Information Retrieval in Digital Libraries of Music

© Stefan Leitich Andreas Rauber

Department of Software Technology and Interactive Systems
Vienna University of Technology
http://www.ifs.tuwien.ac.at/ifs

Abstract

Recently, large music distributors have started to recognize the importance and business potential offered by the Internet, and are now creating large music stores for electronic distribution. However, these huge musical digital libraries require advanced access methods that go beyond metadata-based queries for artists or titles, facilitating the retrieval of a certain type of music, as well as the discovery of new titles by unknown artists similar to ones likings. Several approaches for content-based indexing of audio files in order to support genre-based access via sound similarity have been proposed recently. In this paper we review the approaches and evaluate their performance both on a kind of de-facto standard testbed, as well as a large-scale audio collection.

Keywords: Music Digital Library, Music Indexing, Audio Retrieval, MP3, Information Retrieval

1 Introduction

In spite of many open issues concerning copyright protection, and probably due to the sheer pressure of illicit music sharing created by a range of successful peer-to-peer platforms, the music industry is now starting to recognize and accept the potential of the Internet as a distribution platform [11]. Forerunners, like Apple's iTune on-line music store already create significant turnover, causing other providers to follow suite, be it either by offering their own music portal, or by relying on the help of B2B providers. These on-line music stores offer thousands of titles, with drastic increases in their holdings to be expected in the very near future, and their success seeming to be guaranteed. This is especially since music is one of the goods that is almost designed for electronic distribution, as it is by its very nature an intangible good, which can be "experienced", i.e. pre-listened to electronically before buying, i.e. downloading it, with both available bandwidths and compression qualities being entirely sufficient for wide-spread use and allowing distribution at marginal costs.

However, a critical success factor with any music repository will be the means of access it provides. Conventional approaches limit themselves to database searches for artists/composers/interprets, combined with title or album searches. While these ways of accessing a music collection are definitely a conditio sine qua non, they only allow the location of already known titles or artists, i.e. providing music that the consumer is knowing and actively looking for. Some sites, recognizing the need to support a more explorative means of access to their holdings, provide structured access via genre hierarchies, which are usually manually tendered to, albeit with only limited user satisfaction when it comes to communicating or accepting the pre-defined genre hierarchies. Additionally, the high intellectual and manual efforts to maintain a clean and clearly structured genre hierarchy become increasingly prohibitive with the fluctuation of new streams of music, the coming and going of new musical styles, and the variety of styles one and the same band is playing. This frequently results in either overly coarse or un-usably detailed genre hierarchies of just a few or several hundreds of branches [10], or music being labeled by, say, a bands or artists "typical" genre tag, rather than the musical style represented by a specific title.

In order to counter these effects and to facilitate access to music based on the audio content, indexing techniques are being devised that extract characteristic features from the audio signal, such as from any audio CD, WAV or MP3 files. These features, in turn, are being used to identify and describe the style of a particular piece of music. Combined with specific metrics or machine learning algorithms these may be used to either classify new music into predefined genre hierarchies, to group music according to its perceived sound similarity, or simply to retrieve titles from an audio database that sound similar to a given query sound. In this paper we provide a review of the most prominent approaches used for content-based indexing and retrieval of audio files. More specifically, we use two prominent music indexing systems, namely *MARSYAS* [18] as well as the *SOMeJB* System [12, 14]. We compare their performance in a controlled study, applying them both to a de-facto standard testbed of about 1.000 files of music stemming from 12 different genres, as well

Proceedings of the 6th Russian Conference on Digital Libraries RCDL2004, Pushchino, Russia, 2004

as to a much larger collection of almost 10.000 files without pre-defined genre assignment. For the larger collection the audio files have been segmented, with different degrees of overlap between neighboring segments, in order to study both the general capabilities as well as sensitivity of the approaches to local variations. Both automatic evaluation as well as a usability study are conducted in order to quantify the performance of the approaches.

