SPARSE CODING BASED MUSIC GENRE CLASSIFICATION USING SPECTRO-TEMPORAL MODULATIONS

Kai-Chun Hsu **Chih-Shan Lin** **Tai-Shih Chi**
Department of Electrical and Computer Engineering, National Chiao Tung University, Taiwan
{kch610596, g104972}@gmail.com, tschi@mail.nctu.edu.tw

ABSTRACT

Spectro-temporal modulations (STMs) of the sound convey timbre and rhythm information so that they are intuitively useful for automatic music genre classification. The STMs are usually extracted from a time-frequency representation of the acoustic signal. In this paper, we investigate the efficacy of two kinds of STM features, the Gabor features and the rate-scale (RS) features, selectively extracted from various time-frequency representations, including the short-time Fourier transform (STFT) spectrogram, the constant-Q transform (CQT) spectrogram and the auditory (AUD) spectrogram, in recognizing the music genre. In our system, the dictionary learning and sparse coding techniques are adopted for training the support vector machine (SVM) classifier. Both spectral-type features and modulation-type features are used to test the system. Experiment results show that the RS features extracted from the log. magnituded CQT spectrogram produce the highest recognition rate in classifying the music genre.

1. INTRODUCTION

For a classification task, selected features and the classifier are critical to the performance of the system. Since the last decade, lots of researchers have proposed music genre classification systems using designed features or classifiers. For instance, the mel-frequency cepstral coefficients (MFCCs), the pitch histogram and the beat histogram were used in [1] as effective features to describe characteristics of timbre, pitch and rhythm of music. The SVM was used in a multi-layer fashion for genre classification [2]. Later on, parameters of autoregressive models of spectral raw features were used for classification by including the temporal variations of the raw features [3]-[5]. In addition to SVM, the adaptive boosting algorithm was used to train the classifier [6]. The non-negative tensor factorization (NTF) was also considered to reduce the dimensionality in a sparse representation classifier (SRC) [7][8]. Another approach was to extract features from the separated cleaner signal [9] by first applying the harmonic-percussion signal separation (HPSS) algorithm [10] to the music clip. The Gaussian supervector, which has been successfully used in speaker identification, was also investigated in music genre classification [11]. A supervised dictionary learning process was proposed for genre classification by using codebooks generated from existing coding techniques [12]. All these methods were operated on audio signals only. In addition, one can also combine features from other sources such as MIDI or lyrics [13]-[17].

In recent years, sparse coding technique has been applied to music genre classification. Most sparse coding based automatic music genre classification systems transform the music signal into frame-level raw features, and then encode the frame-level features into frame-level sparse codes. Since the encoding only considers information in one frame, temporal pooling technique has been included in this kind of system. For instance, the combinations of statistical moments of a multiple frame representation were used for temporal pooling on raw features [18]. Histogram and pyramid based bag-of-segments schemes were also considered for temporal pooling on encoding [19].

In addition to considering temporal pooling on spectral features, amplitude modulations shown on the short-time Fourier transform (STFT) spectrogram, which depict the spectral patterns varying across time, were extracted using a set of 2-D Gabor filters for genre classification [20]. It has been shown that joint spectro-temporal modulations (STMs) on the auditory (AUD) spectrogram are helpful for music signal categorization, hence helpful for music separation [21]. No doubt that STMs carry critical information and are suitable for genre classification. However, does the information conveyed by the STMs provide more benefit than the spectral features? If so, what kind of spectrogram provides the most informative STMs for genre classification? Is it the STFT spectrogram or the hearing-morphic spectrogram such as the constant-Q transform (CQT) spectrogram or the AUD spectrogram? This paper is trying to answer these questions. Here, we built a sparse coding based genre classification system for evaluations.

The rest of this paper is organized as follows. A brief introduction of the tested spectrograms and STM features are presented in Section 2. Section 3 describes the sparse coding and dictionary learning. Section 4 describes the genre classification method and shows evaluation results. Lastly, Section 5 draws the conclusion.

2. FEATURE EXTRACTION

In this section, we introduce the various features used in this paper. Two types of raw features are considered: the frame-level features extracted from STFT, CQT and AUD spectrograms; and their corresponding STM fea-

tures. For the STM features, we apply Gabor filters to STFT spectrogram [20] and rate-scale (RS) filters to the hearing-morphic CQT and AUD spectrograms. Features mentioned in this section are considered as raw features in the dictionary for the sparse coding system.

2.1 Frame-based Features

2.1.1 STFT Spectrogram

The STFT spectrogram is the most conventional time-frequency representation of audio signals. In this paper, we computed 1024-point FFT for each frame and adjacent frames are with 50% overlap. This computation resulted in a 513-dimensional magnitude spectrum which served as a feature vector.

