ACOUSTIC SCENE CLASSIFICATION USING SPARSE FEATURE LEARNING AND EVENT-BASED POOLING

Kyogu Lee and Ziwon Hyung *

Seoul National University
Music and Audio Research Group
{ziotoss,kglee}@snu.ac.kr

Juhan Nam

Stanford University
CCRMA
juhan@ccrma.stanford.edu

ABSTRACT

Recently unsupervised learning algorithms have been successfully used to represent data in many of machine recognition tasks. In particular, sparse feature learning algorithms have shown that they can not only discover meaningful structures from raw data but also outperform many hand-engineered features. In this paper, we apply the sparse feature learning approach to acoustic scene classification. We use a sparse restricted Boltzmann machine to capture manyfold local acoustic structures from audio data and represent the data in a high-dimensional sparse feature space given the learned structures. For scene classification, we summarize the local features by pooling over audio scene data. While the feature pooling is typically performed over uniformly divided segments, we suggest a new pooling method, which first detects audio events and then performs pooling only over detected events, considering the irregular occurrence of audio events in acoustic scene data. We evaluate the learned features on the IEEE AASP Challenge development set, comparing them with a baseline model using mel-frequency cepstral coefficients (MFCCs). The results show that learned features outperform MFCCs, event-based pooling achieves higher accuracy than uniform pooling and, furthermore, a combination of the two methods performs even better than either one used alone.

Index Terms— acoustic scene classification, environmental sound, feature learning, restricted Boltzmann machine, sparse feature representation, max-pooling, event detection

1. INTRODUCTION

Popularly used audio features, such as MFCCs, chroma and low-level spectral features (spectral centroid, flux, roll-off, etc.), were designed based on domain-specific knowledge. As an alternative to the engineering approach, researchers recently have made great efforts to find salient features by unsupervised learning algorithms. In particular, using sparsity constraint, they have demonstrated that the algorithms can discover meaningful hidden structures from audio data, for example, harmonic or transient patterns in a spectrogram domain, and also representing data given the learned structures can beat many engineered features [1, 2, 3, 4]. While the feature learning approach has been actively applied to speech or music data, relatively less attention has been paid to environmental sounds. In this paper, we examine the feature learning approach on environmental sounds and evaluate it on an acoustic scene classification task.[1]

1.1. Related Work

Lyon *et al.* presented a sparse auditory feature representation to retrieve and rank general sounds in a large-scale framework [5]. They obtained feature bases on sub-patches of auditory filter images using K-means and summarized extracted features given the learned bases. They showed that the sparse auditory features outperform those from MFCC front ends. Cotton and Ellis also used K-means but they used multiple frames of a mel-frequency spectrum and reduced dimensionality with principal component analysis (PCA) [6]. They also showed that these learned features are superior to those from MFCCs.

While these approaches represent audio data using unsupervised learning, the idea is based on traditional vector quantization, handling extracted local features as count data. In this paper, we represent audio data using a sparse restricted Boltzmann machine (RBM) that has better expressive power and sparsity control [7]. In [4], we applied the sparse RBM to music data and showed good performance in music annotation and retrieval tasks. Using a similar data processing pipeline, we learn local audio features on a mel-frequency spectrogram and summarize them using max-pooling and averaging. However, considering the infrequency of sound events in acoustic scene data, we first detect sound events using mean activation of local features and then perform pooling over the events. We will show that this event-based pooling is effective in acoustic scene classification.

2. PROPOSED METHOD

2.1. Sparse Feature Learning

An RBM is a bipartite undirected graphical model that consists of visible nodes $\mathbf{v}$ and hidden nodes $\mathbf{h}$. The visible nodes correspond to input vectors in a training set and the hidden nodes correspond to the feature detectors. We used real-valued Gaussian units for the visible nodes and binary units for the hidden nodes. The joint probability of $\mathbf{v}$ and $\mathbf{h}$ is defined by an energy function $E(\mathbf{v}, \mathbf{h})$:

$$p(\mathbf{v}, \mathbf{h}) = \frac{e^{-E(\mathbf{v},\mathbf{h})}}{Z} \quad (1)$$

$$E(\mathbf{v}, \mathbf{h}) = \mathbf{v}^T\mathbf{v} - \left(\mathbf{b}^T\mathbf{v} + \mathbf{c}^T\mathbf{h} + \mathbf{v}^T\mathbf{W}\mathbf{h}\right) \quad (2)$$

where $\mathbf{b}$ and $\mathbf{c}$ are bias terms, and $\mathbf{W}$ is a weight matrix. The normalization factor Z is called the partition function, which is obtained by summing all possible configurations of $\mathbf{v}$ and $\mathbf{h}$. The RBM has symmetric connections between the two layers but no connections

*This research was supported by the Basic Science Research Program, through the National Research Foundation of Korea (NRF), funded by the Ministry of Education, Science and Technology (2011-0013476).

