SPECTRAL VS. SPECTRO-TEMPORAL FEATURES FOR ACOUSTIC EVENT DETECTION

Courtenay V. Cotton and Daniel P. W. Ellis

LabROSA, Dept. of Electrical Engineering
Columbia University
1300 S. W. Mudd, 500 West 120th Street,
New York, NY 10027, USA
{cvcotton,dpwe}@ee.columbia.edu

ABSTRACT

Automatic detection of different types of acoustic events is an interesting problem in soundtrack processing. Typical approaches to the problem use short-term spectral features to describe the audio signal, with additional modeling on top to take temporal context into account. We propose an approach to detecting and modeling acoustic events that directly describes temporal context, using convolutive non-negative matrix factorization (NMF). NMF is useful for finding parts-based decompositions of data; here it is used to discover a set of spectro-temporal patch bases that best describe the data, with the patches corresponding to event-like structures. We derive features from the activations of these patch bases, and perform event detection on a database consisting of 16 classes of meeting-room acoustic events. We compare our approach with a baseline using standard short-term mel frequency cepstral coefficient (MFCC) features. We demonstrate that the event-based system is more robust in the presence of added noise than the MFCC-based system, and that a combination of the two systems performs even better than either individually.

Index Terms— Acoustic signal processing, acoustic event detection, acoustic event classification, non-negative matrix factorization

1. INTRODUCTION

Detection and classification of acoustic events is important in a number of applications. In particular, we are motivated to examine this problem in the context of automatically finding and tagging events that occur in an unconstrained environmental audio stream, such as the soundtrack of a YouTube video.

Standard approaches to acoustic event detection utilize sequences of feature vectors that each capture spectral information over a very short time window, e.g., 30 ms. Additional modeling of temporal structure, for example a hidden Markov model (HMM), may then be used to reincorporate the temporal context that has been lost. In contrast, we are interested in taking advantage of temporal context directly when detecting and comparing acoustic events, rather than trying to describe events only by the statistics of their component frames. Intuitively, we would think of an acoustic event as something that is defined by both its spectral energy and its characteristic temporal shape.

In [1] we explore an approach to discovering and comparing acoustic events with their temporal context. We do this by extracting spectro-temporal patches around "transient" points that exhibit a large increase in spectral energy. However, this method can be very sensitive to the parameters of the transient detector, and therefore have poor robustness to environmental factors. The extracted features also suffer because there is no separation between an event's energy and the background noise floor, which will also be captured in the surrounding spectro-temporal patch.

To address these problems, we present an algorithm based on non-negative matrix factorization (NMF). NMF is an algorithm for describing data as the product of a set of bases and a set of activations, both of which are non-negative; it was first described in its current form in [2]. It is useful for finding parts-based decompositions of data; since all components are non-negative, each basis contributes only additively to the whole. This promotes a solution in which high-energy foreground events and constant low-level background energy may be described by different bases, and therefore separated in the feature representation.

Most applications of NMF to audio processing decompose spectral magnitude frames (e.g., columns of a spectrogram), and have each NMF bases consist of a single short time frame [3]. Since we are interested in learning bases that correspond to entire events, we use the convolutive formulation of NMF [4, 5]. In this version, bases consist of spectro-temporal patches of a number of spectral frames stacked together. The pattern described by each frame is then activated as a whole to contribute to the reconstruction of the data. Additionally we would like to ensure some level of sparsity in the activations of these bases. This is in order to encourage the bases to learn more foreground event patterns and fewer patterns that mimic the background, which would be activated non-sparsely over large segments of the data.

This NMF algorithm allows us both to locate transients in time, and to learn a dictionary of event-patch codewords, within a single optimization framework, avoiding the separate transient detection and patch clustering of our earlier approach.

2. EVENT DETECTION WITH CONVOLUTIVE NMF

Our algorithm for the detection of acoustic events is as follows. We downsample all data to 12 kHz, and take a standard STFT or spectrogram of the entire signal, using 32 ms frames and 16 ms hops. We warp the frequency axis to mel-frequency as though we were taking MFCC features, using 30 mel-frequency bands from 0 to 6 kHz. We then concatenate all training data spectrograms and per-

This work was supported Eastman Kodak Corp., by the National Geospatial Intelligence Agency, and by the IARPA Aladdin program.

form convolutive NMF across the entire set of training data, using 20 basis patches which are each 32 frames wide (approximately 500 ms). This produces a set of basis patches W and a set of basis activations H. The number of bases was tuned for performance; this also seems like a reasonable number to use since we are modeling data containing 16 classes of acoustic events, as described in section 5. In our experiments we use the implementation of sparse convolutive NMF described in [4], with KL divergence as the objective and a sparsity parameter $\lambda = 1$. For testing, we then use the fixed set of learned patch bases and perform NMF to find the corresponding activation values for the test files. An example set of 20 patch bases learned from training data is shown in figure 1. Figure 2 shows an example event ('door knock') and the activation patterns of the three bases that contributed the most energy in representing it. It also shows the event as reconstructed by NMF using the full set of bases. Note that the activations appear to occur somewhat before the onset of energy; this is because the patch activation is placed at the left-hand edge of the 500 ms patch.

