Random Forest and PCA for Self-Organizing Maps based Automatic Music Genre Discrimination

Xin Jin, Rongfang Bie*

***Abstract*—Digital music distribution industry has seen a tremendous growth in resent years. Tasks such us automatic music genre discrimination address new and exciting research challenges. Automatic music genre recognition involves issues like feature extraction and development of classifiers using the obtained features. As for feature extraction, we base on Self-Organizing Maps to map the high-dimensional audio signals into SOM features. In addition we use Principle Components Analysis (PCA) to reduce feature dimension to improve the classification performance. Regarding the task of genre modeling, we introduce a new method Random Forest. Experiment results show that SOM feature is feasible for music genre classification. Comparisons with traditional classification methods show that the new introduced method can achieve the highest recognition rate. In addition we find that PCA can further improve the discrimination performance of almost all the classifiers we investigate.**

I. Introduction

Musical genres are categorical descriptions that are used to characterize music in music stores, radio stations and now on the Internet. Traditionally, music search and retrieval is carried out mostly in a textual manner, based on author, title or genre. This method leads to a great deal of difficulties for service providers. Real-world music databases from sites like AllMusicGuide or CDNOW grow larger and larger on a daily basis, which requires a tremendous amount of manual work for keeping them updated. Thus, simplifying the task of music database organization would be an important advance. This calls for automatic music classification systems [10].

Automatic genre discrimination of musical audio signals is often performed on spectral features that have been averaged over a large number of audio frames. Several classification strategies have been employed, including RBF Network [1], Multilayer Perceptron [10, 11], Support Vector Machines [9], k-Nearest Neighbors [8], etc. In this paper we introduce a new method Random Forest for music genre classification. We investigate the use of Self-Organizing Maps (SOM) for extracting features from music signals for classification. In addition, we present the use of Principle Components Analysis for reducing feature dimension to improve the classification performance.

The remainder of this paper is organized as follows: Section 2 describes the SOM and PCA based features for music. Section 3 introduces the new method Random Forest and three traditional classifiers, RBF Network, Support Vector Machines and k-Nearest Neighbors. Section 4 presents the dataset and the experiment results. Conclusions are presented in Section 5.

II. Music Features

Sound information in any music file (such as WAV, MP3, MIDI, etc.) is basically stored in the form of pulse code modulation (PCM) using a very high sampling rate of 44.1 KHz. Thus the sound signal is represented by 44,100 integer numbers per second and the resulting feature vector for a several-minute music piece will be greatly high dimensional. We first base on Self-Organizing Maps (SOM) to map the high dimensional musical signals into SOM features and then use Principle Components Analysis to further reduce the feature dimensions.

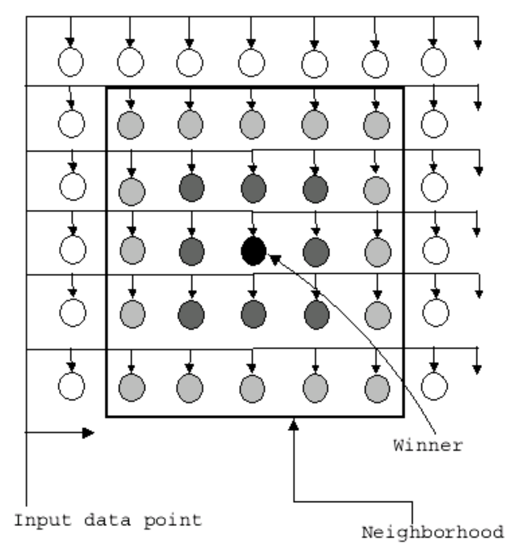

Fig. 1. Self-Organizing Map learning scheme [6].

A. Self-Organizing Map

The Self-Organizing Map provides a mapping from a high-dimensional input space to a usually two-dimensional output space while preserving topological relations as faithfully as possible [3, 4]. SOM based feature was originally proposed for content-based music clustering by Malheiro et al [4]. But whether it is useful for music genre

This work was supported by the National Science Foundation of China under the Grant No. 60273015 and No. 10001006.

Xin Jin is with the College of Information Science and Technology, Beijing Normal University, Beijing 100875, People's Republic of China (Email: xinjin796@126.com).