2.1.2 CQT Spectrogram

The constant-Q transform (CQT) produces another kind of time-frequency audio representation with logarithmic frequency scale and different temporal/spectral resolutions at different frequency bands. The CQT spectrogram is considered closely suited to human perception of sound.

In this paper, we set 8 octaves for the frequency range with the frequency resolution of 64 bins per octave, resulting in 512-dimensional feature vectors. For implementation, we used the Constant-Q Transform Toolbox [22][23] which implements the computationally-efficient CQT transform based on FFT [24].

2.1.3 Auditory (AUD) Spectrogram

The AUD spectrogram is produced by the cochlear module of the auditory model [25]. An input sound is first filtered by a bank of 128 overlapping asymmetric bandpass filters which mimic the frequency selectivity of the cochlea. The center frequencies of the cochlear filters are evenly distributed along a logarithmic frequency axis, over 5.3 octaves (180Hz ~ 7246Hz) with the frequency resolution of 24 filters per octave. The output of each filter is fed into a non-linear compression stage, which models the saturation of inner hair cells while transducing the vibrations of the basilar membrane into intracellular potentials. Next, a simple lateral inhibitory network (LIN) is implemented by a first-order differentiator across filters to account for the masking effect between adjacent filters. A half-wave rectifier combined with a lowpass filter serves as an envelope extractor after the LIN. At the end, the cochlear module produces 128-dimensional feature vectors.

The block diagram of the cochlear module is shown in Figure 1. Outputs at different stages can be formulized as follows:

$$y_1(f,t) = s(t) *_t h(t;f) \quad (1)$$

$$y_2(f,t) = g(\partial_t y_1(f,t)) *_t l(t) \quad (2)$$

$$y_3(f,t) = \max(\partial_f y_2(f,t), 0) \quad (3)$$

$$y_4(f,t) = y_3(f,t) *_t \mu(t;\tau) \quad (4)$$

where $s(t)$ is the input audio signal, $h(t;f)$ is the impulse response of the cochlear filter with the center frequency f, $*_t$ depicts convolution in time, $g(.)$ is a sigmoid function, $l(t)$ a lowpass filter, ∂_t, ∂_f are partial derivative along t, f axes, $\mu(t;\tau) = e^{-t/\tau}u(t)$ is the integration window with the time constant τ, and $u(t)$ is the unit step function. Detailed discussions about this module can be assessed in [26].

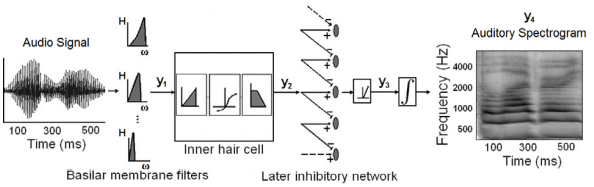

Figure 1. Block diagrams for deriving an AUD spectrogram.

2.2 Modulation Features

2.2.1 Gabor Features

Gabor features are the spectro-temporal "visual features" extracted from a STFT spectrogram as proposed in [20]. To obtain these features, an input audio signal was first transformed into a STFT spectrogram. Then, the STFT spectrogram was divided into 7 sub-spectrograms according to the following 7 subbands: 0Hz ~ 200Hz, 200Hz ~ 400Hz, 400Hz ~ 800Hz, 800Hz ~ 1600Hz, 1600Hz ~ 3200Hz, 3200Hz ~ 8000Hz, and 8000Hz ~ half sampling frequency. Third, each sub-spectrogram was filtered by a set of 42 pre-defined 2-D Gabor filters:

$$\psi(x',y') = \exp\left(-\left(\frac{x'^2+y'^2}{2\sigma^2}\right)\right)\exp\left(\frac{j2\pi x'}{\lambda}\right) \quad (5)$$

$$x' = x\cos(\theta) + y\sin(\theta) \quad (6)$$

$$y' = -x\sin(\theta) + y\cos(\theta) \quad (7)$$

where x and y represent the time and frequency axes of the STFT sub-spectrogram, $\theta \in \{0°, 30°, \ldots, 150°\}$ indicates the orientation of the Gabor filter, $\lambda \in \{2.5, 5, \ldots, 17.5\}$ denotes the thickness of the Gabor filter, and $\sigma = 0.5\lambda$ for the standard deviation of the Gaussian function. This process transformed the STFT spectrogram into 294 modulation sub-spectrograms. The energy contour of each modulation sub-spectrogram was obtained by averaging the modulation sub-spectrogram along the frequency axis. At this stage, the STFT spectrogram was transformed into 294 modulation energy contours in the time domain. Finally, the mean and standard deviation of these contours were concatenated to form the "visual features", referred to as the Gabor features in this paper.

