INTERSPEECH 2017
August 20–24, 2017, Stockholm, Sweden

Audio Classification Using Class-Specific Learned Descriptors

Sukanya Sonowal[1], *Tushar Sandhan*[2], *Inkyu Choi*[3], *Nam Soo Kim*[3]
[1]Samsung Electronics Co. Ltd., South Korea
[2]Department of ECE, Seoul National University, ASRI, South Korea
[3]Department of ECE, Seoul National University, INMC, South Korea
sk.sonowal@samsung.com, tushar@snu.ac.kr, ikchoi@hi.snu.ac.kr, nkim@snu.ac.kr

Abstract

This paper presents a classification scheme for audio signals using high-level feature descriptors. The descriptor is designed to capture the relevance of each acoustic feature group (or feature set like mel-frequency cepstral coefficients, perceptual features etc.) in recognizing an audio class. For this, a bank of RVM classifiers are modeled for each 'audio class'-'feature group' pair. The response of an input signal to this bank of RVM classifiers forms the entries of the descriptor. Each entry of the descriptor thus measures the proximity of the input signal to an audio class based on a single feature group. This form of signal representation offers two-fold advantages. First, it helps to determine the effectiveness of each feature group in classifying a specific audio class. Second, the descriptor offers higher discriminability than the low-level feature groups and a simple SVM classifier trained on the descriptor produces better performance than several state-of-the-art methods.

Index Terms: feature bagging, audio event, representation, classification

1. Introduction

Audio event classification (AEC) or recognition has received increased attention in the recent years. This is mainly because of its role in many applications such as audio retrieval, multimedia surveillance, computer or robotic assistance in a meeting room and ambient assisted living [1]. Several audio features have been used for the classification task. The most widely used among them are the mel-frequency cepstral coefficients (MFCCs) [2, 3, 4] and the perceptual features consisting of brightness, bandwith, sub-band energies, zero-crossing rate, pitch etc. [1, 5, 6]. Other features like gammatone-frequency cepstral coefficients (GFCCs) [7], linear prediction coefficients (LPCs) [7, 8] and perceptual linear prediction (PLP) features [9] have also been used with considerable success.

As audio events are generated from a wide variety of sources (e.g. phone ring, door movement, water tap etc.) the spectral structures of the audio signals vary immensely from one audio class to another. Also each of the audio feature groups (or feature set) are crafted to convey a specific set of information about an audio signal, which may be fully or partially expressed in the samples of some audio class while it may not be expressed at all in the samples of some other audio class. For example, LPC features are particularly good at estimating spectral peaks and can be more effective in representing tonal sounds (like spoon/cup jingle sounds) than the MFCC features. At the same time, LPC features may be less effective in representing noise-like unstructured signals (like cough, chair movement sounds) [8]. In other words, the audio features will not have the same discriminative power for all the classes [10].

One of the ways to address this problem is to concatenate all the feature groups to represent the signal. However, this worsens the 'curse of dimensionality'; as a single classifier model is overloaded with a lot of features. Applying feature selection techniques helps remove the 'noisy' or 'irrelevant' features but they also carry the risk of removing valuable features which may seem less important [11]. Recent approaches for audio signal representation have been focused on building a set of learned descriptors (e.g. representations using bag-of-words model [7, 12], object detectors, non-negative matrix factorization [1] etc.) for capturing the inherent structures of an audio signal.

In this paper, we apply an ensemble-based learning approach to the audio classification task by modeling a separate classifier for each feature group. We apply the one-vs-rest strategy in modeling the classifier for each feature group, which effectively results in one classifier model for an 'audio class'-'feature group' pair. An audio signal is then represented by concatenating the response of each model (in the classifier bank) to the input signal. The resulting signal representation is semantically rich as it measures the proximity of the input signal to each audio class based on a certain feature space. This method of creating the classifier bank offers many advantages. First, the classifier is not overloaded with many diverse features as each individual classifier is modeled using only a single feature group. Second, we utilize all of the feature groups to build the classifier bank and so do not discard any seemingly 'noisy' features. Also the representation is more useful in determining the significance of each feature group in classifying an audio class.

We use the relevance vector machine (RVM) classifier to model the individual classifiers in the classifier bank. The response of the classifier bank thus consists of the output probability scores provided by each RVM model. The signal representation consisting of the responses of the RVM bank is then fed to a simple SVM classifier to predict the class labels. Experimental results show that our proposed signal representation achieves better classification performance than prior art and other feature selection based methods on various datasets.