*Corresponding Author. Rongfang Bie is with the College of Information Science and Technology, Beijing Normal University, Beijing 100875, People's Republic of China (Email: rfbie@bnu.edu.cn).

discrimination is unkown. Fig. 1 shows the learning scheme of SOM. After data preprocessing, input signals $\mathbf{x} \in \mathrm{R}^n$ are presented to the map consisting of a grid of units with n-dimensional weight vectors. An activation function based on certain distance metric is used to determine the winning unit—winner (colored black in Fig. 1). The weight vector of the winner as well as the weight vectors of the neighboring units (defined in Fig. 1 to be of radius 2, shown as a square of size 5, with the winner at the center) are adjusted so that they responds even better for a similar input data point in the future. The winner unit is adjusted the most towards the input data point, while other units in the neighborhood are adjusted with the adjustment factor getting lower and lower as the distance from the winner increase [6]. As results, after the training process, similar input patterns are mapped onto neighboring units of the SOM.

A feature vector representation can be created for each piece of music using the location of its segments as descriptive attributes. More specifically, given an $x \times y$ SOM one can create an $x \cdot y$ dimensional weight vector. Each vector attribute represents the number of segments of a particular piece of music mapped onto the respective unit in the SOM. For instance, given an 2×2 SOM and a piece of music that has 5 segments mapped onto unit (0/0) in the upper right corner of the map, 3 segments onto unit (1/0) in the upper left corner of the map, 8 segments onto unit (0/1) in the lower right corner of the map, and 19 segments onto unit (1/1) in the lower left corner of the map, the four attributes of the song's feature vector are basically set to the according values $(5/3/8/19)^T$, with subsequent normalizing to unit length to make up for length differences of songs.

Fig. 2 shows SOM features for 27 pieces of music, which fall to two classes, Blues (1) and Jazz (2). The SOM used is of the size 18×18, so there is a 324-dimensional vector for each piece of music.

B. Principal Component Analysis

Principal Component Analysis (PCA) is a multivariate data analysis method that is useful in linear feature extraction [14]. The PCA finds a linear transformation $\mathbf{y}=\mathbf{W}\mathbf{x}$ such that the retained variance is maximized. It can be also viewed as a linear transformation that minimizes the reconstruction error [5]. Each row vector of $\mathbf{W}$ corresponds to the normalized orthogonal eigenvector of the data covariance matrix.

One simple approach to PCA is to use singular value decomposition (SVD). Let us denote the data covariance matrix by $\mathbf{R}_{\mathbf{x}}(0) = E\{\mathbf{x}(t)\mathbf{x}^{\mathrm{T}}(t)\}$. Then the SVD of $\mathbf{R}_{\mathbf{x}}(0)$ gives $\mathbf{R}_{\mathbf{x}}(0) = \mathbf{U}\mathbf{D}\mathbf{U}^{\mathrm{T}}$, where $\mathbf{U}= [\mathbf{U}_s,\mathbf{U}_n]$ is the eigenvector matrix (i.e. modal matrix) and $\mathbf{D}$ is the diagonal matrix whose diagonal elements correspond to the eigenvalues of $\mathbf{R}_{\mathbf{x}}(0)$ (in descending order). Then the PCA transformation from m-dimensional data to n-dimensional subspace is given by choosing the first n column vectors, i.e., n principal component vector $\mathbf{y}$ is given by $\mathbf{y}=\mathbf{U}_s^{\mathrm{T}}\mathbf{x}$.

We use PCA to reduce the large SOM feature space to lower than 30-dimension with little information losing.

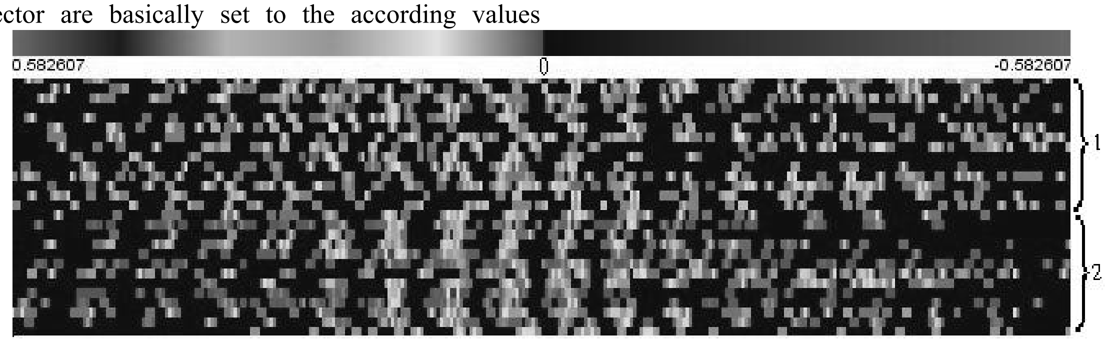

Fig. 2. SOM features of 27 pieces of music which fall into two genres: Blues (1) and Jazz (2). Each row is the feature vector of a piece of music. Colored version of this figure is available under request.