[1] This research was conducted for the IEEE AASP Challenge 2013.

978-1-4799-0972-8/13/$31.00 ©2013IEEE

within the hidden nodes or visible nodes. This conditional independence makes it easy to compute the conditional probability distributions, when nodes in either layer are observed:

$$p(h_j|\mathbf{v}) = g(c_j + \sum_i W_{ij} v_i) \tag{3}$$

$$p(v_i|\mathbf{h}) = \mathcal{N}(b_i + \sum_j W_{ij} h_j, 1), \tag{4}$$

where $g(x) = 1/(1 + \exp(\text{-}x))$ is the logistic function and $\mathcal{N}(x)$ is the Gaussian distribution. The parameters are estimated by maximizing the log-likelihood of the visible nodes:

$$\Delta W_{ij} \propto \frac{\partial \log p(\mathbf{v})}{\partial W_{ij}} = \langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{model} \tag{5}$$

$$\Delta b_i \propto \frac{\partial \log p(\mathbf{v})}{\partial b_i} = \langle v_i \rangle_{data} - \langle v_i \rangle_{model} \tag{6}$$

$$\Delta c_j \propto \frac{\partial \log p(\mathbf{v})}{\partial c_j} = \langle h_j \rangle_{data} - \langle h_j \rangle_{model} \tag{7}$$

The angle brackets denote expectation with respect to the distributions from the training data and the model. $\langle v_i h_j \rangle_{data}$ can be easily obtained because hidden units h_j given the training data can be directly computed using Equation 3. However, exact computation of $\langle v_i h_j \rangle_{model}$ is intractable, i.e., needs to perform block Gibbs sampling between the two layers for a very long time. In practice, the learning rules in Equation 5, 6 and 7 converges well only with a single iteration of Gibbs sampling when it starts by setting the states of the visible units to the training data. This is called the *contrastive-divergence* [8].

Furthermore, a sparsity constraint can be added on hidden units. We used the method in [7], which promotes sparsity by forcing each hidden unit to have a pre-determined expected activation using a regularization penalty:

$$\lambda \sum_j (\rho - \frac{1}{m}(\sum_{k=1}^{m} \left\langle h_j | \mathbf{v}^k \right\rangle))^2, \tag{8}$$

where $\{\mathbf{v}^1, ..., \mathbf{v}^m\}$ is the training set and ρ determines the target sparsity of the hidden unit activations. This regularization term is taken into account by adding to the updating rule in Equation 7.

2.2. Feature Summarization

After training the sparse RBM, we extract local sparse features in a convolutional manner over an audio clip using Equation 7. We need to summarize them for acoustic-scene-level classification. A typical approach is to perform max-pooling over the local features; find the maximum value at each feature dimension over uniform segments, and then aggregate them. In particular, max-pooling followed by averaging (as a way of aggregation) was shown to be effective in music classification [4]. However, music is usually full of acoustics events (e.g. musical notes) in a dense and periodic manner, whereas environmental sounds are often silent and the acoustics events are somewhat irregular. Accounting for these properties of environmental sounds, we suggest a new pooling method based on acoustic event detection.

