IEICE TRANS. INF. & SYST., VOL.E94–D, NO.6 JUNE 2011

LETTER

An Informative Feature Selection Method for Music Genre Classification

Jin Soo SEO[†a)], ***Member***

SUMMARY This letter presents a new automatic musical genre classification method based on an informative song-level representation, in which the mutual information between the feature and the genre label is maximized. By efficiently combining distance-based indexing with informative features, the proposed method represents a song as one vector instead of complex statistical models. Experiments on an audio genre DB show that the proposed method can achieve the classification accuracy comparable or superior to the state-of-the-art results.

key words: *musical genre classification, feature selection, mutual information*

1. Introduction

Music information retrieval (MIR) is becoming widespread due to commercial demands for online music searching, streaming, and downloading services. For a successful MIR system, we need to have various metadata of music content, such as genre, tempo, chord, instrumentation, style, mood, singer, and composer, which could be extracted either manually or automatically. This paper focuses on one of the issues, automatic musical genre classification. To avoid the time-consuming and tedious manual annotation, it is necessary to automatically classify the musical genre of a given audio signal.

Most of the musical genre classification systems employ low-level spectral features, such as mel-frequency cepstral coefficients (MFCC) or other spectrum descriptors [1]–[5], which describe the timbral texture of an audio signal. The low-level spectral features are converted into the intermediate representation, which is used for training and testing the statistical classifiers, such as support vector machines (SVMs). There are typically two types of the intermediate representation: song-level and segment-level representation. The song-level representation in [3]–[5] models each song with Gaussian Mixture Model (GMM) of the low-level spectral features. The distance between the song-level representations is estimated by either KL divergence [3], [4] or earth-mover distance (EMD) [5]. The segment-level representation [2] models each segment of an audio (typically between 1 and 6 seconds) with various statistical measures, such as mean, variance, and correlation. The genre classification is performed at every segment, and either the majority or the weighted voting rule is used in combining the classification result of each segment.

In this letter, we present a study on an informative song-level representation for musical genre classification. In most of the previous approaches which deal with the song-level representations, the distance between the statistical models, such as GMMs, of the two audio signals is used as a metric for music classification [1], [4], [5]. Despite their excellent performance, the previous methods mentioned above have several shortcomings. First of all, the construction of the song-level representations is based on an iterative process, which may not converge in some cases. Second, the pairwise distance using KL or the EMD is computationally expensive and does not have a closed-form solution in most of the cases. Moreover, the distance computation in the previous approaches is rather redundant in that uses all the components of GMM (or all the clusters in K-means) which may not be relevant for determining genres. To mitigate those problems, we propose a novel song-level feature modeling method in which all the frame-level spectral features are first converted into indexes, and a simple statistics (such as mean or variance) of the indexes is used as a song-level feature of an audio signal. To obtain genre-specific indexes, an informative feature selection method based on mutual information is applied. Experimental results show that the proposed song-level representation is promising for the musical genre classification.

2. Proposed Musical Genre Classification Method Based on an Informative Song-Level Representation

The functional diagram of the proposed song-level representation is shown in Fig. 1. Basically the proposed method is based on the distance-based indexing in which the distances from a few selected codewords are used to index the data. The average distances, between the low-level spectral features and the selected codewords, are used as the song-level representation of an audio signal. With the appropriately chosen codewords, the simple statistics (average in our case) can be used instead of a rather complex statistical model, such as GMM. The obtained song-level representation is used for constructing the genre classifier as shown in Fig. 2. The genre information of the training songs is used in both selecting the informative codewords and training a statistical classifier over the song-level representation. Details of the song-level representation and the codeword selection are given in the Sects. 2.1 and 2.2 respectively.

Manuscript received March 12, 2010.
Manuscript revised January 20, 2011.

[†]The author is with the Department of Electrical Engineering, Gangneung-Wonju National University, Gangneung, Rep. of Korea.
a) E-mail: jsseo@gwnu.ac.kr

DOI: 10.1587/transinf.E94.D.1362

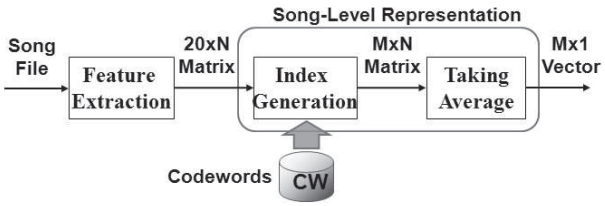

Fig. 1 Functional diagram of the proposed song-level representation.