2. Method

The system overview is described in Figure 1. An audio signal is decomposed into small non-overlapping frames and features corresponding to each feature group are extracted for all frames in the signal. In the next stage, class-specific RVM classifiers are modeled on each of the feature groups, the outputs of which form the frame-level feature descriptor. The descriptors are then fed to a pair of SVM classifiers which helps in predicting the

http://dx.doi.org/10.21437/Interspeech.2017-982

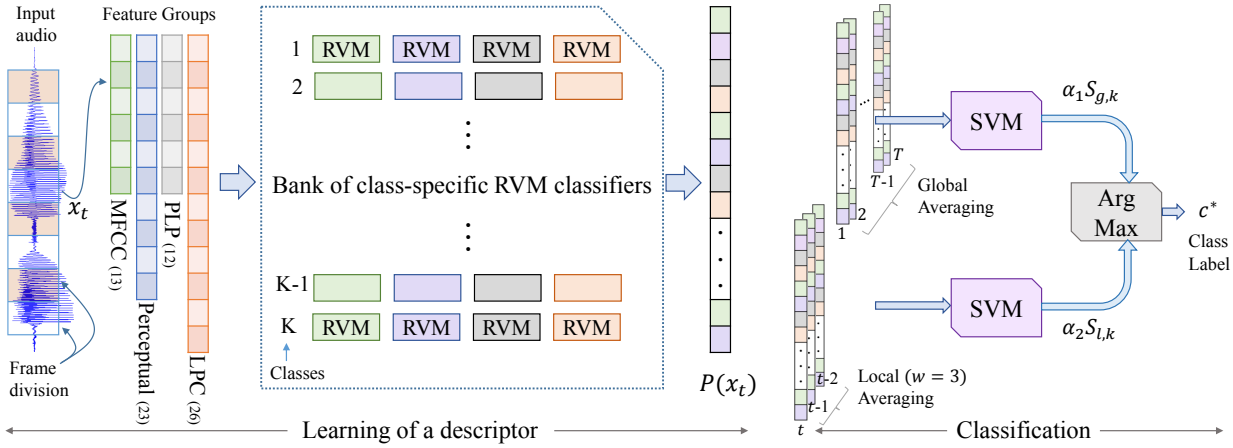

Figure 1: *Overview of the proposed system describing construction and integration of the class specific learned descriptors.*

class labels. The final pooling step combines frame-level and event-level SVM responses to assign the class label.

2.1. Feature groups

We extract the four feature groups for each audio frame:

- **MFCC** features consisting of 12 MFCCs and the zero-th order term.
- 23-dimensional **Perceptual** features which consist of zero-crossing rate, sixteen sub-band energy ratios, brightness, bandwidth, pitch, spectral slope, spectral flux and spectral roll-off frequency.
- **PLP** features comprising of 12 coefficients.
- **LPC** features consisting of 26 coefficients.

2.2. Class-specific modeling of feature groups using RVM

At this stage, the RVM classifier is applied to model the probability distribution of the audio classes for each feature group. The RVM classifier uses Bayesian inference to provide posterior probability as an estimate of class membership. Moreover, its sparse nature and the use of kernel functions ensure speed and accuracy. In this work, the RVM classifier is employed in a one-vs-rest manner for each feature category.

Assuming K audio classes $c_1,...,c_K$, for the j^{th} feature group inputs we learn K RVM models $p_{1j},...,p_{Kj}$ in accordance with the one-vs-rest technique. The RVM p_{ij} is modeled on samples of the feature group j, considering samples of the class c_i as positive instances and all other samples as negative instances. For a feature input $\mathbf{x}$ from feature group j, the output $p_{ij}(\mathbf{x})$ of the RVM p_{ij} thus denotes the posterior probability of class c_i given $\mathbf{x}$. In other words it provides a probabilistic ('soft') estimate for the output audio class being c_i given the input $\mathbf{x}$.

With the number of our feature groups being four, we learn a total of $4K$ RVM models. The output scores of this bank of $4K$ RVM models is then used for the audio frame-level representation. An audio frame $\mathbf{x}_t$ with inputs $\mathbf{x}_{t1},...,\mathbf{x}_{t4}$ for each feature group, is represented using the bank of RVM models as

$$\mathbf{P}(\mathbf{x}_t) = [p_{11}(\mathbf{x}_{t1}), ..., p_{14}(\mathbf{x}_{t4}), ..., p_{K1}(\mathbf{x}_{t1}), ..., p_{K4}(\mathbf{x}_{t4})]^T, \quad (1)$$

and given in Figure 1. This feature representation provides an insight into the discriminative power of an individual feature group for a specific audio class.