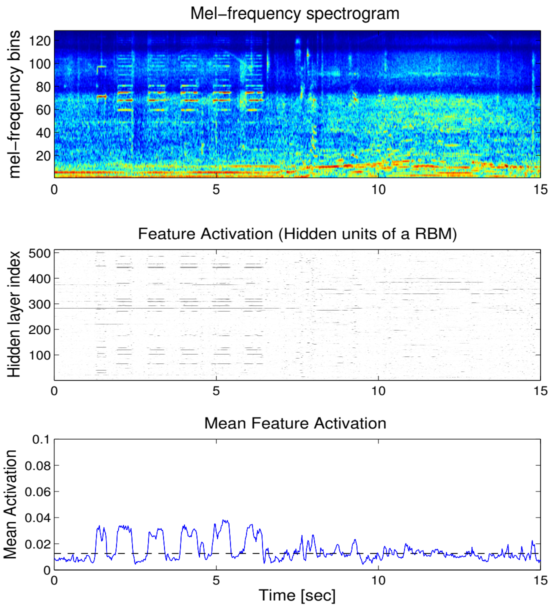

Figure 1: Mel-frequency spectrogram (top), sparse feature activation (middle) and mean feature activation (bottom). The mean feature activation is computed by averaging the hidden units in the middle pane (vertically). The black dashed line at the bottom indicates a threshold to detect events. Note that, while the target sparsity ρ is set to a constant value over the training set, the mean activation level varies depending on density and dynamics of acoustic events.

In Equation 8, the sparse RBM controls the hidden unit activation by adjusting mean value of hidden layer units to a target sparsity value. While this forces the mean activation to approximate to a constant level over the whole training set, mean activation of local hidden units is not necessarily close to the target sparsity value. That is, the local mean activation can be greater than the target sparsity level if acoustic events have a strong energy or can be less than that if there is no acoustic events. The bottom pane of Figure 1 shows the mean activation of hidden units. This indicates that the mean feature activation has such physical meaning and thus can be used as a way of detecting acoustic events. Leveraging the behaviors of the local mean activation in the sparse RBM, we first detect events and then perform max-pooling only over detected events. The procedure is detailed as follows:

1. Average the mean feature activation for each sound clip and set it as a threshold to detect acoustic events (a black dashed line at the bottom in Figure 1).
2. Mark the onset[offset] of an event at the time that the mean activation becomes greater[less] than the threshold. The onset and offset determine the duration of an event.
3. Discard short-lived events (e.g. those instantaneously meeting the threshold) by a pre-determined minimum timespan of an event.
4. Perform max-pooling over the detected event.
5. Repeat step 2 to 4 for all detected events and average them to construct a scene-level feature.

We term this feature summarization *event-based pooling* and the previous method (that performs max-pooling over uniformly divided segments) *uniform pooling* for comparison. In the experiment, we evaluated not only features from either of the two pooling methods but also the combined features from the both methods.

A similar event-based feature summarization was proposed in [6], where they detected sound events using spectral subband energy in multiple scales. However, our method exploits local sparse features which capture only important characteristics from spectral data and also have a regularized level that makes it easy to use a threshold.

2.3. Supervised Training for Classification

As a result of the feature summarization, we are given a pair of scene-level feature vectors and text labels. We finally performed supervised training using an L2-regularized linear support vector machine (SVM) in a one-vs-all manner. The regularization parameter is determined by cross-validation.

3. EVALUATION

3.1. Dataset

We evaluated the development dataset provided by the IEEE AASP challenge. The dataset contains ten different classes of acoustic scene audio clips [9]. For each class, they include ten audio clips and each of them is 30 seconds long. We first split the dataset into five folds of training (80 clips) and test sets (20 clips) for cross validation (CV). In order to simulate evaluation in the AASP challenge, we assume that the test set is unseen. That is, we additionally split the training set into four subsets and cross-validated over them. Then we performed re-training with the best set of parameters to finally evaluated the test set. In order to avoid possible overfitting caused by the small size of the AAAP development dataset, we repeated the 5-CV evaluation five times such that each round has different combinations of train/test split.

3.2. Baseline

We evaluated MFCCs and Gaussian mixture models (GMMs) as a baseline. For MFCCs, we computed 40 bins of mel-frequency spectrograms and took 13-dimensional features without deltas. We used GMMs as a classifier by training a GMM separately for each class of acoustic scene data. We cross-validated the number of mixture components over 16, 32, 64 and 128.

3.3. Preprocessing Parameters

We basically followed the preprocessing procedure in [4]. First we resampled the waveform (left channel only) to 22.05kHz and applied the time-frequency automatic gain control to regularize the volume of audio data in ten subbands. Then, we computed a spectrogram with a 46ms Hann window and 50% overlap, and mapped the linear frequency to a mel scale with 128 bins. Finally we compressed the amplitude using a log scale.