The remainder of the paper is structured as follows: Section 2 provides an overview of related work on audio indexing and style-based retrieval. Section 3 introduces a set of features used in the MARSYAS framework, while Rhythm Patterns are presented in Section 4. Section 5 discusses experimental results on both the standard test set as well as the larger audio repository, combining both the outcome of automatic evaluation as well as feedback obtained during a user study. The results of our findings are summarized in Section 6.

2 Related Work

A significant amount of research has been conducted in the area of content-based music retrieval, cf. [3, 6]. Methods have been developed to search for pieces of music with a particular melody. Users may formulate a query by humming a melody, which is then usually transformed into a symbolic melody representation. This is matched against a database of scores given, for example, in MIDI format. Research in this direction is reported in, e.g. [1, 2]. Other than melodic information it is also possible to extract and search for style information using the MIDI format. Yet, only a small fraction of all electronically available pieces of music are available as MIDI. A more readily available format is the raw audio signal, which all other audio formats can be decoded to. A system where hummed queries are posed against an MP3 archive for melody-based retrieval is presented in [8]. Both melody-based retrieval of music, as well as access to music available in MIDI-format are outside the scope of this paper.

Rather, this paper focuses on methods extracting style or genre information directly from the audio content, i.e. by indexing e.g. MP3 or WAV files. This kind of genre based organization and detection has gained significant interest recently. One of the first works to incorporate psychoacoustic modeling into the feature extraction process and utilizing the SOM for organizing audio data is reported in [5]. A first approach, classifying audio recordings into speech, music, and environmental sounds is presented in [21]. A system performing trajectory matching using SOMs and MFCCs is presented in [17]. Specifically addressing the classification of sounds into different categories, loudness, pitch, brightness, bandwidth, and harmonicity features are used in [20] to train classifiers. A wide range of musical surface features is used by the *MARSYAS* system [18, 19] to organize music into different genre categories using a selection of classification algorithms. The features extracted by this system will be discussed and evaluated in more detail in this paper. The second set of features to be evaluated are *Rhythm Patterns* used in the *SOMeJB* system [12, 14].

3 The MARSYAS System

The *MARSYAS* system, as presented in [18, 19] and available via the project homepage [9], is the implementation of a general framework for the extraction of various content-based features for audio files. It follows a client-server based architecture, is implemented in C++, and is available for download from the SourceForge repository. The set of features implemented in the system analyzes music with respect to timbral, rhythmic, as well as pitch characteristics. Some of these features are particularly aimed at speech vs. audio classification, whereas others are targeted towards genre classification of music. For the experiments reported in this paper we use the following subset of features, recommended for genre classification:

- FFT: This set of 9 features consists of the means and variances of the spectral centroid, rolloff, flux and zerocrossings, based on the Short Time Fourier Transform (STFT) of the signal, as well as a low energy feature. *Spectral centroid* is the center of gravity of the energy of the STFT, with "brighter" signals having a stronger high-frequency part, resulting in higher spectral centroid values. The *spectral rolloff* is calculated as the frequency below which 85% of the energy is concentrated. The amount of local spectral change is measured through *spectral flux*, calculated as the squared differences between the normalized magnitudes of successive spectral distributions, measuring temporal changes in the frequency domain. *Low energy* is the percentage of texture windows that have less than average energy, being a particularly good discriminator for speech against music discrimination. *Zero crossings* of the frequency signal provides a good measure of the noisiness of a signal, differentiating between voiced and unvoiced audio signals.

- MFCCs: The first five Mel-Frequency Cepstral Coefficients, i.e. FFT bins that are grouped and smoothed according to the Mel-frequency scaling are used to describe the content of an audio signal. The filter-bank used for grouping the audio signal consists of 13 linearly-spaced filters below 1kHz, followed by 27 log-spaced filters above. The filter-bank is perceptually motivated, and similar in principle to the bark-scale used for the Rhythm Patterns of the *SOMeJB* system.