We believe the NMF algorithm captures a relevant set of event-like patches, but we still need a reasonable way to represent the (continuous) activation patterns of these bases as discrete event-like features. In order to do this, we use a sliding window of 1 s, with hops of 250 ms. Within this window, we summarize the local basis activation pattern by taking the log of the maximum of each activation dimension, producing a set of 20 features per window. These activation values are pre-normalized such that each basis has a maximum activation of 1 over the entire dataset.

In order to perform event detection with these features, we use a simple HMM. The dataset we use for evaluation (described in section 5) labels specific time intervals as containing acoustic events of a given class. Our HMM consists of a single state for each of the 16 classes and a 17th state for the background. The observation matrix is trained using the interval labels of the training data, with the simple assumption that the observations for each event class can be modeled as a single Gaussian. The transition matrix is trained on the stream of labels, which in practice prohibits direct transitions between two classes other than background (since the training data has no overlap or adjacency between different classes). A stream of predicted labels is produced for each test file and scored as explained in section 6. Finally, in order to produce a reliable event stream, events shorter than 6 frames are removed.

3. BASELINE MFCC EVENT DETECTOR

In order to evaluate our algorithm, we compare it with a baseline which has a similar HMM structure, but that employs a standard set of short-term MFCC features. We extract MFCC features from all data (25 ms frames with 10 ms hops, 40 mel-frequency bands) and retain 25 coefficients as features. We feed this into an HMM, trained in the same way described above. Again, we use a single state for each class, plus a background state, and observations are modeled as a single Gaussian for each class. We generate a series of predicted labels at the frame level, as above.

Because the MFCC frame spacing is much shorter than the NMF system's frame spacing (10 ms vs. 250 ms), we need to post-process the predicted label stream to evaluate it fairly. We first use a median filter across 250 frames (2.5 sec), and then we remove any remaining events that are less than 100 frames long. Both these parameters were selected to optimize performance on clean test data.

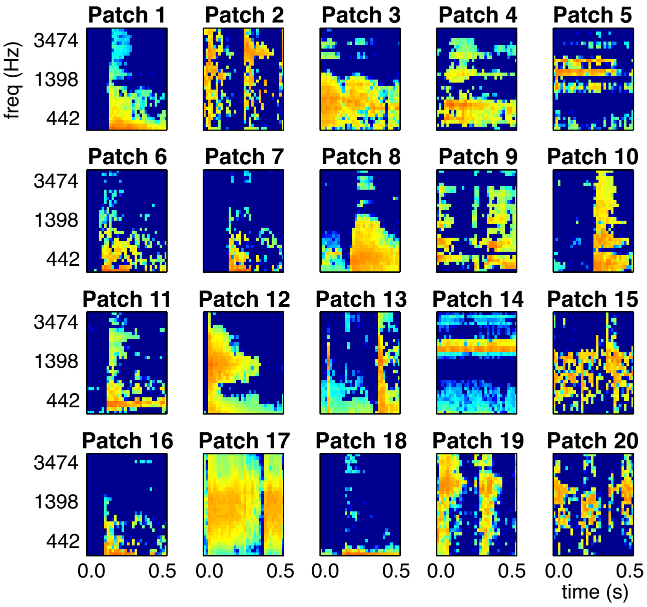

Figure 1: 20 NMF patch bases learned on training data (20 minutes of meeting room sound event recordings).

4. COMBINED SYSTEM

Since we have built two event detectors based on different sets of features, we were interested to see if they could be combined in a complementary way to produce better performance. We did this by taking the two predicted event streams and requiring that they both agree on a predicted event label for some overlapping period of time. If this is true, then we consider that predicted event to extend to the entire period of time that either system has predicted the event. This yields a combined prediction event stream that can be evaluated alongside the two individual systems. Requiring both component systems to agree tends to reduce insertions while increasing deletions; however, it turns out that insertions were the dominant problem in noise, so this approach can be beneficial.