Figure 2 demonstrates the meaning of the Gabor features. The upper panel shows a segment of a sample STFT sub-spectrogram, while the bottom panel shows two energy contours derived from outputs of the two Gabor filters ($\lambda = 7.5,\ \theta = 0°$ and $\lambda = 7.5,\ \theta = 90°$). We can observe that strong responses of the contours result from strong vertical and horizontal patterns in the spectrogram.

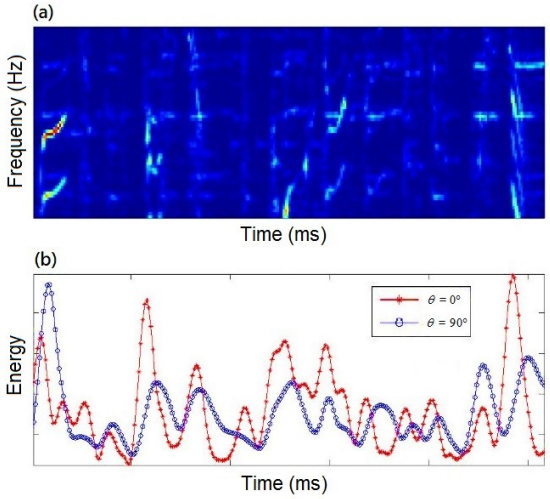

Figure 2. (a) A sample STFT sub-spectrogram. (b) The energy contours derived from outputs of two Gabor filters ($\lambda = 7.5,\ \theta = 0°$ and $\lambda = 7.5,\ \theta = 90°$).

2.2.2 Rate-Scale (RS) Features

RS feature is another kind of modulation feature extracted by the cortical module [25]. The cortical module, which is inspired by neural recordings of the auditory cortex (A1), models the spectro-temporal selectivity of neurons in A1 [26].

Specifically, in the auditory model, the AUD spectrogram is further analyzed by neurons in A1. From functional point of view, the cortical neurons are modeled as a bank of two-dimensional filters with different spectro-temporal selectivity. These two-dimensional filters can be characterized by spectro-temporal modulation parameters, rate and scale. The rate parameter characterizes the velocity of the modulation varying along the temporal axis on the AUD spectrogram and the scale parameter characterizes the density of the modulation distributed along the logarithmic frequency axis on the AUD spectrogram. The filtering process can be formulized as follows:

$$r(f,t,\omega,\Omega) = y_4(f,t) *_{ft} STIR(f,t;\omega,\Omega) \qquad (8)$$

where r denotes the 4-dimensional output of the cortical module, y_4 is the AUD spectrogram, $*_{ft}$ denotes the two-dimensional convolution along temporal and logarithmic frequency axes, $STIR$ is the impulse response of the two-dimensional modulation filter, ω and Ω denote the rate and the scale parameter respectively. Figure 3 demonstrates examples of rate-scale features of three time-frequency (T-F) units in the AUD spectrogram. The top panel shows a sample AUD spectrogram and the bottom three panels show the rate-scale plots, which record the local amplitude resolved by each of the rate-scale modulation filters, of the three T-F units indicated by the arrows. The rate-scale plot reflects local modulation energy distribution and the sweeping direction of the modulation (positive/negative rate representing the downward/upward directivity) of a particular T-F unit in the AUD spectrogram. Detailed explanations about the information encoded by the rate-scale plot can be assessed in [21].

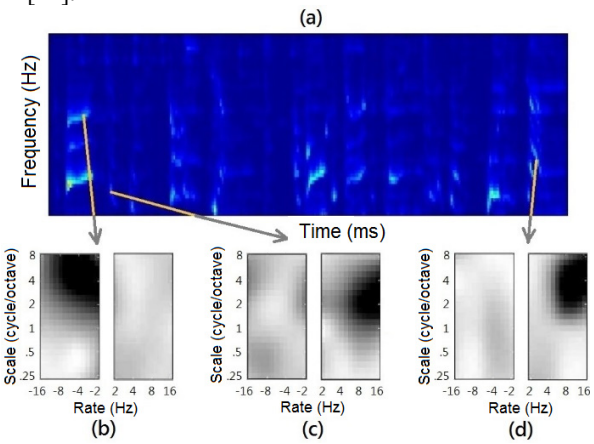

Figure 3. (a) A sample AUD spectrogram. (b)(c)(d) Rate-scale plots of the T-F units indicated by the arrows in (a).