3.4. Feature Learning Parameters

We randomly sampled 100K examples for feature learning, taking four frames of the preprocessed mel-frequency spectrogram as a single example. Before applying the sparse RBM, we used PCA whitening as an additional preprocessing stage to reduce dimension. For the sparse RBM, the hidden layer size (feature size) was set to 256, 512 and 1024 and the target sparsity was to 0.01, 0.02, 0.03 and 0.05. In event-based pooling, the minimum timespan for events were cross-validated over 5, 8, 10, 12, 15, 18 and 20 frames. In uniform pooling, the pooling size was cross-validated over 22, 43, 86, 172 and 344 frames. In the combined features, we fixed the uniform pooling size to 86 or 172.

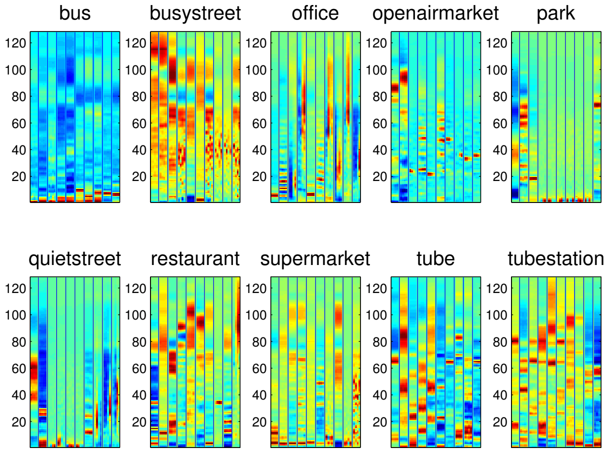

Figure 2: Feature bases learned by a sparse RBM. Ten most dominantly activated feature bases for each scene class are displayed.

4. RESULTS AND DISCUSSION

4.1. Feature Visualization

Figure 2 shows feature bases learned from the sampled training set. They were selected by finding most dominantly activated ones for each class of acoustic scene data given learned feature bases (the weight matrix **W** in the RBM). The selected bases explain timbral characteristics of each acoustic scene well. For example, *Bus* feature bases have low-frequency energy patterns that characterize engine sounds. *Restaurant*, *OpenAirMarket* and *SuperMarket* feature bases include some harmonic patterns which seem to correspond to human speech. *Park* and *QuietSteet* feature bases look "calm" whereas *BusyStreet*, *Tube* and *TubStation* ones are relatively noisy and have many broadband patterns. Overall, these spectral patterns differentiate one acoustic scene from the others well.

4.2. Results

Table 1 summarizes classification accuracy on the AASP development set. The results show that all learning-based features beat MFCCs. Among different pooling methods, event-based pooling achieves higher accuracy than uniform pooling for the same feature size. Furthermore, the combined features from the both pooling methods outperform those from either one.

Table 2 details the performance using confusion matrices. The results show that, in uniform pooling, there is a strong confusion between the audio scenes that have similar sound sources in common. For example, all *OpenAirMarket* and *Restaurant* examples have babble sounds in a consistent manner, and *Tube* and *TubeStation* examples have train engine sounds as a dominant source. The second confusion matrix shows that event-based pooling discriminates these audio scenes better than uniform pooling, indicating that the proposed method successfully emphasizes characteristic sound events other than such common sound sources. However, event-based pooling has side effects as well. For example, more *QuietStreet* examples are classified into *BusyStreet* (from 2 to 5) due to

	Feature Size	Mean (%)	Standard Dev.
MFCC + GMM		54.4	8.94
Uniform	256	63	7.64
	512	65.4	8.15
Event-based	256	67	8.42
	512	68.6	7.97
Combined	256+256	70	8.29
	512+512	72	7.63

Table 1: Classification results on the AASP development dataset

	Bus	BS	Off.	OAM	Park	QS	Res.	SM	Tube	TS
Bus	**46**	0	0	4	0	0	0	0	0	0
BusyStreet	0	**46**	0	0	0	3	0	0	0	1
Office	0	0	**42**	0	3	3	1	1	0	0
OpenAirMrkt.	1	0	0	**33**	1	0	11	3	1	0
Park	0	0	7	0	**25**	18	0	0	0	0
QuietStreet	0	2	0	3	8	**36**	0	1	0	0
Restaurant	0	0	5	17	0	0	**23**	3	1	1
SuperMarket	0	0	7	1	0	3	6	**30**	0	3
Tube	3	8	1	0	0	1	3	1	**14**	19
TubeStation	0	0	0	1	1	0	5	0	11	**32**

(a) Uniform Pooling (Feature Size = 512)

