Classification of Musical Audio Recordings

Igor Marić and Vladimir Risojević

Abstract—**In this paper, the automatic classification of musical audio recordings into a hierarchy of musical genres is explored. Three features sets for representing timbral texture, rhythmic content and pitch content of musical audio signals are reviewed. We give classification results using described features and a k-NN classifier. Accuracy of classification of 61% for ten musical genres is promising and this result is comparable to the results reported for human musical genre classification. We also analyzed the significance of individual features for classification and we show that timbral texture features yield the best results for this dataset.**

Index Terms—**Digital signal processing, music, musical genre, classification, audio recording.**

I. Introduction

THE creation of huge digital musical audio databases coming from both the restoration of existing analog archives as well as the creation of the new content is demanding reliable and fast tools for content analysis and description. These tools will enable searching, browsing, and interactive access to the musical content. In that context, musical genres are crucial descriptors since they have been widely used for years by music dealers and librarians to organize music catalogs, libraries, and stores. Musical genres are labels created and used by humans to explore similarities between musicians and compositions, as well as to organize musical collections. They have no strict definitions and boundaries as they arise through a complex interaction between the public, marketing, historical, and cultural factors. Despite their use, music genres remain a poorly defined concept, which makes the automatic classification problem a nontrivial task [1].

In this paper, an algorithm for automatic classification of musical audio recordings into a hierarchy of musical genres is proposed. The algorithm has two basic steps. The first step is the representation of an audio recording using features which are extracted using digital signal processing techniques. The second step is the classification of feature vectors into the predefined categories.

The paper is structured as follows. Features used for representation of musical audio recordings and their extraction are reviewed in Section II. Section III deals with the statistical evaluation of results of the proposed classifier and Section IV contains the concluding remarks and directions for future research.

II. Features of the Audio Signal

To be able to classify an audio signal, it is necessary to represent the signal using features, which reflect certain characteristics of the signal in some domain, e.g. time or frequency. Extracted features are then used for training of the classifier, and classification of a new signal is done on basis of its features extracted using the same procedure. In this paper we used three types of features, namely: timbral texture features, rhythmic content features and pitch content features.

A. Timbral texture Features

Audio signals are non-stationary, i.e. their spectral characteristics are changing in time. Therefore, they are analyzed in short time intervals within which the signal can be considered stationary and its parameters constant. This time interval is called the *analysis window*. When intervals with different spectral characteristics are interchanged with certain regularity, we can talk about sound texture. To examine this phenomenon quantitatively, it is necessary to observe the signal in a longer interval, which is called the *texture window*. Texture window consists of several analysis windows and its duration is approximately one second. Research on human subjects has shown that humans need only three seconds of music recording to identify the music genre [2]. Thus, we may conclude that humans for recognition of musical genres use the musical texture, in addition to the other characteristics of audio signals.

The musical texture can be quantitatively described using the following features which are based on the spectral characteristics of the signal:

1) *Spectral Centroid* is computed for each analysis window. It is the center of gravity of the magnitude spectrum of the window computed via STFT:

$$C_t = \frac{\sum_{k=1}^{N} k \cdot M_t(k)}{\sum_{k=1}^{N} M_t(k)} \tag{1}$$

where $M_t(k)$ is the magnitude of the Fourier transform in analysis window t and frequency bin k. Higher values of this feature correspond to more high frequencies in the analysis window. Value of the spectral centroid in music signal windows is greater than in voice signal windows, because musical instruments produce tones with higher frequencies than those of the human voice.

I. Marić is with the Faculty of Electrical Engineering, University of Banja Luka, Bosnia and Herzegovina (e-mail: igorica@teol.net).

V. Risojević is with the Faculty of Electrical Engineering, University of Banja Luka, Bosnia and Herzegovina (e-mail: vlado@etfbl.net).

2) *Spectral Rolloff* is defined as the frequency R_t below which 85% of the magnitude distribution is concentrated

$$\sum_{k=1}^{R_t} M_t(k) \approx 0.85 \cdot \sum_{k=1}^{N} M_t(k) \qquad (2)$$

The value of this feature is higher if more signal energy is contained in high frequencies.