In this paper, the local amplitude of the cortical output r is averaged in each subband and concatenated,

$$\bar{r}_i = \frac{1}{N_i}\sum_{j=1}^{Ni} r(f_j,t,\omega,\Omega) \qquad (9)$$

$$\bar{r} = [\bar{r}_1, \bar{r}_2, \cdots \bar{r}_O] \qquad (10)$$

where $o \in \{1,2,3,\ldots,O\}$ is the index of the subband, Ni is the number of bins in the i-th subband, O is the total number of subbands and $\bar{r}$ is our final RS feature. We extract RS features from the AUD spectrogram and the CQT spectrogram. For the cases of AUD spectrogram, the parameters were selected as $\omega \in \pm\{2,4,8,16\}$, $\Omega \in \{0.25,0.5,1,2,4,8\}$ and O=6(180Hz ~ 200Hz, 200Hz ~ 400Hz, 400Hz ~ 800Hz, 800Hz ~ 1600Hz, 1600Hz ~ 3200Hz, 3200Hz ~ 7246Hz), resulting in 288-dimensional feature vectors. For the cases of CQT spectrogram, the parameter were selected as $\omega \in \pm\{2,4,8\}$, $\Omega \in \{0.25,0.5,1,2,4,8,16\}$ and O=7(0Hz ~ 200Hz, 200Hz ~ 400Hz, 400Hz ~ 800Hz, 800Hz ~ 1600Hz, 1600Hz ~ 3200Hz, 3200Hz ~ 8000Hz, 8000Hz ~ half-sampling frequency), resulting in 294-dimensional feature vectors. These parameters were selected mainly to have comparable feature dimensions with the restriction posed by the 5.3-octave frequency coverage of the AUD spectrogram.

3. SPARSE CODING AND DICTIONARY LEARNING

Generally speaking, the sparse coding technique decomposes the original signal into a combination of a few codewords in a given codebook (or dictionary). The objective function of sparse coding can be formulated as:

$$\alpha^* = \arg\min_{\alpha} \frac{1}{2}\|x - D\alpha\|_2^2 + \lambda\|\alpha\|_1 \qquad (11)$$

where $x \in \mathbb{R}^d$ is the input signal (the feature vector in our case), $\alpha \in \mathbb{R}^k$ is the sparse code of x, $D \in \mathbb{R}^{d \times k}$ is a given dictionary and λ is a parameter which controls the sparsity of α. The d is the dimension of the feature vector and the k is the codebook size. Equation (11) is usually referred to as the Lasso problem and can be solved by the LARS-lasso algorithm [27].

Furthermore, the objective function of dictionary learning can be formulated as:

$$D^* = \arg\min_{D,\alpha} \frac{1}{n} \sum_{i=1}^{n} \left(\frac{1}{2} \|x_i - D\alpha_i\|_2^2 + \lambda \|\alpha_i\|_1 \right) \qquad (12)$$

where n is the total number of data to train the dictionary. Equation (12) represents a joint optimization problem in α and D. In this paper, the online dictionary learning (ODL) algorithm [28][29] was used to train the dictionary and the SPArse Modeling Software (SPAMS) [30] was used for implementation.

4. EXPERIMENTS

In this section, we describe the settings of the experiments and the classification results using various kinds of features.

4.1 Dataset

4.1.1 GTZAN Dataset

This public dataset is frequently used in literature for evaluation of automatic music genre classification. The dataset is composed of 100 30-second music clips in each of the ten genres (blues, classical, country, disco, hip-hop, jazz, metal, pop, reggae, and rock). Each clip is sampled at 22050 Hz.

4.1.2 H-L Dataset

This dataset was collected by ourselves and used in this work for learning the dictionary of music. The dataset is composed of 100 30-second clips in each of the 21 genres (a cappella, A-pop, blues, bossa nova, classical, C-pop, electropop, funk, hip-hop, jazz, J-pop, latin, metal, musical, new age, opera, R&B, reggae, rock, romantical, and soul). All the clips in this dataset are different from those in the GTZAN dataset. Each clip is sampled at 44100 Hz.

4.2 System Overview

In our classification system, an input music clip is first transformed into the frame-level feature vectors. The feature vectors are then normalized to unit l_2-norm vectors. Second, the normalized feature vectors are encoded into frame-level sparse codes. Next, we summarize the frame-level sparse codes over the entire clip to obtain the song-level feature w. Finally, w is power normalized using Equation (13) to train/test the classifier. The block diagram of the classification system is shown in Figure 4.

$$w^* = sign(w)|w|^a \qquad (13)$$

In all of the experiments, we set the codebook size to 1024, the regularization parameter λ to $1/\sqrt{d}$, and the power normalization parameter a in Equation (13) to 0.5. The linear-SVM implemented in LIBSVM [31] was used as the classifier. Evaluation results were obtained by averaging results from 100 ten-fold cross-validation.

