Audio Music Genre Classification Using Different Classifiers and Feature Selection Methods

Yusuf Yaslan and Zehra Cataltepe
Istanbul Technical University
Computer Engineering Department
34469 Maslak, Istanbul/Turkey
E-mail: {yyaslan,zehra}@cs.itu.edu.tr

Abstract

We examine performance of different classifiers on different audio feature sets to determine the genre of a given music piece. For each classifier, we also evaluate performances of feature sets obtained by dimensionality reduction methods. Finally, we experiment on increasing classification accuracy by combining different classifiers. Using a set of different classifiers, we first obtain a test genre classification accuracy of around 79.6 ± 4.2% on 10 genre set of 1000 music pieces. This performance is better than 71.1 ± 7.3% which is the best that has been reported on this data set. We also obtain 80% classification accuracy by using dimensionality reduction or combining different classifiers. We observe that the best feature set depends on the classifier used.

1. Introduction

With the growth of the internet and multimedia systems applications that deal with the musical databases gained importance and demand for Music Information Retrieval (MIR) applications increased. Automatic analysis of the musical databases is one of the required components of the MIR. Most of the current music databases are indexed based on song title or artist name and in this format improper indexing can result in incorrect search results [1]. These methods become useless when text descriptions of the title or the artist name are not available. More effective systems extract important features from audio and classify the audio genre based on these features. However there aren't any strict distinguishing boundaries between audio genres and it is too difficult to find mathematical formulas that can identify them [2]. It is also hard to systematically describe the audio genres and no complete agreement exists in their definition [3].

In literature several MIR techniques, such as audio fingerprinting, audio identification, and score based retrieval have been proposed for genre classification [4]. Previous studies that deal with genre classification use symbolic representations of music such as MIDI files [3, 5, 6] or wav or mp3 [7, 8] audio files. Most of the proposed methods have two processing steps. The first one is frame-based feature extraction step where feature vectors of low-level descriptors of timbre, rhythm are computed from each frame. In the second step pattern recognition algorithms are applied on the feature vectors to achieve genre classification [7]. Most common features used for genre classification are; timbre, rhythm and pitch-related features [9]. Timbre related features are; FFT coefficients, Cepstrum and Mel Frequency Cepstrum Coefficients (MFCC), Linear Prediction (LP) coefficients, MPEG filterbank components, Spectral Centroid, Spectral Flux, Zero Crossing Rate, Spectral Roll-Off, low order statistics and Delta coefficients [7]. More detailed descriptions of the features can be found in [7, 9,10].

In this paper first audio genre classification performance of different classifiers are evaluated. Then, dimensionality reduction techniques, such as forward and backward feature selection and principal component analysis (PCA) are used to obtain smaller feature sets, and the test genre classification accuracy on these reduced feature sets are obtained. Finally, classifier combination is used to improve the genre classification accuracy.

2. Data Set

In order to be able to compare our work to previous studies, we use audio data set of Tzanetakis [9]. This data set contains 1000 music pieces each of 30 seconds length. There are 10 pieces from each of the following genres: classical (cl), country(co), disco(d), hiphop(h), jazz(j), rock(ro), blues(b), reggae(re), pop(p), metal(m). We use the freely available MARSYAS software to extract the audio features[11].

3. Genre Classification Using Audio Features

Using data set of [9] and Marsyas software [11], we obtained the following feature sets:

- BEAT: Beat related, 6 features.
- STFT: Short-Time Fourier Transform, 9 features.
- MFCC: Mel-Frequency Cepstral Coefficients, 10 features.
- MPITCH: Pitch related, 5 features.
- All: All features above, 30 features.

We experimented with 10 different classifiers, namely: Fisher (Fisher classifier), LDC (Linear classifier assuming normal densities with equal covariance matrices), QDC (Quadratic classifier assuming normal densities), UDC (Quadratic classifier assuming normal uncorrelated densities), NBC (Naïve Bayes Classifier), PDC (Parzen Density Based Classifier), KNN(Knearest neighbor with optimal k computed using leave one out cross validation), KNN1 (1 nearest neighbor), KNN3 (3 nearest neighbor), KNN5 (5 nearest neighbor) [12].