3) *Spectral Flux* is defined as the squared difference between the normalized magnitudes of successive spectral distributions

$$F_t = \sum_{k=1}^{N} (N_t(k) - N_{t-1}(k))^2 \qquad (3)$$

where $N_t(k)$ and $N_{t-1}(k)$ are the normalized magnitudes of the Fourier transform in the current window *t*, and in the previous window *t-1*, respectively. Magnitude in each window is normalized with the sum of magnitudes at all frequencies in the current window. The spectral flux is a measure of the amount of local spectral change.

4) *Zero Crossings feature* is computed in the time domain. Its value is the number of zero crossings in the current window:

$$Z_t = \tfrac{1}{2} \sum_{m=1}^{M} \left| \operatorname{sgn}\left(x(m)\right) - \operatorname{sgn}\left(x(m-1)\right) \right| \qquad (4)$$

where $x(m)$ is the signal in the window *t*. This feature is higher for unvoiced than for voiced speech. In speech signal, windows of voiced and unvoiced speech are interchanged which means that windows with low and high values of this features are interchanged. On the other hand the number of zero crossings in the window for musical signals is pretty much constant.

5) *Low-Energy Feature* is defined as the percentage of *analysis windows* that have less RMS energy than the average RMS energy across the *texture window*. If there is a large number of "silent" analysis windows the value of this feature will be high. A large number of "silent" analysis windows is a characteristic of speech signals.

6) *Mel-Frequency Cepstral Coefficients:* Mel-frequency cepstral coefficients (MFCC) are perceptually motivated and they are frequently used in speech recognition systems. In order to compute MFCC we pass the signal through a filter bank with central frequencies uniformly distributed on a logarithmic transformed frequency axis. In this paper we used the ISP (Intelligent Sound Implementation) model implementation of MFCC [3].

Most of these features are time-variant, i.e. their value changes in analysis windows in which we consider the sound signal to be stationary. Spectral centroid, spectral rolloff, spectral flux, zero crossings, and MFCC are computed for each analysis window. Means and variances of these features are computed for each texture window. On the other hand, low energy feature is computed for a texture window and the value of this feature is added to the feature vector for a texture window. The signal is represented with a unique feature vector which is a mean value of feature vectors for individual texture windows.

B. Rhythmic Content Features

Although rhythm as a music concept is easy to understand, it is not easy to define. Human perception of rhythm is a subjective experience, but basically rhythm has always been described as repetition of emphasized elements or segments within the whole composition. The regularity of rhythm, the relation of the main beat to subbeats, and relative strengths of subbeats and the main beat are some of the characteristics we would like to represent in a feature vector. To compute the feature vector, it is necessary to perform beat detection, and construct *beat histogram* (BH). The procedure for beat detection is based on the discrete wavelet transformation (DWT) and is illustrated in Fig. 1 [2].

The following signal processing techniques are used for beat analysis:

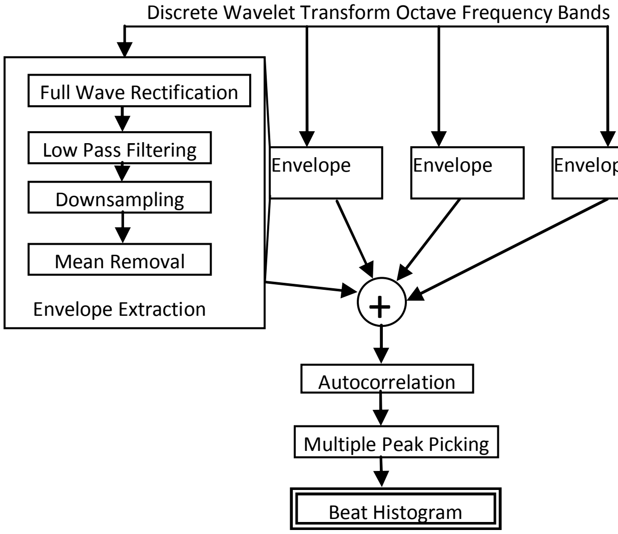

Fig. 1. Beat histogram calculation flow diagram.

1) *Full Wave Rectification:*

$$y(n) = \left| x(n) \right| \qquad (5)$$

is applied in order to extract the temporal envelope of the signal rather than the time domain signal itself.