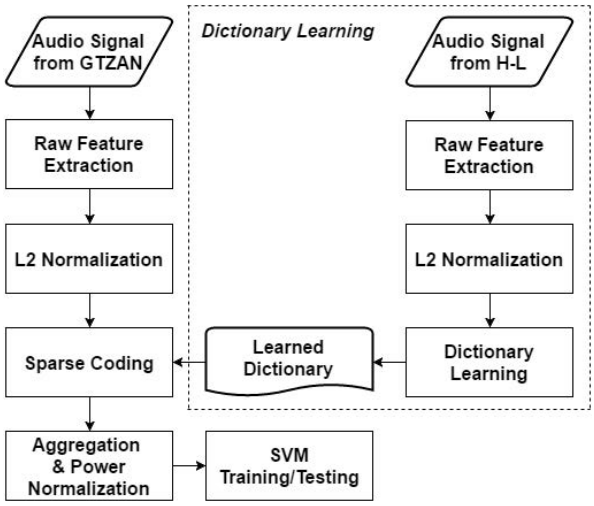

Figure 4. The block diagram of the sparse coding based classification system. The H-L dataset was mainly used to generate the dictionary of music.

4.3 Experiment Results

Experiment results are shown in this section. For simplicity, the name of the classification system is referred to as the name of the used raw features (e.g., the sparse coding based automatic genre classification system using STFT features is referred to as the STFT system).

4.3.1 Spectrogram Features versus Modulation Features

Recognition rates using the spectrogram features and the corresponding modulation features (STFT/Gabor, AUD/RS, CQT/RS) are shown in Figure 5. We can see that corresponding modulation features of the STFT spectrogram have a negative impact to system performance (75.8% to 73.8%) while they provide significant benefit to AUD spectrogram (71.0 to 79.7%) and CQT spectrogram (77.0% to 84.7%).

The main difference among the three spectrograms is the frequency scale, linear scale in STFT but logarithmic scale in AUD and CQT spectrograms. We postulate that modulation features extracted from the logarithmic frequency spectrogram are beneficial to genre classification. For validation, Gabor features were extracted from all three spectrograms and tested for system performance. Figure 6 shows the results of three Gabor systems (STFT/Gabor, AUD/Gabor, CQT/Gabor). Clearly, comparing with spectrogram features, Gabor features demonstrate a positive effect on the genre classification rate when extracted from the AUD and CQT spectrograms but a negative effect when extracted from the STFT spectrogram.

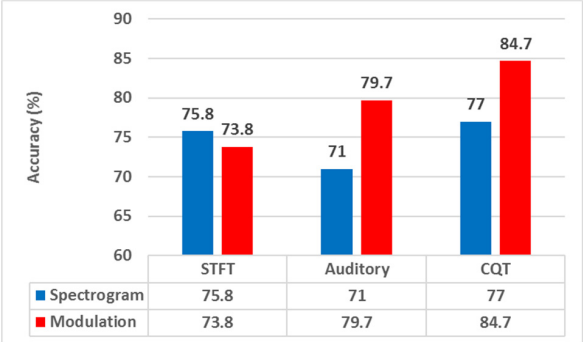

Figure 5. The recognition rates using the spectrogram features and corresponding modulation features (STFT/Gabor, AUD/RS, CQT/RS).

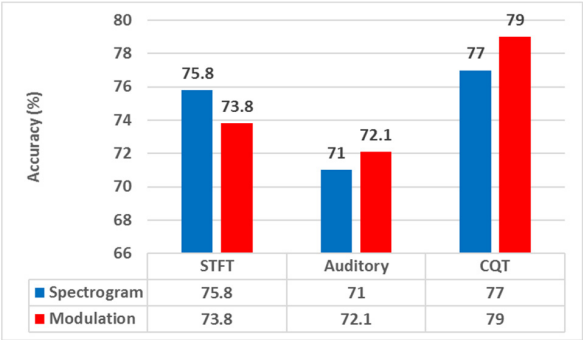

Figure 6. The recognition rates using the spectrogram features and Gabor modulation features (STFT/Gabor, AUD/Gabor, CQT/Gabor).

4.3.2 AUD Spectrogram versus CQT Spectrogram

Figure 5 and 6 show both Gabor and RS modulation features extracted from the CQT spectrogram perform better than those extracted from the AUD spectrogram. The main differences between CQT and AUD spectrograms are the filter shape, the frequency resolution (64 bins per octave on CQT and 24 bins per octave on AUD), and the covered frequency range (40Hz ~ 10700Hz on CQT and 180Hz ~ 7246Hz on AUD).