III. CLASSIFICATION SCHEMES

This section we first introduce the new classification algorithm Random Forest for music genre discrimination, then we describe three traditional classifiers for comparison: RBF Network, Support Vector Machines and K Nearest Neighbors, which have been successfully used for music pattern recognition.

A. Random Forest

Random Forest is a relatively new algorithm for classification developed by Leo Breiman [19] that uses an ensemble of classification trees [20, 24]. Each classification tree is built using a bootstrap sample of the data, and at each split the candidate set of variables is a random subset of the variables. Thus, random forest uses both bagging, a successful approach for combining unstable learners [22], and random variable selection for tree building. Each tree is unpruned, so as to obtain low-bias trees. The algorithm yields an ensemble that can achieve both low bias and low variance [27].

More specifically, a forest is grown by using *ntree* (the number of trees in the forest, is chosen appropriately to achieve low error rate of convergence) bootstrapped samples each of size n randomly drawn from the original data of n points with replacement. This first type of randomization helps in building an ensemble of trees and in reducing dependence among the trees. About two-thirds of the data set are used to grow a classification tree. These data are used to obtain unbiased estimates of correct classification rates and variable importance. The second type of randomness is used during building classification trees. For each node of a tree, Random Forest randomly selects *mtry* variables and uses only them to determine the best possible split using certain splitting criterion. This algorithm is fairly robust to the choice of the number *mtry*, the value of which is usually taken to be the square root of the total number of variables. Random forest trees are built without pruning. Predictions for test samples are carried out either by the majority vote of classification trees in the forest or are based on a threshold value selected by the user [25].

Random forest has excellent performance in pattern recognition tasks [18, 21, 23, 25, 26, 27]. Although random forest has not been used in the music literature, it has several characteristics that make it ideal for music genre classification:

a. Can be used when there are many more variables than samples.

b. Has good predictive performance even when most predictive variables are noise.

c. Incorporates interactions among predictor variables.

d. The output is invariant to monotone transformations of the predictors.

Given these promising features, it is important to understand the performance of random forest compared to alternative state-of-the-art learning methods with music genre.

B. Other classifiers

RBF Network: The Radial Basis Function (RBF) Network has shown a great promise in this sort of problems because of its faster learning capacity [16]. A traditional RBF network takes Gaussian functions as its basis functions and adopts the least-squares criterion as the objective function. The input to RBF Network is a vector $\boldsymbol{x}$ of extracted features from a musical signal [1]. We use the k-means clustering [17] algorithm to provide the basis functions for the network, where each function computes the distance from $\boldsymbol{x}$ to a prototype vector. Then RBF Network learns a logistic regression on top of that. Symmetric multivariate Gaussians are fit to the data from each cluster. For classification it uses the given number of clusters per class [2].

Support Vector Machines (SVM): Suppose we have a set of training music pieces, $\boldsymbol{x}_1,\ldots,\boldsymbol{x}_m$, where $\boldsymbol{x}_i \in R^d$. Each sample has a corresponding label $y_1,\ldots,y_m$ (where $y_i \in \{-1,1\}$) that indicates which of two classes each sample belongs to. Then we find a hyperplane ($\boldsymbol{w} \cdot \boldsymbol{x} + b$) to separates the data

To find the optimal separating hyperplane, we need to find the plane which minimize $|\boldsymbol{w}|^2/2$. By forming the Lagrangian, and solving the dual problem, this can be translated into the following [12, 13]:

$$\text{Minimize} \quad \sum_i a_i - \frac{1}{2}\sum_{i,j} a_i a_j y_i y_j \boldsymbol{x}_i \boldsymbol{x}_j \tag{1}$$

$$\text{Where} \quad \alpha_i \ge 0 \qquad \sum_i \alpha_i y_i = 0 \tag{2}$$

The α_i are the Lagrange multipliers; there is one Lagrange multiplier for each training sample.

K-Nearest Neighbors: K-Nearest Neighbors (k-NN) algorithm is a memory based learner [15]. The approach assumes that reasoning is based on direct reuse of stored experiences rather than on the application of knowledge abstracted from training. The similarity between the new instance and an example in memory is computed using a distance metric. In our experiments, we used IB1 [2], a k-NN classifier (k=1) that uses Euclidian distance metric. The main idea is to treat all music as points in the m-dimensional space and for an unseen music the algorithm classifies it by the nearest training music.