- MPitch: This set of features represents harmonic content based on multiple pitch analysis, calculating a pitch histogram over analysis windows of 20ms length, using both unfolded, as well as folded pith histograms, i.e. histograms

where the notes are mapped onto a single octave scale.

- Beat: This set of features represents the beat structure of music calculated by a beat detection algorithm based on Discrete Wavelet Transform, analyzing beats between 40 and 200 bpm. This feature is closely related to our Rhythm Patterns described in Section 4, computing the histogram over the whole spectrum rather than individually for different frequency bands, and within a more restricted value range.

This results in 30-dimensional feature vectors for each piece of music. As these attributes have significantly different value ranges, attribute-wise normalization to the interval [0,1] is performed, allowing for subsequent comparison and retrieval of weight vectors using Euclidean distance.

4 Rhythm Patterns and the SOMeJB System

Starting from a standard Pulse-Code-Modulated (PCM) signal, a pre-processing step is performed, where stereo channels are combined into a mono signal, which is further down-sampled to 11kHz. The feature extraction process for the Rhythm Patterns itself is composed of two stages [13]. First, the specific loudness sensation in different frequency bands is computed, which is then transformed into a time-invariant representation based on the modulation frequency.

Using a Fast Fourier Transform (FFT), the raw audio data is further decomposed into frequency ranges using Hanning Windows with 256 samples (corresponding to 23ms) with 50% overlap, resulting in 129 frequency values (at 43Hz intervals) every 12 ms. These frequency bands are further grouped into so-called critical bands, also referred to by their unit bark [22], by summing up the values of the power spectrum between the limits of the respective critical band, resulting in 20 critical-band values. A spreading function [16] is applied to account for masking effects, i.e. the masking of simultaneous or subsequent sounds by a given sound. The spread critical-band values are transformed into the logarithmic decibel scale, describing the sound pressure level in relation to the hearing threshold. Since the relationship between the dB-based sound pressure levels and our hearing sensation depends on the frequency of a tone, we calculate loudness levels, referred to as *phon*, using the equal-loudness contour matrix. From the loudness levels we calculate the specific loudness sensation per critical band, referred to as *sone*.

To obtain a time-invariant representation, reoccurring patterns in the individual critical bands, resembling rhythm, are extracted in the second stage of the feature extraction process. This is achieved by applying another discrete Fourier transform, resulting in amplitude modulations of the loudness in individual critical bands. These amplitude modulations have different effects on our hearing sensation depending on their frequency, the most significant of which, referred to as fluctuation strength [4], is most intense at 4Hz and decreasing towards 15Hz (followed by the sensation of roughness, and then by the sensation of three separately audible tones at around 150Hz). We thus weight the modulation amplitudes according to the fluctuation strength sensation, resulting in a time-invariant, comparable representation of the rhythmic patterns in the individual critical bands. To emphasize the differences between strongly reoccurring beats at fixed intervals a final gradient filter is applied, paired with subsequent Gaussian smoothing to diminish un-noticeable variations.

The resulting 1.200 dimensional feature vectors (20 critical bands times 60 amplitude modulation values) capture beat information up to 10Hz (600bpm), going significantly beyond what is conventionally considered beat structure in music. They may optionally be reduced down to about 80 dimensions using PCA. These Rhythm Patterns (RP) are further used for similarity computation. MATLAB toolboxes for feature extraction are available for download via the *SOMeJB* project homepage [15].

5 Experiments

5.1 Data Sets

For the experiments reported in this paper we use two different sets of music. The first set of music (Collection 1) consists of a collection of 9360 titles from a wide range of genres, including mostly western music, but also smaller numbers of ethnic music from various regions. The files in this collection have been segmented into segments of 30 seconds length, where four different segments were created from every file, namely Segment 1: seconds 30-60, 2: 45-75, 3: 70-100, and 4: 150-180. This segmentation was chosen in order to evaluate the locality and stability of the various approaches for retrieval, i.e. searching a particular piece as well as type of music given a short segment of it. The four segments exhibit different types of overlap, ranging from 50% overlap of segments one and two, via 5 seconds overlap between segments two and three, up to no overlap between segments one and both 3 and four, where the former two are still close to each other, i.e. only 10 seconds apart.