To investigate the effects from the frequency resolution and the frequency range, we tested RS modulation features extracted from CQT spectrograms with different settings listed in Table 1. The recognition rates are shown in Figure 7. We can observe that higher frequency resolution (64 bins/octave versus 24 bins/octave) does not necessarily produce higher recognition rate. Finding the optimal frequency resolution for recognition rate, however, is beyond the scope of this work. On the other hand, wider frequency coverage is more beneficial to system performance. In Exp.4, the CQT spectrogram was computed using the same frequency resolution and frequency coverage as the AUD spectrogram yet its RS features outperforms the RS features of AUD (82.4% versus 79.7% shown in Figure 5). It is probably because the CQT spectrogram possesses a higher Q value than the AUD spectrogram. A higher Q value generates a more sharpened spectrogram hence producing better performance.

Exp.	frequency range	bins/octave	ω	Ω
1	40Hz ~ 10700Hz	64	2~16	0.25~8
2	40Hz ~ 10700Hz	24	2~16	0.25~8
3	180Hz ~ 7246Hz	64	2~16	0.25~8
4	180Hz ~ 7246Hz	24	2~16	0.25~8

Table 1. Different frequency settings for generating CQT/RS modulation features

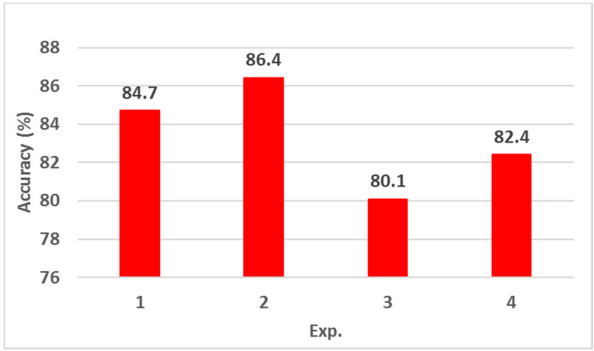

Figure 7. The recognition rates using CQT/RS features with different frequency range and frequency resolutions as listed in Table 1.

4.3.3 Gabor Features versus RS Features

From Figure 5 and 6, we can observe that the RS feature set performs better than the Gabor feature set on both of CQT and AUD spectrograms. RS features and Gabor features are produced using different sets of 2-D modulation filters. The parameters of the Gabor filter, θ and λ, and the parameters of the rate-scale filter, ω and Ω, affect the shape of the 2-D filter by changing its center frequency and bandwidth.

To demonstrate the effect of using different modulation filters, we tested RS features extracted from CQT spectrogram using a different set of ω and Ω. Experiment setting is listed as Exp.2 in Table 2 and the results are shown in Figure 8, where Exp.1 and Exp.3 are the RS/Gabor features using the original set of 2-D filters, respectively. We can observe that selecting different 2-D modulation filters significantly affect system performance. Therefore, selecting an appropriate set of 2-D filters for modulation feature extraction is important to system performance.

Exp.	2-D filter	ω	Ω	θ	λ
1	RS	2~8	0.25~16		
2	RS	0.25~16	2~8		
3	Gabor			0°, 30°, ..., 150°	2.5, 5, ..., 17.5

Table 2. Different modulation filters used to extract modulation features from the CQT spectrogram

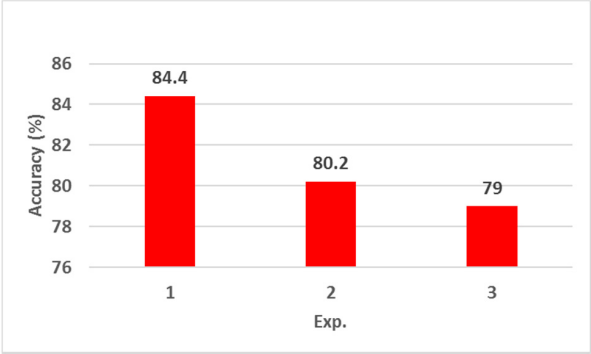

Figure 8. The recognition rates using different sets of 2-D modulation filters on the CQT spectrogram. Parameters of the modulation filters are listed in Table 2.

4.3.4 Linear Magnitude versus Log. Magnitude

It has been shown that logarithmic magnitude STFT produces better features than the linear magnitude STFT for genre classification [12]. In this sub-section, we demonstrate the effect of using log. magnitude on spectrograms. In addition to using spectrogram features, the system performance using modulation features extracted from the log. magnitude spectrograms were also examined.

Experiment results of using log. magnitude spectrograms versus using linear magnitude spectrograms are shown in Figure 9. Results of using their corresponding modulation features (Gabor from STFT, RS from AUD, and RS from CQT) are shown in Figure 10. We can see that both spectrogram features and modulation features extracted from log. magnitude spectrograms perform better than those extracted from linear magnitude spectrograms.