In figure 1 test classification performance of different classifiers (x axis) using different audio feature sets (each curve) is shown. In order to obtain the test performance curves, a classifier is trained using 90% of all the available data. The remaining 10% is used to measure the test performance. This procedure is repeated 30 times in order to get the error bars around each number.

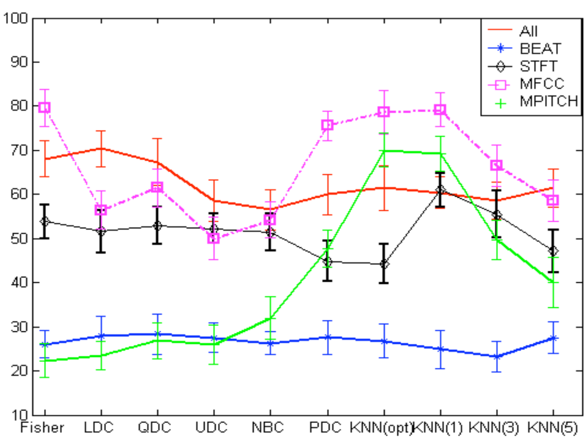

Figure 1: Test classification performance of different classifiers (x axis) using different feature sets.

According to figure 1, using Fisher classifier with MFCC features gives the best performance of 79.6 ± 4.2%. Using again MFCC with different flavors of KNN classifier results in performances between 59% and 79%. Using KNN(opt) with MPITCH features or using LDC with ALL features gives approximately 70% performance. These results are close or better than the ones reported for the same data set and using different classifiers, for example GMM with 3 Gaussians per class: 61± 4% [9]; Support Vector Machine (SVM): 69.1% ± 5.3% [6]; Linear Discriminant Analysis (LDA): 71.1±7.3% [13].

3.1. Feature Selection to Improve Classification Accuracy

We wanted to find out the set of features that results in the best classification accuracy for the data set. Identifying the best feature set could result in better classification accuracy as well as less time spent on feature extraction and classification. Although [13] experimented with different feature set combinations (for example, using set of MFCC features with set of STFT features), we think it is important to search for the right feature combination, which could use different features from each of the different four (BEAT, STFT, MFCC, MPITCH) original feature sets. In order to find the best feature subsets, we used the following approach: We first partitioned all available data into 90-10% training-test data. Then, we used i) forward selection ii) backward selection [14] algorithms to find the best set of features. We decided on which feature to add/subtract at a step based on the training error of a classifier trained on the training data and using the current set of features. Although the forward and backward feature selection algorithms make local decisions and may not be able to find the optimal set of features, they are still used in practice since exploring all possible combinations of features is usually infeasible.

In figures 2 and 3, we show the test classification accuracy of forward and backward feature selection for different number of features used (x axis) and different classification methods. QDC, a more complex classifier than others, show best performance of around 61% using 10 features, while UDC shows the best performance using 25 features. On the other hand, simpler classifiers, such as Fisher and Parzen show best performance using all features. Naïve Bayes shows very little improvement over its 78% performance after more than 10 features are used. In short, the classifier complexity plays an important role in how many features result in the best test classification accuracy. It is also apparent from figures 2 and 3 that forward and backward selection result in similar performance.