The smaller collection (Collection 2) consist of 1203 pieces of music, each of 30 seconds length and organized into 12 categories, namely Ambient, Ballad, Blues, Classical, Country, Disco, Hip-Hop, Jazz, Metal, Pop, Reggae, and Rock. This collection of music, put together by George Tzanetakis [18], has evolved into a kind of standard-set for music IR.

5.2 Performance Evaluation

These two sets of music form the basis for two types of experiments, namely retrieval based on query vectors performed on both data collections (Experiments 1 & 2) using recall/precision-based evaluation, as well as a usability study done with a group of stu-

i	MARSYAS	RP	best
1	1440	6037	9360
3	2410	12924	28080
5	2821	14500	28080
10	3410	16124	28080
20	4135	17147	28080

Table 1: Absolute recall values at (i) 20, 10, 5, 3 and 1 for the music Collection 1 in case of feature extraction by the different prototypes and in a best case scenario.

R_i	MARSYAS	RP	best
1	0.0513	0.2150	$0.\dot{3}$
3	0.0858	0.4603	1
5	0.1005	0.5164	1
10	0.1214	0.5742	1
20	0.1473	0.6106	1

Table 2: Recall values (R_i) at (i) 20, 10, 5, 3 and 1 for the music Collection 1 in case of feature extraction by the different prototypes and in a best case scenario.

P_i	MARSYAS	RP	best
1	0.1538	0.6450	1
3	0.0858	0.4603	1
5	0.0603	0.3098	0.6
10	0.0364	0.1723	0.3
20	0.0221	0.0916	0.15

Table 3: Precision values (P_i) at (i) 20, 10, 5, 3 and 1 for the music Collection 1 in case of feature extraction by the different prototypes and in a best case scenario.

dents on the larger Collection 1 dataset (Experiment 3).

For the retrieval performance evaluation on Collection 1 the segments of set 1 starting at second 30 are used as query set and the others (segments starting at second 45, 70 and 150) form the data set which the queries are performed on. This resembles the process of searching a piece of music in a digital music library. It serves to evaluate the locality and stability of the feature representation with respect to distance in time of the segments in the database from the query segment.

The second recall and precision evaluation is based upon Collection 2. Here, query and data collection are the same and therefore pairwise distances between the files are computed. While the retrieval of the identical piece of music based on different segments is the target of the experiment in Collection 1, the target ground truth in Collection 2 is the genre labels assigned to the titles, i.e. retrieving all pieces of Reggae music for a Reggae query object.

Recall and precision are computed for answer sets with a size of 1, 3, 5, 10 and 20. Precision P_i and recall R_i are defined by Equation 1, where N_{rd} denotes the number of relevant titles retrieved, N_{rt} are the number of relevant pieces of music in the whole music collection, N_i stands for the total number of pieces of music retrieved and i is the size of the answer set.

$$P_i = \frac{N_{rd}}{i} \quad , \quad R_i = \frac{N_{rd}}{N_{rt}} \tag{1}$$

Secondly, we report on results from a user study, where perceived similarity in style is evaluated for the top-10 titles retrieved for a small set of selected query vectors. User were presented the query title as well as the 10 top-ranking songs for each feature set based on simple Euclidean-distance based retrieval, and asked to rate them in three categories with respect to genre similarity, i.e. *very similar, somewhat similar* and *not similar*.

5.3 Experiment 1: Segment Retrieval

In Tables 1 and 2 the results of the recall evaluation on Collection 1 can be found. The labels MARSYAS and RP denote the related prototypes used for the feature extraction.