IV. Experiments

The dataset we used are publicly available on [7]. We conduct four music genre discrimination experiments: “Blues vs. Jazz”, “Country vs. Folk”, “Hip-Hop vs. Rock” and “Metal vs. Punk”. The selected principal components vary from 1 to 30. We use *hit rate*, defined as the number of cases predicted correctly, divided by the total number of cases, as the performance measure. 10-fold cross-validation (10-CV) is used for estimating classification performance. So hit rate actually means the overall hit rate of the 10-CV.

Fig. 3 shows the hit rate of Random Forest and the other three classifiers (RBF Network, SVM and IB1) for each of the music genre discrimination experiments. The results show that by PCA feature extraction on the original SOM features, all the four classifiers’ performance is enhanced. Random Forest is the best for “Country vs. Folk” and “Hip-Hop vs. Rock” and is the second best for the other two pairs. For “Country vs. Folk”, Random Forest achieves a maximum hit rate of 90% with the first 5 PCs.

V. Conclusions

In this paper, we investigate the use of SOM for extracting audio signal features for music genre classification. We introduce a new classification algorithm Random Forest for music analysis. Experiment results show that Random Forest gets the highest recognition rate of 90%. We demonstrate that PCA based feature dimension reducing can improve the classification performance.

References

[1] Douglas Turnbull and Charles Elkan: Fast Recognition of Musical Genre using RBF Networks. IEEE Transactions on Knowledge and Data Engineering,Volume 17, Issue 4, 580-584 April (2005)

[2] I.Witten and E.Frank: Data Mining - Practical Machine Learning Tools and Techniques with Java Implementation. Morgan Kaufmann (2000)

[3] T. Kohonen. Self-Organizing Maps.Springer-Verlag, Berlin (1995)

[4] Markus Frühwirth, Andreas Rauber: Self-Organizing Maps for Content-Based Music Clustering. In: Proc of the 12th Italien Workshop on Neural Nets (WIRN01), Vietri sul Mare, Italy, May 17-20. Springer,Germany (2001)

[5] Diamantaras, K.I. and Kung, S.Y.: Principal Component Neural Networks: Theory and Applications. John Wiley & Sons, INC (1996)

[6] Manikantan Ramadas, Shawn Ostermann and Brett Tjaden: Detecting Anomalous Network Traffic with Self-organizing Maps. Proceedings of Recent Advances in Intrusion Detection: 6th International Symposium, RAID 03, Pittsburgh, PA, USA, September 8-10 (2003)

[7] http://www.ifs.tuwien.ac.at/~andi/somejb/experiments/somejb1_free

[8] George Tzanetakis and Perry Cook: Musical Genre Classification of Audio Signals. IEEE Transactions on Speech and Audio Processing (2002)

[9] Changsheng Xu, Namunu C Maddage, Xi Shao,Fang Cao, and Qi Tian: Musical Genre Classification using Support Vector Machines, Tech. Rep., Laboratories for Information Technology, 21 Heng Mui Keng Terrace, Singapore 119613, (2003)

[10] R. Malheiro, RP Paiva, AJ Mendes, T. Mendes, and A. Cardos: A Prototype for Classification of Classical Music using Neural Networks. Proceedings of the Eighth IASTED International Conference PArtificial Intelligence and Soft Computing, (2004)

[11] Paul Scott: Music Classification using Neural Networks, Tech. Rep., Stanford University, Stanford,CA94305 (2001)

[12] Christopher Burges. A Tutorial on Support Vector Machines for Pattern Recognition. Data Mining and Knowledge Discovery, 2(2) (1998)

[13] Berhard Schölkopf, Chris Burges, and Vladimir Vapnik: Extracting Support Data for a Given Task. In Proceedings of the First International Conference on Knowledge Discovery and Data Mining (1995)

[14] Zeng, X.Y., Chen, Y.W., et al: A New Texture Feature based on PCA Maps and Its Application to Image Retrieval. IEICE Trans. Inf. and Syst., E86-D 929-936 (2003)

[15] Aha, D.W., Kibler, D., Albert, M.K.: Instance-based Learning Algorithms. Machine Learning, Vol.6, 37-66 (1991)

[16] Chien-Cheng Lee, Pau-Choo Chung, Jea-Rong Tsai, and Chein-I Chang: Robust Radial Basis Function Neural Networks. IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics, Vol. 29, No. 6, December (1999)

[17] H. Wang, et al.: Clustering by Pattern Similarity in Large Data sets. SIGMOD, 394-405 (2002)

[18] Bureau A, Dupuis J, Hayward B, Falls K, Van Eerdewegh P: Mapping Complex Traits using Random Forests. BMC Genet, 4(Suppl 1):S64 (2003)

[19] Breiman L: Random Forests. Machine Learning, 45:5-32 (2001)