2) *Low-Pass Filtering:*

$$y(n) = (1-\alpha) \cdot x(n) + \alpha \cdot y(n-1) \qquad (6)$$

using a one-pole filter with $\alpha = 0.99$ is used to smooth the envelope. Full wave rectification followed by low-pass filtering is a standard envelope extraction technique.

3) *Downsampling:*

$$y(n) = x(kn) \qquad (7)$$

where *k*=16 in our implementation. Because of the large periodicities for beat analysis, downsampling the signal reduces the computation time for the computation of the autocorrelation without affecting the performance of the algorithm.

4) *Mean Removal:*

$$y(n) = x(n) - E\left[x(n)\right] \qquad (8)$$

is applied in order to make the signal centered about zero for the autocorrelation stage.

5) *Enhanced Autocorrelation:*

$$y(n)=\frac{1}{N}\sum_{n}x(n)\cdot x(n-k)\,. \tag{9}$$

is a method used to detect periodicities (similarities) in the signal, i.e. beat in our case. Rhythmic features are computed using Enhanced *Summary AutoCorrelation Function* (ESACF) [4].

In order to compute the ESACF we clip the sum of envelopes to positive values, upsample the result with factor 2, and subtract it from the original clipped function. The same process is repeated with other integer factors such that repetitive peaks at integer multiples of the main beat are removed. The first three peaks of the ESACF that are in the appropriate range for beat detection are selected and added to a *beat histogram (BH).* The bins of the histogram correspond to beats-per-minute (bpm) from 60 to 220 bpm. Thus, peaks in the BH correspond to the self similarities of the signal.

Fig. 2 shows four beat histograms for 30s excerpts from different musical genres. In the upper left corner is a beat histogram of classics. This is a histogram of *Symphony No. 40* by Mozart.