Given 28080 MP3's in the data set of the Collection 1 (9360 titles times 3 segments each) in the best case all of the three relevant MP3's per query should be in the answer set and therefore the best case are 28080 relevant pieces of music retrieved. (Obviously, in case of an answer set with a size of one ($i = 1$) only 9360 relevant pieces of music can be retrieved. Expressing this in relative recall values the best case scenario with $i = 1$ gives a recall of $0.\dot{3}$) As the absolute (Table 1) and relative (Table 2) recall values show, the Rhythm Patterns features outperform the other approach, achieving a recall rate of about 50-60%, as opposed to the MARSYAS feature set with about 10%.

This trend is similar for the precision values provided in Table 3. (Again, precision values in the best case are smaller than one for answer sets bigger than three ($i > 3$), as there are only three relevant pieces of music in the collection for each query.) The precision values are higher for the RP feature set, and in more than 45% of all cases the 3 top-ranked retrieved results are the 3 segments from the piece of music used as a query.

It should be noted, however, that neither of the two feature sets were specifically designed for identity-detection, i.e. for retrieving a specific piece of music based on an arbitrary segment of it. Both representations were primarily developed with a focus on genre detection, i.e. capturing the characteristics of a specific style of music. Systems optimized for identifying a specific piece of music using audio fingerprinting (but not capturing similarities between pieces of music of the same genre), form a specific area of music IR research, cf. [7] for an example of such a system. Thus, performance values on retrieval in this setting may not be taken directly as a quality measure of the two feature sets.

Regarding the stability of the feature representation the second segment of a piece piece of music should be most similar to the query segment because they overlap each other for the half of their length. As the distance in time between the segments grows, one can anticipate a growing distance between the

segment	MARSYAS	RP
45	2032	5378
70	740	1632
150	293	921

Table 4: Number of times a segment of set 2, 3, and 4 is the highest-ranking segment for a query with a segment from set 1.

segment	MARSYAS	RP
45	4.5723	2.3708
70	5.694	3.7867
150	5.7061	4.3277

Table 5: Average position of the relevant segments in an answer set with size $i = 20$, for the different analysis methods.

respective feature vectors. To determine the stability of a feature vector over the analyzed segments of a piece of music, a count of the different best ranked segments is done (answer set size $i = 20$). Table 4 shows for each analysis method that, the closer in time the segments are, the better they get recognized. So the segments starting at second 45 of playtime of the original pieces of music are most often the best ranked ones, followed by the segments starting at second 70 and then by segments starting at second 150. In Table 5 the absolute average position of the segments in the answer set ($i = 20$) is listed. The ranking in the RP case is better than the MARSYAS ranking, but also a bit more spread apart.

Tables 6 to 8 depict the corresponding recall values, considering only one segment as valid answer to the query segment. The values in these three tables show, corresponding to the average position (see Table 5) and the stability information given in Table 4, a similar relative loss of recall for segments located at second 70 and 150 of playtime against the segment located at second 45 for the MARSYAS approach. The RP approach shows a significant smaller relative loss of recall for the segments located further away in playtime from the query segment, aside from the good recall values themselves.

Figure 1 lists the relative distance to the query vector within an answer set of size 20. The characteristics of the increase in relative distance is nearly the same for MARSYAS and RP. The spacing of distances between ranks is not linear, exhibiting faster increases in distance around the query vector, and leveling out afterwards.

In order to take a closer look at the specific performance of the two systems, Table 9 and Table 10 show the 20 best ranked answers to a query conducted with the segment starting at second 30 of the the track *B Please II* by artist *Eminem* from the album *The Marshall Matters* for the RP and MARSYAS system, respectively. The tables show the raw answers to the queries, containing multiple entries of the same song if different segments where found as similar.

The RP results are very homogeneous, partially

R_i^{S45}	MARSYAS	RP
1	0.1057	0.4420
3	0.1563	0.6840
5	0.1783	0.7262
10	0.2093	0.7677
20	0.2475	0.7946

Table 6: Recall values (R_i) at (i) 20, 10, 5, 3 and 1 for Collection 1, considering segments from set 2 as the only valid answers.

