 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp.3163–3167, 2013.

[22] H. Hermansky, D.P.W. Ellis, and S. Sharma, "Tandem connectionist feature extraction for conventional HMM systems," 2000 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp.1635–1638, 2000.

[23] T. Heittola, A. Mesaros, A. Eronen, and T. Virtanen, "Context-dependent sound event detection," EURASIP Journal on Audio, Speech, and Music Processing, vol.2013, no.1, pp.1–13, 2013.

[24] L. Deng, M.L. Seltzer, D. Yu, A. Acero, A. Mohamed, and G. Hinton, "Binary coding of speech spectrograms using a deep auto-encoder," INTERSPEECH'2010, pp.1692–1695, 2010.

[25] Y. Lu and P.C. Loizou, "Estimators of the magnitude-squared spectrum and methods for incorporating SNR uncertainty," IEEE Trans. Audio, Speech, Language Process., vol.19, no.5, pp.1123–1137, 2011.

[26] A. Papoulis and S. Pillai, Probability, random variables and stochastic processes with errata sheet, McGraw-Hill Education, New York, NY, 2002.

[27] R. Stiefelhagen, K. Bernardin, R. Bowers, R.T. Rose, M. Michel, and J. Garofolo, "The CLEAR 2007 evaluation," in Multimodal Technologies for Perception of Humans, pp.3–34, Springer, 2008.

[28] M. Delcroix, T. Yoshioka, A. Ogawa, Y. Kubo, M. Fujimoto, N. Ito, K. Kinoshita, M. Espi, T. Hori, T. Nakatani, and A. Nakamura, "Linear prediction-based dereverberation with advanced speech enhancement and recognition technologies for the reverb challenge," Proc. Reverb Challenge 2014, 2014.

[29] H. Hirsch and D. Pearce, "AURORA-4." http://aurora.hsnr.de/aurora-4.html

[30] S. Thomas, S. Ganapathy, G. Saon, and H. Soltau, "Analyzing convolutional neural networks for speech activity detection in mismatched acoustic conditions," 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp.2519–2523, May 2014.

Miquel Espi received the B.E. degree from Universidad Politecnica de Valencia, Spain, in 2006, M.E. from Kagoshima University, Japan, in 2010, and Ph.D. from The University of Tokyo, Japan, in 2013. He is a research associate at NTT Communication Science Laboratories, NTT Corporation, Japan since 2013. He has been contributing to the field of acoustic event detection throughout the Ph.D. and as a research associate at the Signal Processing Group at NTT CS Laboratories. His research interests include acoustic scene analysis, and social dynamics. Dr. Espi is a member of IEEE, and ASJ.

Masakiyo Fujimoto received the B.E., M.E., and Dr. Eng. degrees from Ryukoku University in 1997, 2001, and 2005, respectively. From 2004 to 2006, he worked with ATR Spoken Language Communication Research Laboratories, Kyoto. He joined NTT Communication Science Laboratories, Kyoto in 2006. His current research interests are noise-robust speech recognition including voice activity detection and speech enhancement. Dr. Fujimoto was honored to receive the Awaya Prize Young Researcher Award from Acoustical Society of Japan (ASJ) in 2003, the MVE Award from IEICE SIG MVE in 2008, the Yamashita SIG Research Award from the Information Processing Society of Japan (IPSJ) in 2011, and the Information and Systems Society (ISS) Distinguished Reviewer Award from Institute of Electronics, Information and Communication Engineers (IEICE) in 2011. He is a member of ISCA, IEEE, IPSJ, and ASJ.

Tomohiro Nakatani received the B.E., M.E., and Ph.D. degrees from Kyoto University, Kyoto, Japan, in 1989, 1991, and 2002, respectively. He is a Senior Research Scientist (Supervisor) of NTT Communication Science Laboratories, NTT Corporation, Kyoto, Japan. Since joining NTT Corporation as a Researcher in 1991, he has been investigating speech enhancement technologies for developing intelligent human-machine interfaces. Since 2005, he has visited the Georgia Institute of Technology as a Visiting Scholar for a year. Since 2008, he has been a Visiting Assistant Professor in the Department of Media Science, Nagoya University. Dr. Nakatani was honored to receive the 1997 JSAI Conference Best Paper Award, the 2002 ASJ Poster Award, the 2005 IEICE Best Paper Award, and the 2009 ASJ Technical Development Award. He is a member of IEICE, IEEE, and ASJ.