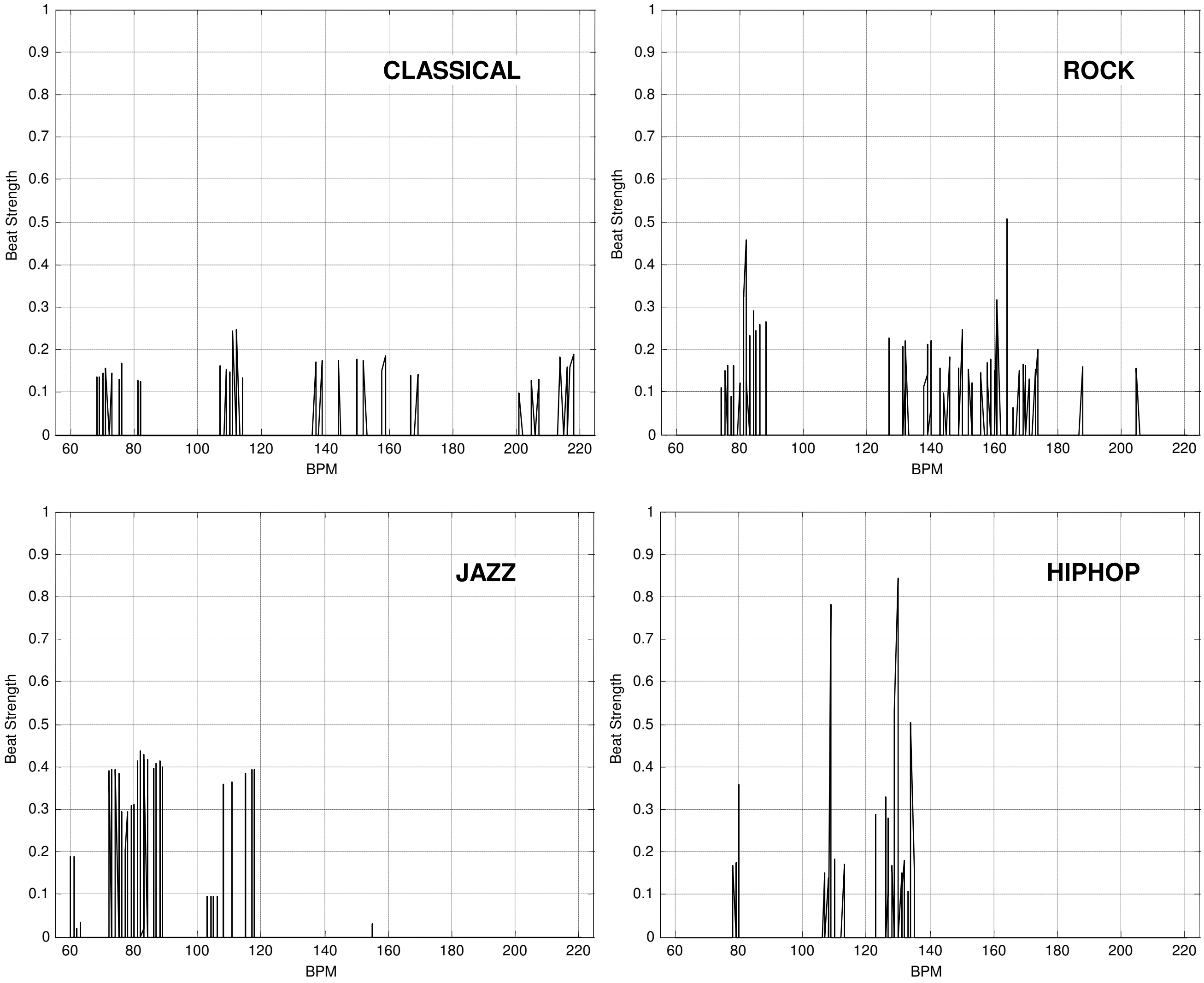

Fig. 2. Beat histogram examples.

We notice that in this composition there are no pronounced peaks in the histogram, as well as that the strength of the existing peaks is very low. This is a characteristic of the classical genre because of the complexity of multiple instruments in the orchestra, and because there is no pronounced rhythm section in classical music. Stronger peaks can be seen at the lower left, where the histogram for an excerpt from *I Can't Stop Loving You* by Ray Charles is shown. This composition belongs to the jazz genre. The values of the histogram are pretty much equal here, as well. There are peaks around 80bpm and 120bpm. In the upper right the histogram of a rock song *Come Together* by The Beatles is shown. Two largest peaks correspond to the mean beat, at approximately 80 bpm, and its first harmonic (twice the speed) at 160bpm. It is shown heuristically that the main beat usually corresponds to the first or second BH peak [2]. Peaks are more pronounced here, because the rock genre has stronger beat. The highest peaks in the lower right show strong rhythmical structure of the hip-hop song *Candy Shop* by 50Cent.

Fig. 2 indicates that the BH of different musical genres can be visually differentiated. The rhythm features include:

- **A0, A1**: relative amplitudes (divided by the sum of amplitudes) of the first and second histogram peak;
- **RA**: ratio of the amplitudes of the second and the first peak;
- **P1, P2**: periods of the first and second peak in bpm;
- **SUM**: overall sum of the histogram (indication of beat strength).

For the BH calculation, the DWT is applied to a window of 65 536 samples with a sampling rate of 22 050 Hz, which corresponds to approximately 3 s. This window is advanced by a hop size of 32 768 samples. This larger window is necessary to capture the signal repetitions at the beat and subbeat levels.

C. *Pitch Content Features*

In systems for audio analysis, pitch content is most often expressed by means of a Pitch Histogram (PH) [2]. PH is a statistical representation of the pitch content. Characteristics of tonality extracted from the PH form a set of tonality features. PH shows the number of appearances of each tone (note) in the musical audio recording. Histogram bins correspond to musical notes labeled using the MIDI note numbering scheme. Genres with more complex sound structures such as jazz or classical music tend to have a higher degree of pitch change than genres with simple chord progressions such as rock or pop music. As a consequence, pitch histograms for pop or rock music will have fewer and more pronounced peaks than the histograms of jazz or classical music. Algorithm for the calculation of PH is known under the name *Multiple Pitch Detection Algorithm* [4]. This algorithm is based on the model of two-channel pitch analysis. Block diagram of this model is shown in Fig. 3. Periodicity is detected by means of the autocorrelation function computed using:

$$x_2 = IDFT\left(\left|DFT(x_{low})\right|^k + \left|DFT(x_{high})\right|^k\right) \qquad (10)$$