R_i^{S70}	MARSYAS	RP
20	0.1201	0.6034
10	0.0953	0.5591
5	0.0755	0.4906
3	0.0629	0.4204
1	0.0338	0.1333

Table 7: Recall values (R_i) at (i) 20, 10, 5, 3 and 1 for Collection 1, considering segments from set 3 as the only valid answers.

due to the strong rhythmic characteristics of the Hip-hop genre. The MARSYAS 10 result is also very consistent, with only two pieces of music not particularly well placed in the result set: Number 10 is a disco-style title but with hip hop vocals. Result number 13 is a mellow pop song with a strong Hip-hop-style beat. Over all the results for the Hip-hop retrieval task are very consistent and reasons for misplacing songs are traceable.

The laid back jazz song *What's new?* by artist *Silje Nergaard* from the album *Port of Call* causes much bigger confusion. The RP results come up with the artist *Tori Amos* three times, who performs with piano and voice like in the query file but has a stronger singer/songwriter association than Jazz. Also a German songwriter, *Reinhard May*, is retrieved twice, and does definitely not fit in the answer set. A piece of music of *Queen* is also found similar, which is not a jazz title at all, but listening into it, the misplacement becomes understandable. The song is piano and voice only and has a similar mood like the query song. Results of the MARSYAS feature set do not perform too well either on that title. Actually, only two results in the answer set do fit, all other tracks are classic, mellow pop in the broadest sense, instrumentals, or soul music. These results show that a numeric only evaluation of the results a music information retrieval system produces may easily lead to a false estimation of the performance of such a system, motivating the user study described in Section 5.5.

5.4 Experiment 2: Genre based Retrieval

Tables 11 and 12 list the results of the evaluation using Collection 2 for genre-based retrieval. Here, a fixed answer set size of $i = 10$ is used and the performance in different genres is displayed. Query and data set are the same and all pieces of music of the same genre are considered to be correct answers to a query. The number of titles in a genre is enlisted

rank	artist	album	title	seg.
1	Eminem	The Marshall Mathers	B Please II	70
2	Eminem	The Marshall Mathers	B Please II	45
3	Eminem	The Marshall Mathers	B Please II	150
4	Outkast	ATLiens	ATLiens	70
5	A Tribe Called Quest	Beats Rhymes & Life	Mind Power	70
6	A Tribe Called Quest	Beats Rhymes & Life	Mind Power	45
7	A Tribe Called Quest	Beats Rhymes & Life	Mind Power	150
8	Shaggy		It Wasn't Me	45
9	Shaggy		It Wasn't Me	45
10	A Tribe Called Quest	Beats Rhymes & Life	The Hop	45
11	Outkast	ATLiens	ATLiens	150
12	Cypress Hill (feat. Eminem)		Rap Superstar	70
13	Mobb Deep	Hell On Earth	Nighttime Vultures	45
14	Mobb Deep	Hell On Earth	Nighttime Vultures	70
15	Mobb Deep	Hell On Earth	Nighttime Vultures	150
16	A Tribe Called Quest	Beats Rhymes & Life	The Hop	150
17	Shaggy		It Wasn't Me	70
18	Shaggy		It Wasn't Me	70
19	Shaggy		It Wasn't Me	150
20	A Tribe Called Quest	Beats Rhymes & Life	Phony Rappers	45

Table 9: 20 best ranked answers to a query with the segment starting at second 30 of the track *B Please II* by artist *Eminem* from the album *The Marshall Matters* using RP features.