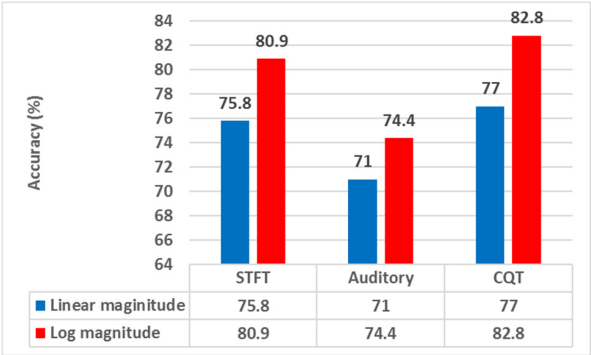

Figure 9. Recognition rates of using spectral profiles extracted from log. magnitude spectrograms and from linear magnitude spectrograms.

5. CONCLUSIONS

In this paper, we investigate the efficacy of features derived from joint spectro-temporal modulations, which intrinsically convey timbre and rhythm information of the sound, on music genre classification using a sparse coding based classification system. We extract two kinds of STM features, the Gabor and RS features, from three kinds of spectrograms, STFT, auditory, and CQT spectrograms, of the music signal and conduct several comparative experiments. The results show that modulation features do outperform spectral profiles in genre classification. In addition, several conclusions can be drawn from our results: 1) modulation features extracted from the logarithmic frequency scaled spectrogram perform better than those extracted from the linear frequency scaled spectrogram; 2) the spectrogram with wider frequency coverage produces more effective modulation features; 3) the selection of modulation filters could be task-dependent; 4) modulation features extracted from log. magnitude spectrograms produce higher genre recognition rates than those extracted from linear magnitude spectrograms.

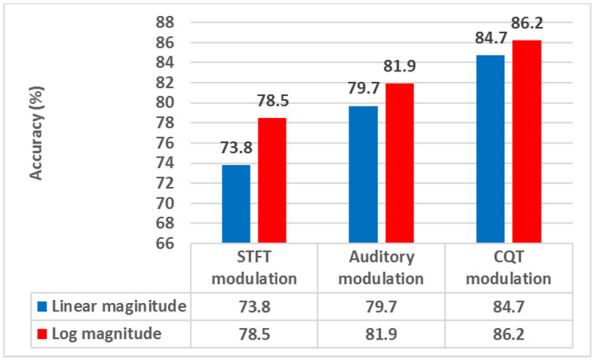

Figure 10. Recognition rates of using modulation features extracted from log. magnitude spectrograms and from linear magnitude spectrograms.

In this paper, the highest genre recognition rate on GTZAN dataset using modulation features is 86.2%, which is obtained by using RS features extracted from the log. magnitude CQT spectrogram. From experiment results shown in Section 4, we can assume the performance can probably be better by fine-tuning the parameters of the classification system, including the rate, scale parameters of the modulation filters and the frequency resolution of the CQT spectrogram.

6. ACKNOWLEDGEMENTS

This research is supported by the National Science Council, Taiwan under Grant No NSC 102-2220-E-009-049.

7. REFERENCES

[1] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Trans. Speech and Audio Processing*, vol. 10, no. 5, pp. 293–302, 2002.

[2] C. Xu, N. C. Maddage, X. Shao, F. Cao, and Q. Tian, "Musical genre classification using support vector machines," *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process.*, no. 5, pp. 429–432, 2003.

[3] A. Meng, P. Ahrendt, and J. Larsen, "Improving music genre classification by short-time feature intergration," *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process.*, no. 5, pp. 497–500, 2005.

[4] A. Meng and J. Shawe-Taylor, "An investigation of feature models for music genre classification using the support vector classifier," *Proc. of the Int. Soc. for Music Inform. Retrieval Conf.*, pp.604–609, 2005.

[5] A. Meng, P. Ahrendt, J. Larsen, and L. K. Hansen, "Temporal feature integration for music genre classification," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 15, no. 5, pp. 1654–1664, 2007.

[6] J. Bergstra, N. Casagrande, D. Erhan, D. Eck, and B. K´egl, "Aggregate features and adaboost for music classification," *Machine Learning*, vol. 65, no. 2-3, pp. 473–484, 2006.

[7] Y. Panagakis, C. Kotropoulos, and G. R. Arce, "Music genre classification using locality preserving non-negative tensor factorization and sparse representations," *Proc. of the Int. Soc. for Music Inform. Retrieval Conf.*, pp. 249–254, 2009.

[8] Y. Panagakis and C. Kotropoulos, "Music genre classification via topology preserving non-negative tensor factorization and sparse representations," *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process.*, pp. 249–252, 2010.