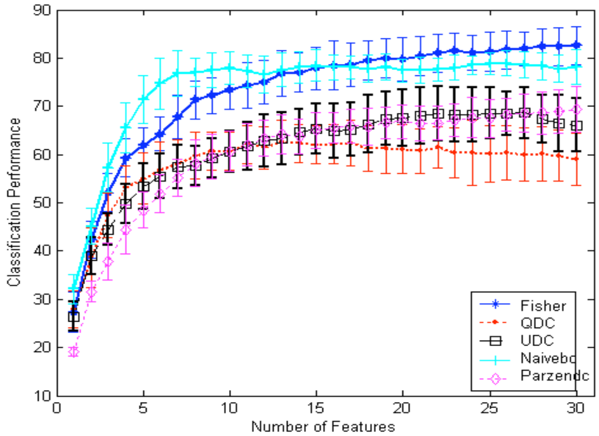

Figure 2: Forward feature selection performance

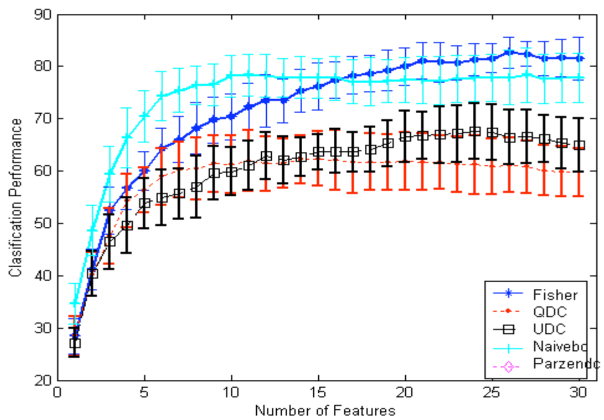

Figure 3: Backward feature selection performance

Table 1 shows the number of entries used from each feature set (beat, stft, mfcc, mpitch) for different classifiers when forward selection with different number of final features is used. MFCC features are selected by all classifiers, while BEAT features are among the least selected.

Table 1. The number of entries from each feature set (beat, stft, mfcc, mpitch) for different classifiers and total number of features (1,5,10,15,20,25) selected using forward selection.

	Fisher	Parzen	QDC	UDC	NaiveBC
1	0,0,1,0	0,1,0,0	0,1,0,0	0,1,0,0	0,1,0,0
5	0,3,2,0	0,3,1,1	0,4,1,0	0,3,2,0	0,2,2,1
10	1,4,4,1	1,4,3,2	0,6,4,0	0,4,3,3	1,2,5,2
15	2,6,6,1	4,6,3,2	1,7,6,1	3,4,5,3	3,3,5,4
20	4,6,8,2	6,7,5,2	1,7,9,3	6,4,6,4	5,4,6,5
25	4,9,8,4	6,9,7,3	5,7,10,3	6,6,8,5	6,6,8,5

3.2 Feature Set Reduction Using Principal Component Analysis to Improve Classification Accuracy

We also evaluated classification performance when Principal Component Analysis (PCA) [14] is used to reduce dimensionality of the ALL feature set. We mapped the feature set to 1, 5, 10, 15, 20, 25 and 30 dimensional feature sets using PCA. Test classification accuracies for different classifiers and different number of features obtained by using PCA are shown in figure 4. The classification performance of Parzendc, KNN(opt) and QDC increases to 80% when using all 30 features after applying PCA transform.

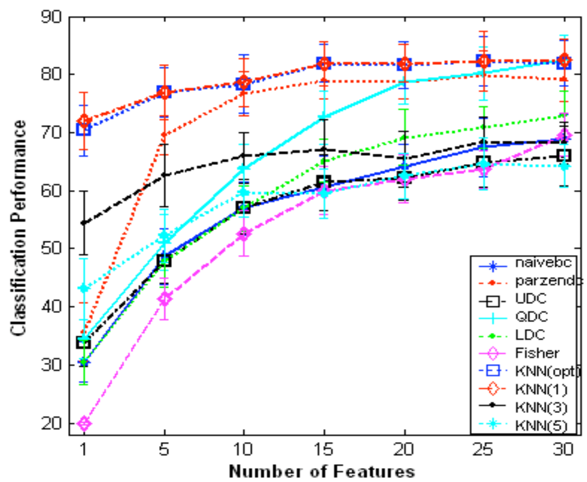

Figure 4: Classification performance of different classifiers using different number of features obtained by PCA.

3.3. Genre Classification by Combining Classifiers

Classifier combination is known to increase performance in certain cases [14]. We investigated the use of classifier combination for different feature sets and different classifier combinations. We selected classifiers that had good performances on their own and wanted to further improve their performance by combinations. In figure 5 the x axis shows the different classifier combinations: LNP (LDC – Naivebc - Parzendc), FNP(Fisher – Naivebc - Parzendc), KNP(knn(3) – Naivebc - Parzenc), FNU(Fisher – Naivebc - UDC), PNU(Parzendc, Naivebc, UDC).

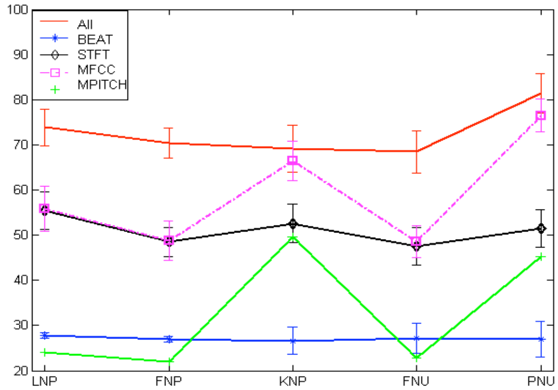

Figure 5: Classification performance of combined classifiers on ALL feature set.