rank	artist	album	title	seg.
1	Eminem	The Marshall Mathers	B Please II	45
2	Mobb Deep	Hell On Earth	Extortion	150
3	Mobb Deep	Hell On Earth	Can't Get Enough Of It	150
4	Eminem	Slim Shady LP	97' Bonnie & Clyde	70
5	Eminem	The Marshall Mathers	Under the Influence	45
6	Wu-Tang Clan	The W	One Blood Under W	45
7	Busta Rhymes	When Disaster Strikes	Turn It Up	45
8	Fettes Brot	Amnesie	Lieblingslied	70
9	Mobb Deep	Hell On Earth	Animal Instinct	70
10	Fettes Brot	Amnesie	Nordisch By Nature	150
11	Wyclef Jean	Masquerade	Oh What a Night	70
12	Eminem	The Marshall Mathers	Drug Ballad	70
13	Morcheeba	Fragments of Freedom	Shallow End	45
14			Get Up	45
15	Busta Rhymes	Extinction Level Event	Iz They Wildin Wit Us & Gettin Rowdy Wit...	45
16	Eminem	The Marshall Mathers	B Please II	150
17	Eminem	The Eminem Show	Drips	70
18	Absolute Beginner	Bambule	Showmaster	70
19	Eminem	Slim Shady LP	97' Bonnie & Clyde	45
20	Outkast	ATLiens	ATLiens	45

Table 10: 20 best ranked answers to a query with the segment starting at second 30 of the track *B Please II* by artist *Eminem* from the album *The Marshall Matters* using MARSYAS features.

R_i^{S150}	MARSYAS	RP
1	0.0144	0.0697
3	0.0382	0.2764
5	0.0475	0.3324
10	0.0597	0.3958
20	0.0741	0.4340

Table 8: Recall values (R_i) at (i) 20, 10, 5, 3 and 1 for Collection 1, considering segments from set 4 as the only valid answers.

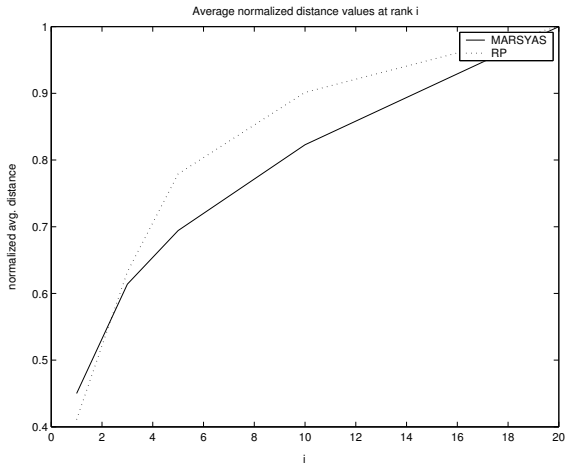

Figure 1: Average distance values between the query vector and documents up to rank 20, normalized to make the two feature spaces comparable.

Genre	n	MARSYAS	RP	best
Ambient	71	211	252	710
Ballad	45	70	79	450
Blues	108	249	292	1080
Classical	110	538	481	1100
Country	104	238	312	1040
Disco	102	281	375	1020
Hip-hop	106	225	385	1060
Jazz	105	342	325	1050
Metal	104	431	336	1040
Pop	131	401	306	1310
Reggae	114	387	272	1140
Rock	103	203	207	1030
all	1203	3576	3622	12030

Table 11: Absolute recall at 10 for Collection 2.

Genre	MARSYAS	RP	best
Ambient	0.0431	0.0514	0.1449
Ballad	0.0362	0.0408	0.2324
Blues	0.0217	0.0255	0.0943
Classical	0.0453	0.0405	0.0926
Country	0.0224	0.0294	0.0980
Disco	0.0275	0.0368	0.1000
Hip-hop	0.0204	0.0349	0.0961
Jazz	0.0316	0.0300	0.0971
Metal	0.0406	0.0317	0.0980
Pop	0.0237	0.0181	0.0775
Reggae	0.0303	0.0213	0.0893
Rock	0.0195	0.0199	0.0990
avg	0.0302	0.0317	0.1099

Table 12: Recall at 10 for Collection 2.

in column n in Table 11, also specifying the number of queries performed. The best case scenario for the recall values results in rather small values, because in an answer set of size 10 only a fraction of all relevant titles can be located. For each query there would be $n-1$ relevant pieces of music in the collection and this value is always bigger than the answer set size ($i = 10$).

