Environmental Sound Recognition

Jules Scholler
ENS Paris-Saclay
MSc Mathematics, Vision, Learning
jules.scholler@gmail.com

March 29, 2017

Abstract

This document is a reading report of [1] realized for the audio processing class[1]. It presents an automatic method for recognizing environmental sounds for the understanding of a scene or context surrounding an audio sensor. Theoretical considerations along with experiments are produced to validate their approach based on classical audio features such as Mel-frequency cepstral coefficients (MFCCs) and features emerging from compressed sensing application such as decomposition over redundant dictionaries of waveforms. The paper focuses on ambient context determination using audio by characterizing the general acoustic environment as a whole rather than discrete sound event recognition.

Introduction

Like in the field of pattern recognition in computer vision, selecting proper features is the key to achieve a good accuracy in audio environment classification. The authors main contribution is to add time-frequency features computed by decomposing the audio signal over a redundant dictionary of waveforms and to show that along with traditional audio features (MCFFs) they were able to achieve more than 83% accuracy in predicting the audio environment on a set of 14 classes. In the first part of this report we present the framework proposed in [1]. In the second part we will focus on their results and analyze the weaknesses of their tests. Finally, in the last part we will examine the method interests by giving some applications.

1 Environmental sound recognition framework

1.1 Features extraction

The authors proposed a sparse decomposition on a dictionary D of Gabor[2] (which was the one giving better results, see Figure 1) wavelets $\phi_\gamma,\ \gamma \in \Gamma$ with a fixed number n of wavelets. In other words, given D they want to find the subset of n wavelets $(\phi_i)_{i\in\Gamma}$ that best represents the original signal (e.g. in an L^2 sens). As this is an NP-complete problem the authors introduce the Matching Pursuit algorithm [2] which aims at finding sparse decompositions of a signal efficiently in a greedy manner. The drawback is that MP is suboptimal in the sense that it may not achieve the sparsest solution. In order to exctract MP features they use a rectangular window of 256 points with a 50% overlap. They decompose each 256-point segment using MP with a dictionary of Gabor atoms that are also 256 points in length. They stop the MP process after obtaining $n = 5$ atoms (see Figure 2). Afterwards, they record the frequency and scale parameters for each of these atoms and find the mean and the standard deviation corresponding to each parameter separately, resulting in a four feature vector. The details of the wavelets and dictionary construction is fully detailled in [1] section IV-B. The finality is a redundant dictionary containing $N = 1120$ atoms which makes the MP algorithm complexity linear in terms of signal length which can thus be done in real time.

1.2 Learning

The authors extracted the features on natural sounds (unsynthetised) on 14 different types of environmental sounds among: *Inside restaurants, Playground, Street with traffic and pedestrians, Train passing, Inside moving vehicles, Inside casinos, Street with police car siren, Street with ambulance siren, Nature-daytime, Nature-nighttime, Ocean waves, Running water/stream/river, Raining/shower, and Thundering* (we discuss the class choice in section 2). They used a traditional Gaussian Mixture Model[3] (GMM) classification method where each data class was modeled as a mixture of several Gaussian clusters. Each mixture component is a Gaussian represented by the mean and the covariance matrix of the data. They trained their system by dividing 1-3 minutes long sound clips into 4 s. segments and by averaging the GMM parameters over 100 trials in order to reduce the learning variance.

[1] given by Yves Grenier and Gael Richard

[2] Gabor functions are sine-modulated Gaussian functions that are scaled and translated, providing joint time-frequency localization.

[3] The EM algorithm was then used to find the maximum likelihood parameters of each class.

By combining MP and MFCC features, the authors claim to achieve an averaged accuracy rate of 83.9% by using 5 mixtures for each class. The authors also show by experiments that both MP and MFCC features are useful to recognize the sound environment: depending on the class MP or MCFF features are determinant to obtain a correct prediction.

1.3 Results analysis

In order to understand whether errors are originating from the classifier or the ambiguity of extracted features, the authors used a pairwise classification method to observe the interaction between all possible pairs of classes. Doing that they were able to retrace the origin of errors (e.g. *inside restaurant* and *inside casino* have similar features).

2 Method analysis

2.1 Features

The paper shows that combining time-frequency features can classify sounds where the pure frequency-domain features fail. Also, it can be advantageous to combine both time-frequency and frequency-domain features to improve the overall performance. Even with a small set of features they were able to correctly recognize non-trivial environmental sound with an elegant sparse decomposition. Their features extraction seems to be generalizable to a great variety of classes and could allow further improvements.

2.2 Learning

They model each class with a GMM and looked at the optimal number of mixtures for the whole system (5 is optimal) but they also looked at the optimal number for mixtures for each class (which is ranging between 4 and 6). We don't understand why the accuracy decreases from 83.9% with 5 mixtures for each class to 83.4% with the optimal number of mixtures for each class. To avoid overfitting and for the sake of generalization it is better to have the same number of mixtures for each class but the accuracy should be higher with optimal parameters.[4] Also, there exists a one-sided confusion between *Train* and *Waves*, where samples of Train were misclassified as Waves, but not vice versa. This suggests that their method is non-symmetric and that certain class could be privileged by the algorithm and one can wonder if the method will stay stable in real use.