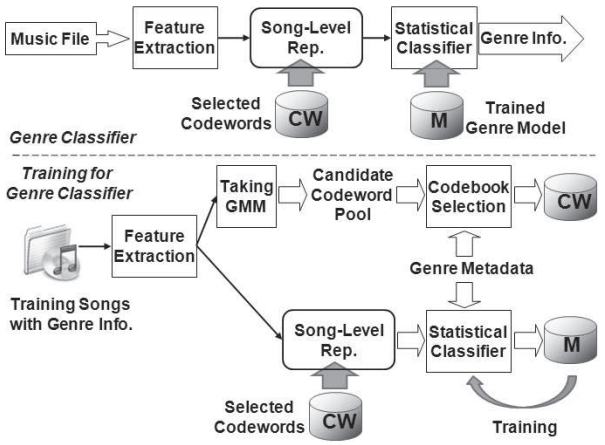

Fig. 2 Construction of the genre classifier based on the proposed song-level representation.

2.1 Proposed Song-Level Representation

As shown in Fig. 1, we first extract the low-level spectral features from an input audio. An audio signal is split into overlapping segments (called frames) of length L with 50% overlap (in our system, L is 46.4 ms). Each frame is windowed by a Hamming window of length L and transformed into the frequency domain. From each frame, we extract the low-level spectral features. We consider the 20-order MFCC as the low-level spectral feature as in [1].

The distance-based index $F[m, n]$ is defined by normalizing the distance D between the n-th frame feature vector a_n and the m-th codeword s_m as follows:

$$F[m, n] = \exp(-\beta D(a_n, s_m)) \tag{1}$$

where β is a normalization constant. We use the square of the Mahalonobis distance as a distance metric D,

$$D(a_n, s_m) = (a_n - s_m)^T \Sigma_m^{-1} (a_n - s_m) \tag{2}$$

where Σ_m is the covariance matrix associated with the codeword s_m. We limit the form of Σ_m as a diagonal matrix to reduce computations. The selection of codewords s_m and Σ_m will be disclosed in Sect. 2.2. For now, we assume that we have M codewords which have discriminative power for the given classification problem. Finally we obtain the M-order song-level feature vector g by taking the average of the indexes of all frames in a song as follows:

$$g[m] = \frac{1}{N} \sum_{n=1}^{N} F[m, n] \quad \text{for} \quad m = 1, 2, \cdots, M \tag{3}$$

where N is the number of the frames in a song. The feature vector g represents average distances between the low-level spectral features and the codewords. By first converting the low-level feature into M by N index matrix F using genre-discriminant codewords, we can obtain song-level feature vector g of an audio signal based on a simple statistical model over index without resorting to an iterative modeling (such as expectation-maximization algorithm). The proposed method is also conducive in constructing musical genre classifier since the obtained representation is given as a vector of the normalized distances which can be used with any type of classifiers.

2.2 Codebook Selection

We consider each Gaussian component of GMMs from the training data as a candidate for codeword. We denote the initial set of the candidate codewords (Gaussian distributions) as $U_0 = \{u_1, u_2, \ldots, u_K\}$. The goal of our codebook selection is to find the most informative and less redundant subset of U_0. Since considering all the candidates in one time is a formidable task, we use a greedy-search algorithm in [6] that adds a codeword iteratively to the set of the already chosen codewords. The probability distribution of the song-level feature g_k in (3) associated with the codeword u_k is estimated in advance from a training data. The search is initialized by selecting the first codeword s_1 which maximizes the mutual information $I(u_k; C)$ between the candidate u_k and the genre label C given by

$$I(u_k; C) = \sum_{y \in C} \int_0^1 P_{g_k,C}(x, y) \log \frac{P_{g_k,C}(x, y)}{P_{g_k}(x) P_C(y)} dx \tag{4}$$

where in practice we use a histogram for representing the probability distributions. At the second stage, the selection criterion is not the mutual information alone, but how much information s_2 can add with respect to the already existing s_1 [6]. Thus the selection criterion is to find the codeword that incurs the highest information gain,

$$s_k = \arg \max_{u_i \in U_m} \min_{u_j \in S_m} (I(u_i, u_j; C) - I(u_j; C)). \tag{5}$$