	Bus	BS	Off.	OAM	Park	QS	Res.	SM	Tube	TS
Bus	**44**	3	0	3	0	0	0	0	0	0
BusyStreet	1	**47**	0	0	0	0	0	0	0	2
Office	0	0	**37**	0	5	1	0	7	0	0
OpenAirMrkt	1	0	0	**43**	0	1	5	0	0	0
Park	0	0	2	0	**38**	9	0	0	1	0
QuietStreet	0	5	1	1	9	**27**	2	3	0	2
Restaurant	0	0	5	11	0	0	**26**	8	0	0
SuperMarket	0	0	10	2	0	2	9	**26**	1	0
Tube	2	1	2	0	0	3	2	1	**27**	12
TubeStation	1	4	0	1	4	4	5	1	2	**28**

(b) Event-based Pooling (Feature Size = 512)

	Bus	BS	Off.	OAM	Park	QS	Res.	SM	Tube	TS
Bus	**46**	0	0	3	0	0	0	0	1	0
BusyStreet	0	**50**	0	0	0	0	0	0	0	0
Office	0	0	**43**	0	1	1	1	3	1	0
OpenAirMrkt	1	0	0	**41**	0	0	6	2	0	0
Park	0	0	4	0	**38**	8	0	0	0	0
QuietStreet	0	2	0	3	11	**33**	0	1	0	0
Restaurant	1	0	5	12	0	0	**29**	3	0	0
SuperMarket	0	0	6	4	1	1	6	**28**	0	4
Tube	1	4	0	0	0	2	4	1	**24**	14
TubeStation	0	0	0	4	5	2	5	0	6	**28**

(c) Combined (Feature Size = 512+512)

Table 2: Confusion matrix

the emphasis on transient events. The problem is resolved by using the both methods as shown in the last confusion matrix for the combined features. Not only that, but also the combined features improve accuracy for most classes of acoustic scenes, achieving synergy between the two pooling methods.

In the the AASP Challenge, we achieved a 60% accuracy with event-based pooling and a 68% accuracy with the combined features.[2] These results are somewhat lower than those on the development set. However, the difference is closer to or within the standard deviation in Table 1.

5. CONCLUSION

We presented acoustic scene classification algorithms using sparse local feature learning and a novel feature summarization. We showed that the feature bases learned from data capture salient spectral features of acoustic scenes. We also proposed an event-based pooling method to summarize strong local events selectively. Through the experiments with real-world environmental sounds, we demonstrated that the learned features outperform MFCCs, a popularly used hand-engineered feature. Also, we showed that the proposed event-based pooling discriminates acoustic scenes with similar sound sources better than uniform pooling and, furthermore, the combined features achieved the best results.

6. REFERENCES

[1] H. Lee, Y. Largman, P. Pham, and A. Y. Ng, "Unsupervised feature learning for audio classification using convolutional deep belief networks," in *Advances in Neural Information Processing Systems 22*, 2009, pp. 1096–1104.

[2] M. Henaff, K. Jarrett, K. Kavukcuoglu, and Y. LeCun, "Unsupervised learning of sparse features for scalable audio classification," in *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR)*, 2011.

[3] J. Wülng and M. Riedmiller, "Unsupervised learning of local features for music classification," in *Proceedings of the 13th International Conference on Music Information Retrieval (ISMIR)*, 2012.

[4] J. Nam, J. Herrera, M. Slaney, and J. O. Smith, "Learning sparse feature representations for music annotation and retrieval," in *Proceedings of the 13th International Conference on Music Information Retrieval (ISMIR)*, 2012.

[5] R. F. Lyon, M. Rehn, S. Bengio, T. C. Walters, and G. Chechik, "Sound retrieval and ranking using sparse auditory representations," in *Neural Computation 22(9)*, 2010.

[6] C. V. Cotton and D. P. W. Ellis, "Soundtrack classification by transient events," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011.

[7] H. Lee, C. Ekanadham, and A. Y. Ng, "Sparse deep belief net model for visual area V2," in *Advances in Neural Information Processing Systems 20*, 2008, pp. 873–880.

[8] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, pp. 1527–1554, 2006.

[9] D. Giannoulis, M. R. M. L. E. Benetos, D. Stowell, and M. Plumbley, web resource, available, http://www.elec.qmul.ac.uk/digitalmusic/sceneseventschallenge/.

[2] Our official submission to the AASP Challenge was originally based on event-based pooling alone. Later, we found a bug in the code and also figured out the combined features work better. The result with the combined features was evaluated apart from the challenge.