2.3 Testing

In order to asses the overall performance of a machine learning method it is common practice to compare with human performance. Their test consisted of 140 audio clips from 14 categories, with ten clips from each of the classes. The duration varied between 2, 4, and 6 s. Interestingly, the overall recognition rate was 82.3% which is slightly worse that their method. We think that their test on human is biased for several reasons:

- They did not familiarize testers with their data. It is very difficult for a human to recognize an unknown sound in only 2 seconds. We think that they should have at least make the testers listen to an example in each class giving them the correct class before starting the test.
- Some testers reported that they have never set foot inside a casino. How it is possible to compare an algorithm trained with hundreds of casino sound samples with someone who never went to a casino. This is clearly a way to give an advantage to their method in the performance assessment.
- The class choices are quite questionable. Indeed, they could have chosen 14 classes that are really different, for instance we think that *Inside casino/Inside restaurant, Street with police car siren/Street with ambulance siren* can be labeled as the same class for a proof of concept before trying to increase the number of classes.

3 Applications

The method developed in [1] can be used for robotic navigation, assistive robotics, and other mobile device-based services, where context aware processing is often desired or required. It could be used for determining interior or exterior locations. Knowing the context provides an effective and efficient way to prune out irrelevant scenarios in scene understanding. Other applications include those in the domain of wearables and context-aware applications e.g., in the design of a smartphone that can automatically change the notification mode based on the knowledge of users surroundings, like switching to the silent mode in a theater or classroom.

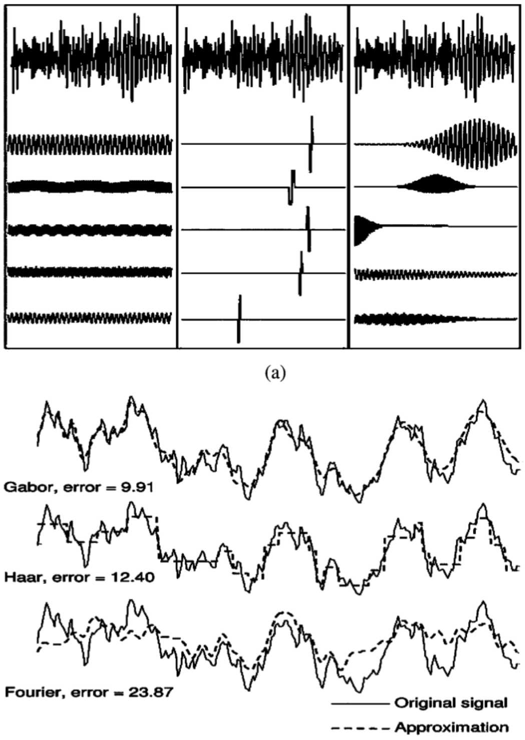
As in computer vision, pattern recognition for audio is a promising approach to draw information from the environment. The main advantage is that its computational complexity is much lower than for vision and it is therefore easier to employ.

Conclusion

Even if the comparison with human performance is questionable the method presented in [1] inspired by pattern theory is a powerful and elegant way to perform sound recognition. The experimental results show promising performance in classifying 14 different audio environments: it provides competitive performance for multi-audio category environment recognition using a comprehensive feature processing approach with a soft category assignment using GMM.

[4] We sent an e-mail to the authors in order to understand their results but as this day it remains unanswered.

Figures



(a)

(b)

Figure 1: (a) Decomposition of signals using MP (the first five basis vectors) with dictionaries of Fourier (left), Haar (middle), and Gabor (right), and (b) approxi- mation (reconstruction) using the first ten coefficients from MP with dictionaries of Gabor (top), Haar (middle), and Fourier (bottom).

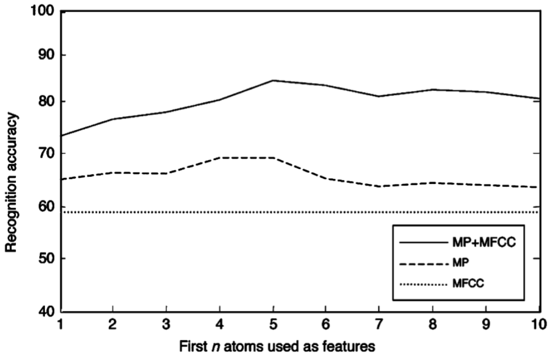


Figure 2: Comparison of classification rates (with the GMM classifier) using the first n atoms, where $n = 1, ..., 10$, as features while the MFCC features are kept the same.

References

[1] S. Chu, S. Narayanan, and C. C. J. Kuo. Environmental sound recognition with time;frequency audio features. *IEEE Transactions on Audio, Speech, and Language Processing*, 17(6):1142–1158, Aug 2009.

[2] S. G. Mallat and Zhifeng Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3415, Dec 1993.