The updates of the selected codewords S_m and the candidate pool U_m are given by [6]:

$$S_{m+1} = S_m \cup \{s_k\} \quad \text{and} \quad U_{m+1} = U_m \setminus \{s_k\} \tag{6}$$

where the operation $\cup$ and the $\setminus$ refer to the set union and the set difference (or relative complement) respectively. The greedy-selection process in (5) and (6) repeats until we gather a sufficient number of selected codewords. The number of codewords M can be adjusted for a given computational capacity. The informative codebook S_M selected in this way is used in the index generation in (1).

2.3 Complexity Consideration

For a computationally-efficient classification system, we

have to reduce two kinds of computations involved in constructing features and classifying them. The previous song-level representations [1], [4], [5] are obtained by modeling the feature vectors of an audio signal using GMM or K-means clusters. In most of the cases, the number of GMM components is between 20 and 50. In modeling the feature vectors into GMM, the Mahalonobis distance between the feature vectors and the GMM components should be calculated in each step of iterations. In classifying an audio signal, the previous song-level representations (GMM or K-means clusters) are compared each other with the KL or EMD [5], which is computationally expensive and does not have a closed-form solution. In these regards, the proposed method has two major advantages over the previous ones in terms of computations. First of all, although the proposed method relies on the Mahalonobis distance as in (2), it is not an iterative process. The distance between the feature vectors of a song and the selected codewords needs to be computed only once. For a 2-minute song on a computer with 2.4 GHz Pentium Processor, it takes 0.39 seconds in computing the proposed song-level representation with $M = 128$ while it takes 5.1 seconds on average (with the standard deviation of 1.61 seconds) in computing 20-component GMM with a diagonal covariance on the same setting. Second, the proposed representation is given by an M-order vector, to which any kinds of classifiers are readily applicable. In practice, a simple linear classifier, that is computationally efficient, can provide enough performance since the codeword selection in Sect. 2.2 preserves genre-specific information while reducing the redundancy and the higher-order interactions among the song-level vector components [6]. Due to the codeword selection, the proposed method requires more computations during the training than the previous ones [1], [4] while it needs less computations during the testing, which is favorable in practice.

3. Experimental Results

The genre-classification accuracy of the proposed method was evaluated on the magnatune genre dataset used for ISMIR 2004. The dataset is composed of the six different types of genres: classical, electronic, jazz_blues, metal_punk, rock_pop, and world. In total there are 1458 songs in the dataset where the number of songs in each genre is not equal. One half of the songs is used for training, and the other half is used for testing. Each song in the dataset is converted to mono at a sampling frequency of 22050 Hz and then divided into frames of 46.4 ms overlapped by 23.2 ms. We computed the 20-order MFCC of each frame as a low-level feature.

The codebook for index generation was selected from the GMMs of the training songs as in Sect. 2.2 with the number of GMM components varying from 5 to 20. Then the song-level representation of each training song was calculated as the average index in (3) with the selected codebook. The linear SVM classifier was constructed from the song-level representations of training songs. The classification accuracy of the constructed classifier on the testing songs is shown in Fig. 3. With a larger number of GMM components in training, the number of candidate codewords K increases while the covariance matrices associated with them generally diminish. In practice, the covariance matrix associated with each codeword represents the effective coverage of it over the feature space from (2). In case that only small value of M is allowed from the computational budget, it is better to choose a small number of GMM components in training (i.e. small value of K and large effective coverage of each codeword). For example, when the value of M is 50, the classification accuracy is much higher with the smaller number of GMM components. In general, as the number of GMM components in training gets larger (i.e. the effective coverage of each codeword is getting smaller), the classification accuracy is expected to rise more gradually with increasing M. However, with a sufficiently larger value of M, the effect of the number of GMM components in training is getting less noticeable in Fig. 3 since the selected codewords have already covered almost whole feature space. In all three cases of the number of GMM components, the best results exceeded 81.5%. On the same dataset, the reported classification accuracy of the works in [7], [8] ranges from 80.95% to 83.5% although they are based on more complicated features, such as higher-order tensors and nonnegative matrix factorization. The reported accuracy of the previous GMM-based method using MFCC was 79% on the same dataset [1]. Even with $M = 50$, the proposed method can achieve 78.2% accuracy. By increasing M, the best result was 84.1%. The results demonstrate that the proposed method with an informative discriminant codebook can reach similar or better performance using a simpler song-level model (actually M-order vector) and type of classifier. We note that computing GMM is needed in training only for codeword selection as shown in Fig. 2.

