MIREX 2011: AUTOMATIC AUDIO TAG CLASSIFICATION VIA SPARSE CODING

Jia-Min Ren
Department of Computer Science
National Tsing Hua University, Hsinchu, Tawan
jmren@mirlab.org

Kaichun K. Chang
Department of Computer Science
King's College London, London, United Kingdom
ken.chang@kcl.ac.uk

ABSTRACT

This extended abstract details our submission to the Music Information Retrieval Evaluation eXchange (MIREX) 2011 for the audio tag classification task. First of all, we extract a fixed-length feature vector (composed of some timbral as well as modulation spectrum features) from each song clip. Then, by using l^1-reconstruction to represent each test song clip as a linear combination of all training songs (also known as sparse coding), we use the label matrix of training song clips to transform the sparse reconstruction coefficients of each test song clip to the label vector space. Finally, the labels with the largest values are used as the final tags for each test song clip.

1. INTRODUCTION

In recent years, modulation spectral analysis and sparse coding have been attracted much attention in the filed of music/image information retrieval (e.g., audio genre classification [1], face recognition [3], and automatic image annotation [4]). In our system, modulation spectral features such as octave-based modulation spectral contrast (OMSC) [5], modulation spectral flatness measure (MSFM) [5], and modulation spectral crest measure (MSCM) [5] are extracted from each long segment (also named *texture window*). In addition, short-time timbral features such as Mel-scale frequency cepstral coefficient (MFCC), octave-based spectral contrast (OSC), spectral flatness/crest measure, spectral centroid, spectral rolloff, spectral flux, spectral skewness, and spectral kurtosis are extracted from each short segment (also named *analysis window*). Then, we compute the mean and standard deviation along each feature dimension (see Section 2 for more details) to obtain a fixed-length feature vector for each clip. In the annotation stage, we first use sparse coding to represent each test clip as a linear combination of training clips. Then, the label vectors of training clips are used to transform the sparse reconstruction coefficient of each test clip to the label space. Finally, the labels with the largest values (in our case, we selected the top six ones) are treated as the final tags of each test clip. Note that this work is similar to our previous work [6], which used compressive sensing to reduce the dimensionality of features; however, here we simply compute mean and standard deviation along each feature dimension to speed up the computation time.

2. FEATURE EXTRACTION

In our system, we extract short-time timbral features from "*analysis window*", and modulation spectrum features from "*texture window*". For the Majorminer dataset (each song clip is 10 seconds in duration), the length of *analysis window* and *texture window* were set to 93 ms and 5 seconds (with half overlapping), respectively. For the mood dataset (each song clip is 30 seconds in duration), these two lengths were set to 93 ms and 10 seconds (also with half overlapping), respectively.

The following describe the extracted timbral features from *analysis windows* of a segment (the number in each parenthesis is the dimensionality of extracted features).

Mel-scale Frequency Cepstral Coefficients (MFCCs) (13): represents the spectral characteristics based on Mel-frequency scaling.

Octave-based Spectral Contrast (OSC) (16): considers the spectral peak and valley in each sub-band independently, where the former corresponds to harmonic components and the latter corresponds to non-harmonic components or noise in music signals. We extracted spectral peaks and the difference between spectral peak and valley (this difference also named *spectral contrast*, reflecting the spectral contrast distribution) from eight sub-bands [1].

Spectral Flatness/Crest Measure (16): measures of the noisiness (flat, decorrelation) sinusoidality of a spectrum, where the former is computed by the ratio of the geometric mean to the arithmetic mean of the energy spectrum value in each sub-band, and the latter is computed by the ratio of the maximum value within each sub-band to the arithmetic mean of the energy spectrum value [7]. Totally eight sub-bands as set in extracting OSC features were used here.

Spectral Centroid (1): the centroid of amplitude spectrum.

Spectral Rolloff (1): the frequency bin below which 85% of the spectral distribution is concentrated.

Spectral Flux (1): the squared difference of successive amplitude spectrum.

Spectral Skewness (1): a measure (the 3rd order moment) of the symmetry of the spectral distribution.

Spectral Kurtosis (1): a measure (the 4th order moment) of the flatness of the spectral distribution.

To summarize the feature vectors extracted from each song clip, the mean and standard deviation along each feature dimension are computed, resulting in a 100-dimensional feature vector for each song clip.