Suppose that the audio frame $\mathbf{x}_t$ belongs to class c_i. The entries $p_{i1},...,p_{i4}$ of $\mathbf{P}(\mathbf{x}_t)$ depict how well each feature group represents the audio class c_i for that frame. On the other hand, entries $p_{1j},...,p_{Kj}$ determine the ability of feature group j to distinguish class c_i from other classes. A consistently high value of the entry p_{ij} for all samples of the class c_i means that the feature group j can well capture the intraclass variations of that class. While a consistently low value of entries $p_{1j},...,p_{i-1j},p_{i+1j},...,p_{Kj}$ means that the feature group j can well separate c_i from other audio classes (i.e. interclass variations are captured here).

This frame-level representation thus offers more semantic information and discriminative power than the low-level audio feature groups. It is learned by using only a portion of the available training data (we used 75% of the training data). The next step involves learning a simple classification model using entire training data on these descriptors to predict the class labels.

2.3. Classification

2.3.1. *Integrating frame-level descriptors*

For integrating the frame-level descriptors before SVM classification we follow two approaches: *global averaging* and *local averaging*. A set of one-vs.-rest K SVM classifiers are then modeled independently on each feature set generated by the integration process.

Global averaging. As implemented in [5, 13], the descriptors are averaged over all frames of the audio signal. This results in one feature vector for one audio event. The SVM trained by this approach outputs one decision score for one audio event and is denoted as the *global*-SVM model.

Local averaging. The audio signal is divided into multiple non-overlapping segments, each consisting of a sequence of w consecutive frames. The feature vector is computed for each audio segment by averaging the w frame descriptors within the segment. The SVM decisions are thus obtained for each segment of the audio event. The segment-wise SVM decision scores are then averaged over the entire signal to produce the final score. This SVM model is denoted as the *local*-SVM model.

2.3.2. *Weighted average decision*

The scores obtained from each of the SVM classifiers above are then combined for assigning the class label to the audio event. We use a weighted average approach to combine the scores.

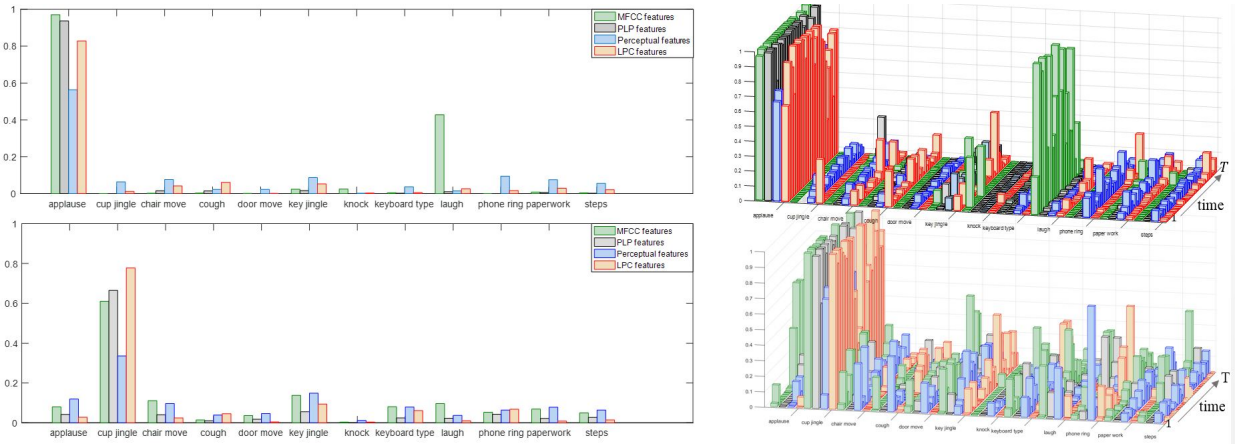

Figure 2: *Global (shown left) and local (shown right) descriptors for the audio events "applause" (on top of) "key jingle" respectively.*

For an audio event X consisting of L segments, let $S_{g,k}(X)$ be the decision score from the *global*-SVM model of class c_k's and $S_{l,k}(X)$ be the decision score from the *local*-SVM model of class c_k's, then final score $S_k(X)$ for class c_k is computed as,

$$S_k(X) = \alpha_1 S_{g,k}(X) + \alpha_2 S_{l,k}(X). \quad (2)$$

This classification process is summarized in Figure 1. Finally, the most likely class c^* is chosen as

$$c^* = \arg\max_k S_k(X) \quad (3)$$

3. Experiments

3.1. Experimental Setting and Dataset description

Parameters. The audio files were downsampled to 16 kHz and decomposed into 30 ms non-overlapping frames. For local averaging operation, the size w of the audio segment was varied from 3 to 15 frames (i.e. from 90 ms to 450 ms) to observe its effect on the classification performance. The RVM models were trained on 75% of the training data and a RBF kernel was used for the RVM training. For training both the *local*-SVM and the *global*-SVM, all of the training data was used.