where x_{low} and x_{high} are signals before periodicity detection in lowpass and highpass channels, respectively, and DFT and IDFT indicate discrete Fourier transform and its inverse. The parameter k determines the frequency domain compression (for standard autocorrelation k=2, optimal k=0.67). Values obtained from (10) are used to calculate ESACF, as described in Section IIB for BH. Three dominant peaks of the ESACF in each analysis window are added to the histogram. The values of the histogram will be the highest when these peaks match. The frequencies corresponding to each histogram peak are converted to musical pitches such that each bin of the PH corresponds to a musical note with a specific pitch. The musical notes are labeled using the MIDI note numbering scheme. The conversion from frequency to MIDI note number can be performed using

$$n = 12\log_2\left(\frac{f}{440}\right) + 69 \qquad (11)$$

where f is frequency in Hertz and n is the histogram bin (MIDI note number).

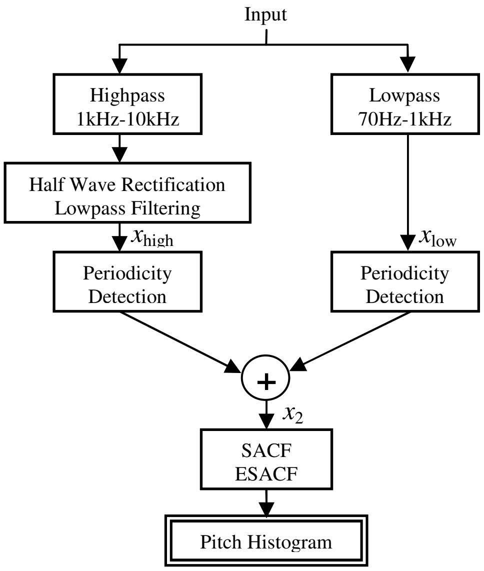

Fig.3. Multiple Pitch Detection Algorithm.

Two versions of the PH are created: *folded* (FPH) and *unfolded* histogram (UPH). The UPH is created using the equation (11). In the folded case, all notes are mapped to a single octave using

$$c = n \cdot \bmod 12 \qquad (12)$$

where c is the folded histogram bin (pitch class), and n is the unfolded histogram bin (or MIDI note number). Finally, the FPH is mapped to a circle of fifths histogram so that adjacent histogram bins are spaced a fifth apart rather than a semitone. This mapping is achieved by

$$c' = (7 \cdot c) \bmod 12 \qquad (13)$$

where c' is the new folded histogram bin after the mapping and c is the original folded histogram bin. The number seven corresponds to the seven semitones in a music interval of a fifth. That way, the distances between adjacent bins after the mapping are better suited for expressing tonal music relations (tonic-dominant) and the extracted features result in better classification accuracy. So, FPH contain information related to the music tonality content while UPH defines the range of tones.

PHs for examples from the jazz and rock genres are given in Fig. 4 and 5. We can see that rock music has fewer and more pronounced peaks in the histogram than jazz. This is a consequence of the fact that genres such as jazz or classical have a wider range of tonality than genres such as rock or pop.

The following features are computed from the UPH and FPH in order to represent pitch content:

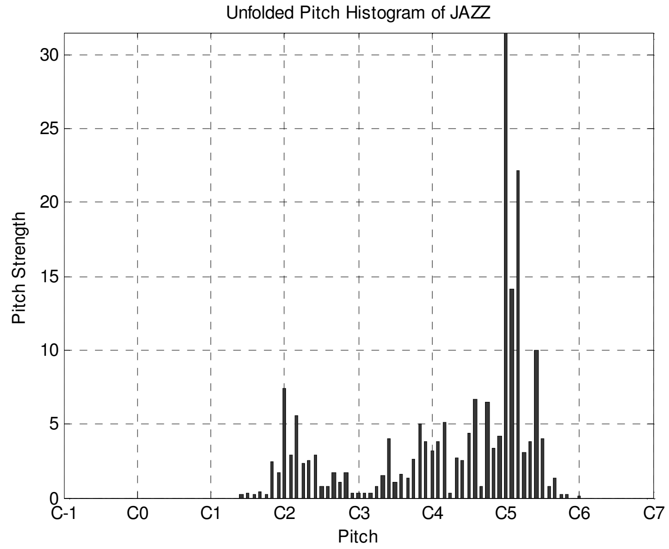

Fig. 4. UPH example for Jazz.