The following describe modulation spectral features extracted from *texture window* of a song clip (the number in each parenthesis denotes the dimensionality of extracted features).

Octave-based Modulation Spectral Contrast (OMSC) (16x12): this feature is extracted using long-term modulation spectral analysis [8], resulting in a two-dimensional joint acoustic frequency and modulation frequency. Here we computed modulation spectral peak and modulation spectral contrast in six sub-bands to obtain a matrix of size 16-by-12.

To capture the frequency variable (acoustic frequency), we computed mean and standard deviation for each row of this matrix. On the other hand, in order to capture time-varying information through temporal modulation (modulation frequency), we computed mean and standard deviation for each column of this matrix. After these two operations, we obtain an 88-dimensional (24+64) feature vector for each song clip.

Modulation Spectral Flatness/Crest Measure (MSFM/MSCM) (8/8): these two features can be used to describe the time varying behavior of the subband energy. A detailed explanation of MSFM and MSCM can be found in [5].

In summary, totally a 204-dimensional feature vector is extracted from each song clip (100-dimensional features from *analysis windows* and 104-dimensional features from *texture windows*).

3. SPARSE CODING FOR AUTOAMTIC MUSIC ANNOTATION

This extended abstract uses sparse coding for automatic music annotation. Similar concept can be found in [4], which applied sparse coding to image annotation. Like images, although music clips may contain quite similar parts, they may also have different parts. Therefore, using sparse coding to reconstruct the *one-to-all* semantic relation among a query song clip and the training song clips is more straightforward than applying traditional *one-to-one* similarity measure (e.g., comparing the similarity between a query song clip and a training song clip) on the derived feature vectors [4]. The sparse coding algorithm for automatic music annotation is described as follows:

Step 1: Concatenate all N training song clips $y_i \in \mathbb{R}^d \, (i = 1, 2, ..., N)$ to form a matrix $Y = [y_1, y_2, ..., y_N]$.

Step 2: Obtain the l^1-reconstruction coefficients $\alpha^t \in \mathbb{R}^N$ of a test song clip $y^t \in \mathbb{R}^d$ over all training song clips by solving the following optimization problem,

$$\min \left\| \alpha^t \right\|_1 \text{, subject to } y^t = Y\alpha^t .$$

Step 3: Propagate the label matrix of the training song clips (C) to obtain c^t (the level vector of y^t) as:

$$c^t = C\alpha^t .$$

Step 4: Select the labels with the top six values in c^t as the final tags of y^t.

4. REFERENCES

[1] C.-H. Lee, J.-H. Shih, K.-M. Yu, and H.-S. Lin, “Automatic Music Genre Classification Based on Modulation Spectral Analysis of Spectral and Cepstral Feature,” *IEEE Trans. Multimedia*, Vol. 11, No. 4, pp. 670–682, 2009.

[2] Y. Panagakis and C. Kotropoulos, “Music Genre Classification via Topology Preserving Non-Negative Tensor Factorization and Sparse Representations,” in *Proceedings of ICASSP*, pp. 249–252, 2010.

[3] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. S. Huang, and S. Yan, “Sparse Recognition for Computer Vision and Pattern Recognition,” in *Proceedings of the IEEE*, Vol. 98, No. 6, pp. 1031–1044, 2010.

[4] C. Wang, S. Yan, L. Zhang, and H.-J. Zhang, “Multi-Label Sparse Coding for Automatic Image Annotation,” in *Proceeding of IEEE CVPR*, pp. 1643–1650, 2009.

[5] D. Jang, M. Jin, and C. D. Yoo, “Music Genre Classification Using Novel Features and A Weighted Voting Method,” in *Proceedings of ICME*, pp. 1377–1380, 2008.

[6] K. K. Chang, J.-S. Roger Jang, and C. S. Iliopoulos, “Music Genre Classification via Compressive

Sampling," in *Proceedings of ISMIR*, pp. 387–392, 2010.

[7] G. Peeters, "A large set of audio features for sound description (similarity and classification) in the CUIDADO project," *CUIDADO I.S.T Project Report*, 2004.

[8] T. Kinnunen, "Joint Acoustic-Modulation Frequency for Speaker Recognition," in *Proceedings of ICASSP*, pp. 14–19, 2006.