Datasets. We evaluated the proposed method on two datasets *UPC-TALP* [14] and *Freiburg-106* [15]. The datasets are summarized below:

1. *UPC-TALP*. It contains 14 classes of audio events that occur in a meeting room environment. The total number of audio files in this dataset is 1,030. As in [1], we merged the classes "door open" and "door close" to form the "door move" class because both the classes are perceptually similar (due to presence of door slam sound in both classes). We also removed the class "unknown" from our experiments. The dataset finally contains 12 audio classes and 904 instances.
2. *Freiburg-106*. It consists of 24 classes of audio events recorded in kitchen and bathroom environments. The total number of instances is 1507.

Each dataset offers many different challenges. The number of classes in UPC-TALP is small but it contains similar number of instances per class and on average, the number of instances per audio class is high. On the other hand Freiburg-106 has more audio classes and the number of instances per class varies greatly with the lowest value being 22 (for class "cup") and the highest being 135 (for class "plates-sorting").

In each run of the experiment, 70% of the dataset samples were randomly assigned as training data and the rest were assigned as test data. This process was repeated for five runs and the results were reported on the average over all the runs.

3.2. Proposed system and comparisons

Proposed system. To observe the effect of averaging on our descriptors we evaluated the individual performances of both the *local*-SVM model (SVM-local) and the *global*-SVM model (SVM-global). We compared their performances along with the performance of the weighted average decision system (SVM-weighted). For SVM-weighted, we report the best performance among the combination of weights α_1 and α_2.

Comparative methods. We compared our performance with four different methods:

1. *SVM-all*: In this system, we formed a super vector by concatenating features from all the four feature groups. The resulting dimension D_{all} of the super-vector is 74. This super-vector was directly fed to the SVM classifier for predicting the class labels.
2. *SVM-fisher*: We applied Fisher's criteria to rank the features in the super-vector. From the resulting rank-ordered features, we selected top D_f ($\leq D_{all}$) features. The selected D_f features were then used for SVM classification.
3. *SVM-IG*: We applied Information gain criteria for ranking and selected top D_i ($\leq D_{all}$) features. The selected D_i features were used for SVM classification.
4. *Bag-of-Super-Features system (BoSF)*: We applied the method as described in [7]. In this method codebooks are built for all classes and concatenated to form a super-codebook. This super-codebook is used for quantizing the input features and the histogram of the quantized features is used as input to the classifier.

For SVM-all, SVM-fisher and SVM-IG methods we applied the same weighted average decision for predicting the class label. So, we trained *local*-SVM and *global*-SVM models separately for each system and assigned the class label based on weighted

Table 1: *F-score and classification accuracy achieved by different methods on the UPC-TALP and Freiburg-106 datasets. SVM-local, SVM-global and SVM-weighted are the respective results of the local-SVM model, global-SVM model and the weighted average decision on our descriptors. The results are compared alongside the comparative methods (SVM-all, SVM-fisher, SVM-IG and BoSF).*

	Dataset	SVM-all	SVM-fisher	SVM-IG	BoSF	SVM-local (ours)	SVM-global (ours)	SVM-weighted (ours)
f-score	UPC-TALP	94.01%	94.48%	94.47%	92.43%	95.63%	95.22%	**96.34%**
	Freiburg-106	96.78%	96.66%	96.63%	96.29%	95.39%	96.35%	**96.82%**
classification accuracy	UPC-TALP	94.14%	94.62%	94.62%	91.85%	95.66%	95.28%	**96.34%**
	Freiburg-106	96.44%	96.5%	96.36%	95.79%	95.6%	96.2%	**96.76%**

average of their scores. We also report the best performance over all values of D_f and D_i for SVM-fisher and SVM-IG.

The BoSF method [7] has applied GMM clustering for building the codebook of each class on the stacked MFCC and GFCC features. Quantization is based on the probability distribution function of the multivariate Gaussian for each cluster and histogram has been obtained by simple averaging over a window. Then the class labels are predicted by multinomial maximum likelihood classifier as it showed better performance than many other classifiers. We have used the original implementation provided by the authors.