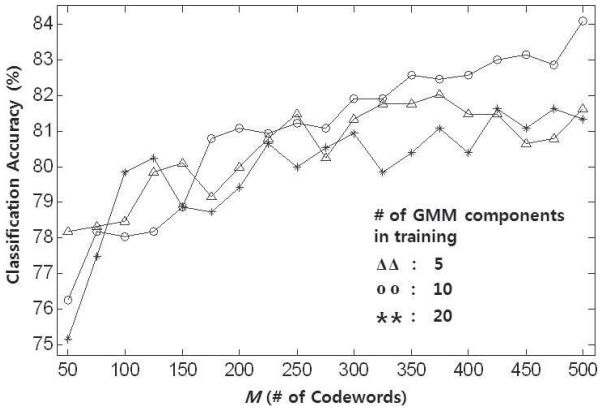

Fig. 3 Classification accuracies versus the number of codewords M.

Table 1 is the confusion matrix of the proposed method with the best classification accuracy (84.1%). The results in Table 1 show that the proposed method generally behave well for most of the genres. The world genre was the most difficult to classify due to its large intravariance of music style [8]. Most of the other misclassifications oc-

Table 1 Confusion matrix of the classification result. (The last row is the classification accuracy of each genre.)

	cl	el	j_b	m_p	r_p	wo
cl	313	0	0	1	2	16
el	3	85	1	1	7	13
j_b	0	4	21	0	0	1
m_p	0	0	0	33	5	1
r_p	0	9	1	10	77	7
wo	4	16	3	0	11	84
	97.8%	74.6%	80.8%	73.3%	75.5%	68.9%

cur among the musically-close genres: electronic, rock_pop, and metal_punk.

4. Conclusion

For musical genre classification, we propose an informative song-level representation using the distance-based indexing incorporated with the codewords selected by a mutual-information maximization criterion. The seamless combination of the proposed distance-based indexing with the feature selection leads to the simplified song-level representations. Experimental results show that the proposed simplified representation can match the classification accuracy of the more complex ones.

Acknowledgments

This work was supported by the research grant from Gangneung-Wonju National University, 2009.

References

[1] E. Pampalk, A. Flexer, and G. Widmer, "Improvements of audio-based music similarity and genre classification," Proc. Int. Conf. on Music Info. Retrieval (ISMIR-05), Sept. 2005.

[2] A. Meng, P. Ahrendt, J. Larsen, and L.K. Hansen, "Temporal feature integration for music genre classification," IEEE Trans. Audio Speech Language Process., vol.15, no.5, pp.1654–1663, July 2007.

[3] J.-J. Aucouturier and F. Pachet, "Improving timbre similarity: How high's the sky?," J. Negative Results in Speech and Audio Sciences, vol.1, no.1, 2004.

[4] M. Mandel and D. Ellis, "Song-level features and support vector machines for music classification," Proc. Int. Conf. on Music Info. Retrieval (ISMIR-05), Sept. 2005.

[5] B. Logan and A. Salomon, "A music similarity function based on signal analysis," Proc. Int. Conf. on Multimedia and Expo (ICME-01), 2001.

[6] M. Vidal-Naquet and S. Ullman, "Object recognition with informative features and linear classification," Proc. Int. Conf. on Computer Vision (ICCV-03), pp.281–288, Oct. 2003.

[7] I. Panagakis, E. Benetos, and C. Kotropoulos, "Music genre classification: A multilinear approach," Proc. Int. Conf. on Music Info. Retrieval (ISMIR-08), Sept. 2008.

[8] A. Holzapfel and Y. Stylianou, "Musical genre classification using nonnegative matrix factorization-based features," IEEE Trans. Audio Speech Language Process., vol.16, no.2, pp.424–434, Feb. 2008.