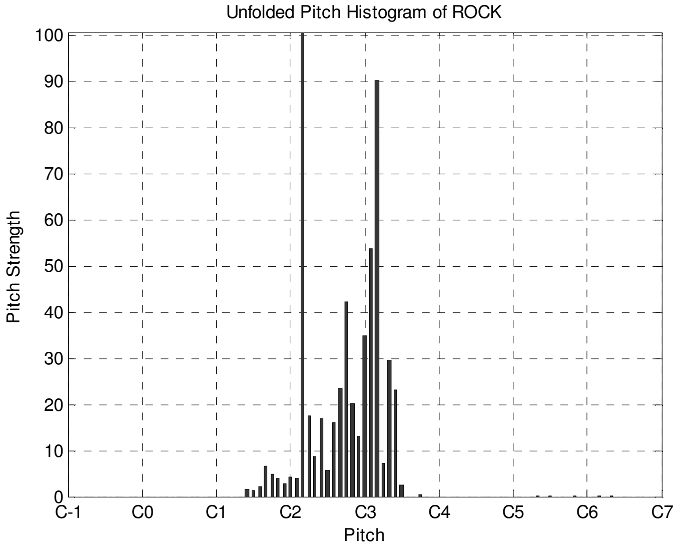

Fig. 5. UPH example for Rock.

- **FA0**: Amplitude of the maximum peak of the folded histogram. This corresponds to the most dominant pitch of the song. For tonal music this peak will typically correspond to the tonic or dominant chord. This peak will be higher for songs that do not have many harmonic changes.
- **UP0**: Period of the maximum peak of the unfolded histogram. This corresponds to the octave range of the dominant musical pitch of the song.
- **FP0**: Period of the maximum peak of the folded histogram. This corresponds to the main pitch class of the song.
- **IPO1**: Pitch interval between the two most prominent peaks of the folded histogram. This corresponds to the main tonal interval relation. For pieces with simple harmonic structure this feature will have value 1 or -1 corresponding to fifth or fourth interval (tonic-dominant).
- **SUM** The overall sum of the histogram. This is feature is a measure of the strength of the pitch detection.

For the computation of the PH, a pitch analysis window of 512 samples at 22 050 Hz sampling rate (approximately 23 ms) is used.

III. Classification of Audio Recordings

The test collection used in this paper consists of 1000 audio records. Each audio record is 30s long and recorded mono with 16 bits and sampling rate of 22050 Hz. Audio recordings contain music from 10 different genres whose hierarchy is shown in Fig. 6. Some of the musical examples are instrumental, and some include vocals. Used audio recordings are of different quality because they are collected from CDs, radio and the Web. This collection is also used in paper [2].

For classification we used a *K*-nearest neighbor (*k*-NN) classifier with Mahalanobis distance. 10-fold cross-validation algorithm is used for testing [6].

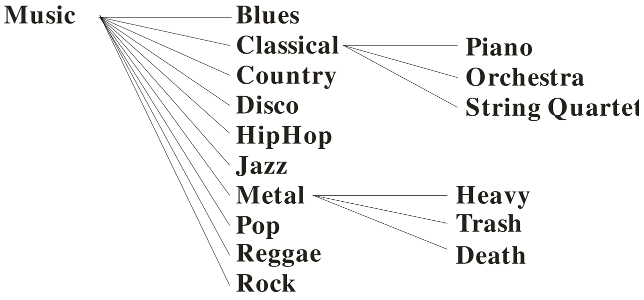

Fig. 6. Hierarchy of musical genres.

A. Classification Results

Total classification accuracy for ten musical genres is 61%. Percentages of examples which are classified correctly genre-wise are shown in Fig. 7. It can be seen that the classical music as a unique genre yields the best classification accuracy of 90%. Metal as a unique genre is another notable example. The lowest accuracy is obtained for rock genre, which can be explained by its relations to other genres. Table 1 provides a detailed insight into the classification of musical genres in the form of a confusion matrix Columns of this matrix correspond to the actual genres and its rows to the predicted genres. For example, cell in the 6th row and 2nd column has a value of 7, which means that 7% of *classical* music (column 2) is incorrectly classified as *jazz* (row 6). Percents of correctly classified genres are given on the diagonal of the matrix.