5. DATABASE

In order to focus on the task of detecting specific acoustic events other than speech, we tested our approaches on the FBK-Irst database of isolated meeting-room acoustic events [6]. This is a dataset which was originally collected under the CHIL (Computer in the Human Interaction Loop) project. Event detection and classification using data of this type has been extensively examined by Temko and others [7].

The data we used consisted of 9 sessions, each around 7 minutes long. Each session was recorded by multiple microphones, although we only use one channel in our experiments. This is because we would like to develop algorithms that will also work on less controlled data, such as video soundtracks which would only have one or two channels available.

The database contains 16 semantic classes of acoustic events: door knock; door open; door slam; steps; chair moving; cough; paper wrapping; falling object; laugh; keyboard clicking; key jingle; spoon, cup jingle; phone ring; phone vibration; MIMIO pen buzz;

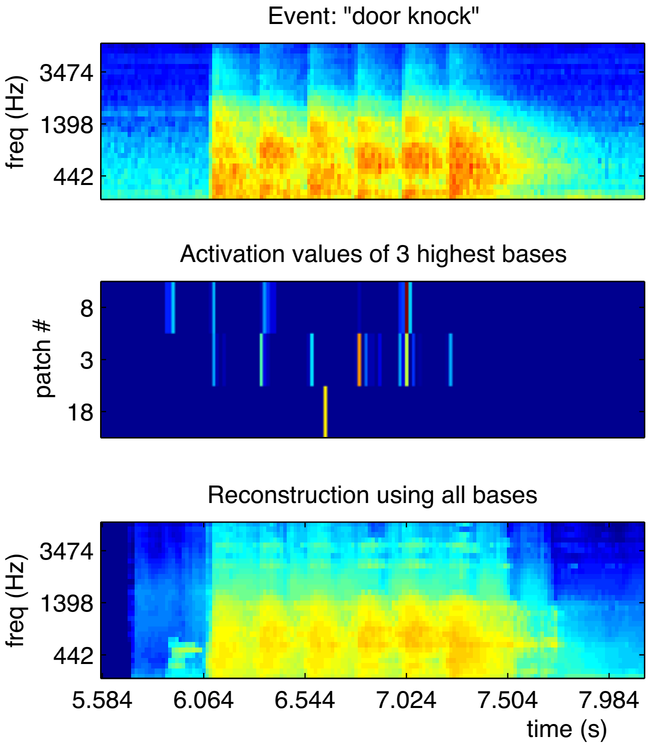

Figure 2: Example of a 'door knock', the top 3 bases used to represent it, and a reconstruction of the event.

and applause. Each session contains around 4 repetitions of each of the 16 classes of events, so there are around 36 examples of each event in the database. Approximately 50 repetitions per event class were recorded.

The data labels consist of short intervals that contain instances of the labeled sound. The events of different classes do not overlap with each other.

In our experiments, we use two folds of the data. In each fold, the data split is 6 training files and 3 test files. For efficiency reasons, in the NMF algorithm only 3 of the training files are used to actually learn the patch bases; the remaining 3 are added back in and used to train the HMM and learn the observation distributions.

6. EXPERIMENT AND METRIC

We are interested in examining the ability of our NMF algorithm to discover acoustic events in the quiet meeting room environment in which this data was recorded, but also in the midst of noisy environments. Our hope is that the additive nature of the NMF algorithm will allow it to represent acoustic events that occur in noise more consistently than standard MFCC features, which will be corrupted by added noise.

In order to test this idea, we performed all experiments with varying levels of additive noise. The noise added was a short clip of background chatter and activity recorded in a cafeteria. For each noise level, the test data was left clean while this noise clip was added to the training data at the specified SNR.

For evaluation we use the acoustic event error rate (AEER) that was used in CHIL evaluations for the event detection task, as de-

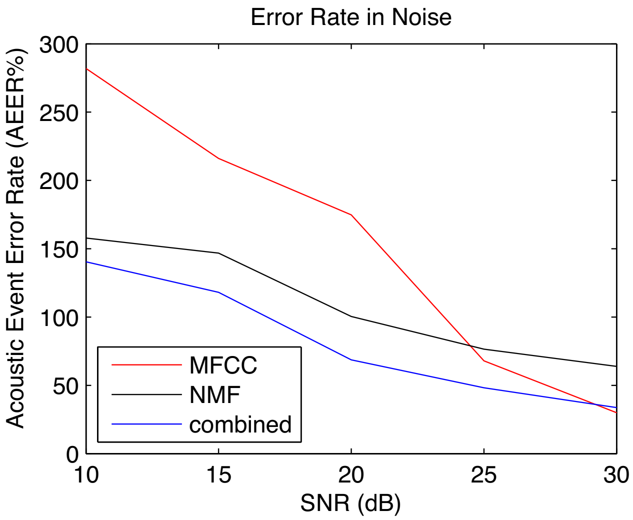

Figure 3: Acoustic event error rate results in noise.

scribed in [7]. This is defined as: $AEER = 100(D + I + S)/N$, where D is the number of deletions, I the number of insertions, S the number of substitutions, and N the total number of events that occur in the ground truth labels.