3.3. Results

For the UPC-TALP dataset, the size of our descriptor is 48 while for the Freiburg dataset, the descriptor size is 96. Figure 2 shows the descriptors for audio examples of classes "applause" and "cup jingle" of the UPC-TALP dataset. The outputs of local averaging and global averaging are separately plotted for these two audio examples. The descriptors in general show good discriminative properties however, with varying magnitudes depending on each "feature group"-specific entry. For class "cup jingle" the entry related to the response of the same class's RVM model trained on LPC features has the maximum value. For classes "key jingle" and "applause", the entry related to the RVM model trained on MFCC features has the maximum value. We also tested linear and RBF kernels for the *local*-SVM and *global*-SVM models and found the RBF kernel to be better performing. The results are thus obtained using the RBF kernel.

Table 1 shows the classification accuracies and the f-score values of the competitive methods and our proposed system for the two datasets. Among our systems, the weighted decision approach (SVM-weighted) performs better than global averaging (SVM-global) and local averaging (SVM-local) approaches. It is interesting to observe that the SVM-global system outperforms the SVM-local system for Freiburg-106 dataset but for the UPC dataset, SVM-local performs better. Our SVM-weighted also outperforms the rest of the methods. For the UPC dataset, it shows 1.86% f-score improvement and 1.72% accuracy improvement over its best competitor (SVM-fisher). While for the Freiburg dataset, it shows f-score and accuracy improvements of 0.05% and 0.32% respectively over its best competitor (SVM-all). This shows that our proposed descriptors can show better classification performance than the system which uses all the low-level features combined or systems using usual feature selection techniques on the low-level features. Our descriptors also show significant performance improvement over the bag-of-features system which also consists of learned descriptors.

4. Conclusion

In this paper, we have proposed a high-level feature descriptor for audio classification. The descriptor determines each feature group's ability to distinguish a specific audio class from other classes. It also determines the feature group's ability to capture the variations within the samples of an audio class. The descriptor consists of the output responses of a bank of RVM classifiers, where each classifier is modeled for a specific audio class and a feature group. Experimental results show that the simple SVM classification on our proposed descriptors show better results compared to previous methods on various datasets.

5. References

[1] T. Sandhan, S. Sonowal, and J. Y. Choi, "Audio bank: A high-level acoustic signal representation for audio event recognition," in *14th International Conference on Control, Automation and Systems (ICCAS 2014)*, 2014, pp. 82–87.

[2] C. Zieger and M. Omologo, "Acoustic event classification using a distributed microphone network with a gmm/svm combined algorithm," in *Proc. Interspeech*, 2008.

[3] H. Phan, L. Hertel, M. Maass, R. Mazur, and A. Mertins, "Representing nonspeech audio signals through speech classification models," 2015.

[4] P. S. Huang, R. Mertens, A. Divakaran, G. Friedland, and M. Hasegawa-Johnson, "How to put it into words - using random forests to extract symbol level descriptions from audio content for concept detection," in *IEEE ICASSP*, 2012, pp. 505–508.

[5] A. Temko, R. Malkin, C. Zieger, D. Macho, C. Nadeu, and M. Omologo, "Clear evaluation of acoustic event detection and classification systems," 2006.

[6] H. Phan, M. Maass, L. Hertel, R. Mazur, I. McLoughlin, and A. Mertins, "Learning compact structural representations for audio events using regressor banks," in *IEEE ICASSP*, 2016.

[7] A. Plinge, R. Grzeszick, and G. A. Fink, "A bag-of-features approach to acoustic event detection," in *IEEE ICASSP*, 2014.

[8] B. Pinkowski, "Lpc spectral moments for clustering acoustic transients," *IEEE Transactions on Speech and Audio Processing*, vol. 1, no. 3, pp. 362–368, 1993.

[9] S. Kamath, S. Ravindran, and D. V. Anderson, "Independent component analysis for audio classification," in *2004 Digital Signal Processing Workshop*, 2004.

[10] P. Gehler and S. Nowozin, "On feature combination for multiclass object classification," in *2009 IEEE 12th International Conference on Computer Vision*, 2009, pp. 221–228.

[11] M. A. Aly and A. F. Atiya, "Novel methods for the feature subset ensemble approach," pp. 1–7, 2006.

[12] S. Pancoast and M. Akbacak, "Softening quantization in bag-of-audio-words," in *IEEE ICASSP*, 2014, pp. 1370–1374.

[13] J.-C. Wang, J.-F. Wang, C.-B. Lin, K.-T. Jian, and W. Kuok, "Content-based audio classification using support vector machines and independent component analysis," in *ICPR*, 2006.

[14] T. Butko, C. Canton-ferrer, C. Segura, C. Nadeu, J. Hern, and J. R. Casas, "Acoustic event detection based on feature-level fusion of audio and video modalities," 2011.

[15] J. A. Stork, J. Silva, L. Spinello, and K. Arras, "Audio-based human activity recognition with robots," in *International Conference on Social Robotics (ICSR)*, 2011.