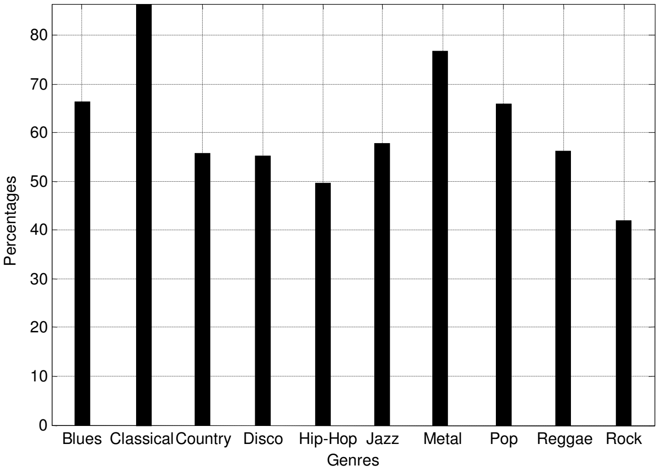

Fig. 7. Classification accuracy percentages.

Confusion matrix shows that the errors in the classification are similar to what a human would do. For example, *classical* music is classified as *jazz* in compositions that have strong rhythm, as in the works of Leonard Bernstein and George Gershwin.

TABLE I
GENRE CONFUSION MATRIX

	cl	co	di	hi	ja	ro	bl	re	po	me
cl	**67**	1	5	4	6	9	2	2	7	7
co	0	**87**	2	1	0	12	0	0	0	2
di	8	1	**56**	7	1	13	2	6	6	17
hi	3	1	5	**55**	11	1	2	7	5	7
ja	3	0	1	6	**50**	2	5	6	10	1
ro	7	7	9	1	1	**58**	0	3	1	2
bl	2	1	2	3	3	0	**77**	1	0	13
re	0	0	1	11	7	1	0	**66**	9	5
po	1	0	4	5	19	0	0	3	**56**	4
me	9	2	15	7	2	4	12	6	6	**42**

Blues genre overlaps with *jazz*, *rock* and *country* music, *country* with *jazz* and *rock*, *reggae* with *hip-hop*, etc. As mentioned, *rock* music has the worst classification accuracy and is easily confused with other genres, which is expected because of its broad nature. The confusion matrix for subgenres of the *classical* genre is given in Table II. Overall classification accuracy is 78%, which is good. It can be seen from the confusion matrix that the *orchestral music* is incorrectly classified as a *string quartet* in 28% of the cases, which is expected if you take into account that most orchestras usually include string instruments.

Confusion matrix in Table 3 presents the results of classification of the subgenres of the *metal* genre. Overall classification accuracy is 65%. Classification accuracy of the d*eath* metal subgenre is notable. This subgenre is easily distinguished by the specific style of singing and the color of the voice, as well as by melody and the way of playing.

TABLE II
CONFUSION MATRIX OF SUBGENRES OF THE CLASSICAL GENRE

	Piano	Orchestra	String Quartet
Piano	82	0	8
Orchestra	1	72	11
String Quartet	17	28	81

Heavy and *trash* metal are largely overlapping. It can be said that the *trash* contains *heavy* and vice versa, because *heavy* is the root of the *metal* music.

TABLE III
METAL CONFUSION MATRIX

	Heavy	**Trash**	**Death**
Heavy	**68**	**51**	**7**
Trash	**23**	**46**	**11**
Death	**9**	**2**	**82**

Table 4 shows the classification accuracy of *k*-NN classifiers for different values of the parameter *k* applied to the three sets of musical genres. Means and standard deviations of correctly classified examples in cross-validation are given. In the first row of the table the results for random classification are given.

TABLE IV
CLASSIFICATION ACCURACY MEAN AND STANDARD DEVIATION

	Genres(10)	**Classical(3)**	**Metal(3)**
Random	**10**	**33**	**33**
***k*NN(1)**	**58 ± 1**	**78 ± 5**	**55 ± 8**
***k*NN(3)**	**61 ± 1**	**72 ± 4**	**65 ± 6**
***k*NN(5)**	**60 ± 1**	**67 ± 8**	**54 ± 6**
***k*NN(7)**	**60 ± 1**	**59 ± 6**	**52 ± 4**

In Table V the individual importances of the proposed feature sets in the automatic classification of musical genres are given. Classification is done for *k* = 3. The first row in the table is random classification, while the last line corresponds to the full set of features. Numbers in brackets behind the labels of features represent are numbers of features for that individual set.