To evaluate a stream of predicted labels, it is broken into predicted events. A predicted event is any string of consecutive frames with the same label. If the center of a predicted event (of the correct class) falls anywhere within the true event's label interval, then it is considered a correctly predicted event. Any predicted event that does not fall within a true event of the same class is considered an insertion or substitution; we count these errors together. Any true event that does not have a (correctly) predicted event fall within it is considered a deletion.

7. RESULTS

Figure 3 shows the performance of the three algorithms under the AEER metric. Table 1 breaks these results down into deletions, insertions, and again the overall AEER for each algorithm. Each system has been tuned (by balancing insertions and deletions) to optimize its performance at 30 dB SNR (nearly clean noise conditions). In clean conditions, the MFCC-based system performs much better than the event-based one, and about the same as the combination of the two.

We then examine how each system breaks down in the presence of added noise. All three systems produce deletions at a roughly comparable rate as noise increases. The MFCC-based system produces a large number of insertions in high noise conditions, while the NMF-based system largely does not (insertion and deletion rates for NMF stay roughly balanced). The combination system achieves even better performance as the noise level increases. This is mostly true because it is limiting the number of insertions by requiring that the two systems agree on events.

8. DISCUSSION AND CONCLUSIONS

Despite the relative crudeness of our NMF-based features, we demonstrate that these type of large-scale event features can be us-

		SNR (dB)				
		10	15	20	25	30
Deletions	MFCC	41	28	17	13	13
	NMF	55	40	32	26	22
	Combined	54	41	27	21	18
Insertions	MFCC	133	105	91	29	6
	NMF	43	50	29	22	18
	Combined	33	32	15	9	3
AEER	MFCC	282	216	175	68	30
	NMF	158	147	100	77	64
	Combined	140	118	69	48	34

Table 1: Average number of deletions and insertions contributing to AEER.

able in the detection and classification of acoustic events. Although our system is not competitive with a conventional short-frame-based system in clean conditions, it proves useful when the test data is even slightly more noisy than the training data. Features based on NMF basis activations seem to be fairly robust under moderate noise conditions (i.e. both systems using NMF features do not degrade much between 30 and 20 dB SNR). The MFCC-based system, on the other hand, performs much more poorly under moderate noise conditions. Presumably this is because the MFCC features are being corrupted by background noise while the NMF-based system is allowing prominent events to be represented by the same bases as they would have in the clean test data. This would therefore yield feature descriptions that are theoretically more constant as the background noise increases.

Since our interest in event detection extends to varied types of data and recording conditions, it is important for an algorithm to be able to detect similar events that occur in the midst of different types of noise. Event modeling based on convolutive NMF bases seems promising for developing noise-robust of the type necessary to detect acoustic events in all types of unconstrained videos and other audio data.

9. REFERENCES

[1] C. Cotton, D. Ellis, and A. Loui, “Soundtrack classification by transient events,” in *Proc. IEEE ICASSP*, Prague, 2011, p. (to appear).

[2] D. Lee and H. Seung, “Learning the parts of objects by non-negative matrix factorization,” *Nature*, vol. 204, pp. 788–791, 1999.

[3] P. Smaragdis and J. Brown, “Non-negative matrix factorization for polyphonic music transcription,” in *Proc. IEEE WASPAA*, Mohonk, 2003, pp. 177–180.

[4] P. D. O’Grady and B. A. Pearlmutter, “Convolutive non-negative matrix factorisation with a sparseness constraint,” in *Proc. IEEE MLSP*, Maynooth, Sept. 2006, pp. 427–432.

[5] P. Smaragdis, “Convolutive speech bases and their application to supervised speech separation,” *IEEE Tr. Audio, Speech, Lang. Process.*, vol. 15, no. 1, pp. 1–12, 2007.

[6] CHIL, “FBK-Irst database of isolated meeting-room acoustic events,” http://catalog.elra.info/product_info.php?products_id=1093, 2008.

[7] A. Temko, *Acoustic Event Detection and Classification (Ph.D. Thesis)*. Barcelona, Spain: Department of Signal Theory and Communications, Universitat Politecnica de Catalunya, 2007.