[9] H. Rump, S. Miyabe, E. Tsunoo, N. Ono, and S. Sagama, "Autoregressive mfcc models for genre classification improved by harmonic-percussion separation," *Proc. of the Int. Soc. for Music Inform. Retrieval Conf.*, pp. 87–92, 2010.

[10] N. Ono, K. Miyamoto, H. Kameoka, and S. Sagayama, "A real-time equalizer of harmonic and percussive componets in music signals," *Proc. of the Int. Soc. for Music Inform. Retrieval Conf.*, pp. 139–144, 2008.

[11] Cao, Chuan, and Ming Li. "Thinkit's submissions for MIREX2009 audio music classification and similarity tasks." *Music Information Retrieval Evaluation eXchange (MIREX)* (2009).

[12] C.-C. M. Yeh and Y.-H. Yang, "Supervised dictionary learning for music genre classification," *Proc. ACM Int. Conf. on Multimedia Retrieval*, no. 55, 2012.

[13] A. Ruppin and H. Yeshurun, "Midi genre classification by invariant features," *Proc. of the Int. Soc. for Music Inform. Retrieval Conf.*, pp. 397–399, 2006.

[14] T. Lidy, A. Rauber, A. Pertusa, and J. M. Inesta, "Improving genre classification by combination of audio and symbolic descriptors using a transcription system," *Proc. of the Int. Soc. for Music Inform. Retrieval Conf.*, pp. 61–66, 2007.

[15] R. Mayer, R. Neumayer, and A. Rauber, "Rhyme and style features for musical genre categorization by song lyrics," *Proc. of the Int. Soc. for Music Inform. Retrieval Conf.*, pp. 337–342, 2008.

[16] C. McKay, J. A. Burgoyne, J. Hockman, J. B. L. Smith, G. Vigliensoni, and I. Fujinaga, "Evaluating the genre classification performance of lyrical features relative to audio, symbolic and cultural features," *Proc. of the Int. Soc. for Music Inform. Retrieval Conf.*, pp. 213–218, 2010.

[17] R. Mayer and A. Rauber, "Music genre classification by ensembles of audio and lyrics features," *Proc. of the Int. Soc. for Music Inform. Retrieval Conf.*, pp. 675–680, 2011.

[18] C.-C. M. Yeh and Y.-H. Yang, "Towards a more efficient sparse coding based audio-word feature extraction system," *Proc. Asia-Pacific Signal and Information Processing Association Annual Summit and Conf.*, pp. 1–7, 2013.

[19] C.-C. M. Yeh, L. Su, and Y.-H. Yang, "Dual-layer bag-of-frames model for music genre classification," *Proc. IEEE Int. Conf. on Acoust, Speech and Signal Process*, pp. 246–250, 2013.

[20] M.-J. Wu, Z.-S. Chen, and J.-S. R. Jang, "Combining visual and acoustic features for music genre classification," *Proc. IEEE Int. Conf. on Machine Learning and Applications and Workshops*, pp. 124–129, 2011.

[21] Yen, Frederick, Yin-Jyun Luo, and Tai-Shih Chi. "Singing Voice Separation Using Spectro-Temporal Modulation Features." *Proc. of the Int. Soc. for Music Inform. Retrieval Conf.*, pp. 617–622, 2014.

[22] [Online]https://code.soundsoftware.ac.uk/projects/constant-q-toolbox.

[23] Schörkhuber, Christian, and Anssi Klapuri. "Constant-Q transform toolbox for music processing." *Proc. of Sound and Music Computing Conference,* pp. 3-64, 2010.

[24] J. C. Brown and M. S. Puckette, "An efficient algorithm for the calculation of a constant q transform," *The Journal of the Acoustical Society of America*, vol. 92, no. 5, pp. 2698–2701, 1992.

[25] [Online]http://www.isr.umd.edu/Labs/NSL/Downloads.html

[26] T. Chi, P. Ru, and S. A. Shamma, "Multiresolution spectrotemporal analysis of complex sounds," *The Journal of the Acoustical Society of America*, vol. 118, no. 2, pp. 887–906, 2005.

[27] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *The Annals of Statistics*, vol. 32, no. 2, pp. 407–499, 2004.

[28] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," *Proc. of Int. Conf. on Machine Learning*, pp. 689–696, 2009.

[29] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *The Journal of Machine Learning Research*, pp. 19–60, 2010.

[30] [Online]http://spams-devel.gforge.inria.fr/downloads.html

[31] Chang, Chih-Chung, and Chih-Jen Lin. "LIBSVM: a library for support vector machines." *ACM Transactions on Intelligent Systems and Technology,* vol. 2, no.3, Article 27, 2011.