TABLE V
INDIVIDUAL FEATURE SET IMPORTANCE

	Genres(10	**Classical(3)**	**Metal(3)**
RND	**10**	**33**	**33**
PHF(5)	**35**	**48**	**48**
BHF(6)	**24**	**46**	**55**
STFT(9)	**45**	**56**	**44**
MFCC(10)	**59**	**70**	**54**
FULL(30)	**61**	**72**	**65**

As can be seen, features that are not based on the timbral texture but on pitch content (Pitch Histogram Features-PHF) and rhythm content (Beat Histogram Features-BHF) give worse results than the features based on the texture (STFT, MFCC) except in the case of the metal genre, where they perform approximately the same. Since the metal music is very melodic, rhythmic, harmonious and rapid greater accuracy is obtained using pitch and rhythm features. In all cases, the proposed set of features gives better results than random classification, which means that certain features give information about musical genres and musical content in general. The classification accuracy of the combined feature set (FULL(30)) in some cases is not significantly better compared to the classification accuracies of the individual feature sets. This fact does not necessarily imply that the features are correlated or do not contain useful information because it is possible that a specific file is correctly classified by two different feature sets that contain different and uncorrelated feature information. In addition, although certain features are correlated, the addition of each specific feature improves the classification accuracy. The rhythmic and pitch content features seem to play a less important role in the classification of the *classical* and *metal* datasets compared to the *genre* dataset. This is an indication that it is possible that *genre* datasets needs to be organized in a deeper hierarchy of subgenres.

IV. Conclusions and Future Work

Despite the fuzzy nature of genre boundaries, classification of musical audio recordings can be performed automatically with the accuracy that can be compared to human classification.

Three feature sets for representing timbral texture, rhythmic content and pitch content of music signals are computed and used for classification of musical audio recordings using a *k*-NN classifier, which was tested on a large collection of various audio recordings. Using the presented feature set classification accuracy of 61% has been achieved on a dataset consisting of ten musical genres, as well as 78% and 65% on *classical* and *metal* datasets.

We also evaluated the importance of individual feature sets for the classification of musical audio recordings. Furthermore, we examined the performance of the *k*-NN classifier, i.e. the mean and the standard deviation of the percentage of correctly classified examples, as a function of the parameter *k*, which affects the voting in the nearest neighbor algorithm. The success of the proposed features for musical genre classification reveals their potential for other tasks such as similarity retrieval, segmentation and audio thumbnailing.

In further work we plan to additionally improve the features, and even add new ones, as well as to work on improving the algorithms for their extraction. From the analysis of the results we believe that the genre hierarchy should be expanded both in width and depth. Two additional sources of information about the musical genre are the melody and the singer voice. Also in the future research the attention should be paid to other semantic descriptors such as the emotions and the style of singing. More research of the pitch content features could also possibly lead to better performance.

References

[1] N. Scaringella, G. Zoila, and D. Mlynek, "Automatic Genre Classification of Music Content," *IEEE Signal Processing Magazine*, Vol. 23, No. 2, pp. 133-141, 2006.

[2] G. Tzanetakis and P. Cook, "Musical Genre Classification of Audio Signals," *IEEE Transactions on Signal Processing*, Vol. 10, No. 5, 2002.

[3] S. Sigurdsson, K. B. Petersen, and T. Lehn-Schiøler, "Mel Frequency Cepstral Coefficients: An Evaluation of Robustness of MP3 Encoded Music," in *Proc. 7th International Conference on Music Information Retrieval, ISMIR 2006*, Victoria, Canada, 2006, pp. 286-289.

[4] T. Tolonen and M. Karjalainen, "A Computationally Efficient Multipitch Analysis Model," *IEEE Transactions on Speech and Audio Processing*, Vol. 8, No. 6, pp. 708-716, November 2000.

[5] D. Despić, *Teorija Muzike*, Zavod za udžbenike, Beograd, 2007.

[6] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, John Wiley & Sons, Inc., 2001.