[20] Breiman L, Friedman J, Olshen R, Stone C: Classification and Regression Trees. New York: Chapman & Hall (1984)

[21] Shengqiao Li, Adam Fedorowicz, Harshinder Singh, and Sidney C. Soderholm: Application of the Random Forest Method in Studies of Local Lymph Node Assay Based Skin Sensitization Data. Chem. Inf. Model., 45 (4) 952-964 (2005)

[22] Breiman L: Bagging Predictors. Machine Learning 24:123-140 (1996)

[23] Izmirlian G: Application of the Random Forest Classification Algorithm to a SELDI-TOF Proteomics Study in the Setting of a Cancer Prevention Trial. Ann NY Acad Sci, 1020:154-174 (2004)

[24] Liaw A, Wiener M: Classification and Regression by Random Forest. Rnews 2:18-22 (2002)

[25] Svetnik V, Liaw A, Tong C, Wang T: Application of Breiman's random forest to modeling structure-activity relationships of pharmaceutical molecules. Multiple Classier Systems, Fifth International Workshop, MCS 2004, Proceedings, 9¨C11 June 2004, Cagliari, Italy. Lecture Notes in Computer Science, Springer, 3077:334-343 (2004)

[26] Guo, L.; Ma, Y.; Cukic, B.; Singh, H.: Robust Prediction of Fault Proneness by Random Forests. Proceedings of the 15th International Symposium on Software Reliability Engineering, 417-428 (2004)

[27] Díaz-Uriarte R, Alvarez de Andrés S: Gene Selection and Classification of Microarray Data using Random Forest. BMC Bioinformatics, Vol. 7 (2006)

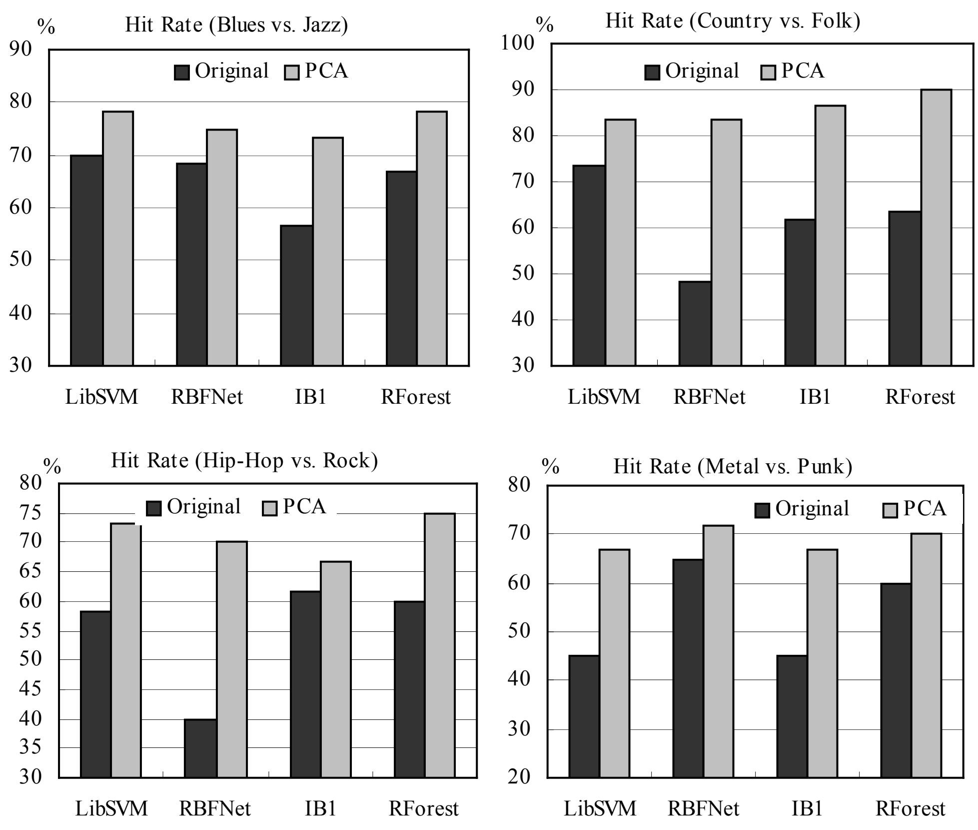

Fig. 3. Best hit rate of the four classifiers for music genre discrimination on both the original SOM feature space and the PCA extracted space. The X-axis denotes the classifiers. Original means hit rate of the classifiers on the original SOM features, PCA means the best hit rate of the classifiers after PCA feature dimension reducing.