Note that using ALL features and PNU combination 80% classification performance is achieved. Table 2 shows the confusion matrix for this combination.

Table 2: Confusion matrix of the PNU(Parzendc, Naivebc, UDC) combined classifiers for All features.

	co	**d**	**b**	**cl**	**h**	**j**	**m**	**p**	**re**	**ro**
co	80	1	4	0	2	4	0	4	4	1
d	0	84	2	0	6	0	0	3	2	3
b	0	2	85	0	1	1	0	0	8	3
cl	0	0	0	96	1	2	0	1	0	0
h	0	1	0	0	86	0	0	1	2	10
j	1	0	3	3	0	85	0	2	3	3
m	0	1	4	0	1	1	83	0	10	0
p	0	1	0	0	3	0	0	82	5	9
re	0	3	2	0	2	3	8	2	78	2
ro	0	1	3	0	7	0	0	8	0	81

4. Conclusion

In this paper we report on audio genre classification performance improvement using different classifiers, feature selection and dimensionality reduction methods and classifier combination techniques. By combining different classifiers we achieve a greater classification accuracy than reported in the literature on the same data set.

5. Acknowledgements

We would like to thank to George Tzanetakis for generously sharing his data set and MARSYAS software [http://marsyas.sourceforge.net]. We also would like to thank Bob Duin and friends for PRTools [www.prtools.org], which was used in some of the experiments we reported.

6. References

[1] Esmaili, S., Krishnan, S., and Raahemifar, K., “Content Based Audio Classification and Retrieval Using Joint Time-Frequency Analysis”, *ICASSP* 2004.

[2] Lippens, S., Martens, J.P., Leman, M., Baets, B., Meyer, H., and Tzanetakis, G., “A Comparison of Human and Automatic Musical Genre Classification”, *ICASSP* 2004.

[3] Basili, R., Serafini, A., and Stellato, A., “Classification of Musical Genre: A Machine Learning Approach”, *Proceedings of the International Symposium on Music Information Retrieval, ISMIR04,* Barcelona, 2004.

[4] Kurth, F., Müller, M., Röder, T., Damm, D., and Fremerey, C., “ A Prototypical Service for Real-Time Access to Local Context-Based Music Information”, *Proceedings of the International Symposium on Music Information Retrieval, ISMIR04*, Barcelona, 2004.

[5]McKay, C., Fujinaga, I., “Automatic Genre Classification Using Large High-Level Musical Features”, *Proceedings of the International Conference on Music Information Retrieval.* 525-30 , 2004.

[6] Tzanetakis, G., Ermolinskyi, A., and Cook, P., "Pitch Histograms in Symbolic and Audio Music Information Retrieval", *Journal of New Music Research* 32(2), pp. 143-152, 2003.

[7] Aucouturier, J.J., and Pachet, F., “Representing Musical Genre:A State of the Art”, *Journal of New Music Research*, Vol.32 No.1, pp 83-93, 2003.

[8] Liu, C. C., and Huang, C.S., “A Singer Identification Technique for content-based classification of MP3 music objects”, *Proceedings of the eleventh international conference on Information and knowledge management*, pp 438-445, Virginia USA, 2002.

[9] Tzanetakis, G. and Cook, P., "Musical Genre Classification of Audio Signals", *IEEE Transactions on Speech and Audio Processing*, vol.10 no.5, pp. 293-302, 2002.

[10] Kinney, M. F., Breebaart, J., “Features for Audio and Music Classification”, *International Symposium on Music Information Retrieval (ISMIR2003)* 2003.

[11] Tzanetakis, G., and Cook, P., “Marsyas: A Software Framework for Audio Analysis” *Organized Sound*, vol.4 issue 3, 2000.

[12] Bishop, C. M., *Neural Networks for Pattern Recognition,* Oxford University Press, 1995.

[13] Li, T., Tzanetakis, G., “Factors in Automatic Musical Genre Classification of Audio Signals”, *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2003.

[14] Webb, A., *Statistical Pattern Recognition*, Wiley, 2002.