The performance of the feature sets varies from genre to genre. RP show the best results with the biggest difference to the MARSYAS features in the Hip-hop genre, which is obvious due to its strong focus on rhythmical structures prominent in this genre. On the other hand, the MARSYAS approach performs best in the Reggae genre. In the genres labeled Ballads and Rock both approaches nearly perform the same. The differences in various genres of the approaches compensate each other and result in a nearly equal overall performance.

5.5 Experiment 3: User Study

As a music information retrieval system based upon music content analysis is intentionally designed for the use by humans, it is obvious and very important to survey users about their assessment of results such a system is providing. We thus performed a user study in order to evaluate whether the numerical performance indicators coincide with users' subjective assessments of retrieval performance. The group of participants are 11 students with an average age of 26.7 years and is balanced in terms of gender. Over 60% of the surveyed people have some basic musical education and over 80% would call themselves "interested in music". In the survey users evaluate the answers returned by Euclidean distance based retrieval on the two feature sets for query songs. 4 songs out of the genres classic, pop, rock and hip hop are selected for each of the 2 feature sets, resulting in 8 query songs. The users are presented the 8 query songs and the resulting 10 best answers, and are asked to judge them with respect to genre similarity. Not only a binary genre decision is possible, because of the assumption, that people will have very different opinions on genres. They also have the opportunity to classify a song as similar, but of different genre. During this survey a think-a-loud protocol was recorded to gain additional information about the decision finding process of the participants. That the perception of genre varies from user to user is for example confirmed through different restrictive classification behavior, as shown in the following statement.

> User:*"This is German hip-hop! Why is this track placed right in the middle of all English [note: hip hop] tracks?"*

The answer set is formed by the 10 most similar titles to the query pieces of music. The results in Table 14 reveal an interesting picture, showing highly similar performance of the two feature sets,

P_{10}	MARSYAS	RP
ambient	0.2972	0.3549
ballad	0.1556	0.1756
blues	0.2306	0.2704
classical	0.4891	0.4373
country	0.2288	0.3000
disco	0.2755	0.3676
hip-hop	0.2123	0.3632
jazz	0.3257	0.3095
metal	0.4144	0.3231
pop	0.3061	0.2336
reggae	0.3395	0.2386
rock	0.1971	0.2010
avg	0.2893	0.2979

Table 13: Precision at 10 for Collection 2.

	same genre	sounds similar	different
MARSYAS	37.17	27.01	35.82
RP	32.11	26.7	41.19

Table 14: Percentage of ratings in the categories, same genre, sounds similar and different genre, for top-10 results to 8 queries, averaged over 11 study participants.

with the MARSYAS features outperforming the RP features by 5 percent points, supporting its strong genre-based performance in experiment setting 2 on genre-based retrieval.

For almost every track the participants made their decision in about 2 or 3 seconds. Music pieces giving users no clear association to a musical genre are quickly analyzed on a different level than the musical content impression. Through figuring out the artist or the band, known influences for the artist or band, or time period of creation the users find additional information for the genre assignment process.

> User:*"This classic tune sounds very baroque! Is this a cembalo? This title is from a totally different time period and is not similar to the classical query song in my opinion!"*

This is a capability a music information retrieval system based plainly on music content analysis cannot possess. Overall the critiques of the participants for the different answer sets ranged from

> User:*"Is this meant serious? Those songs have nothing in common!"*

to

> User:*"Cool, this would be nice to automatically generate play-lists out of my private MP3 collection!"*

Results of the user survey should be seen as an encouragement to involve users in the performance evaluation process of a music information retrieval system.

6 Conclusions

This paper provided a comparison of the performance of two prominent sets of features for content-based music retrieval. Rhythm Patterns as well as the genre-oriented subset of features implemented in the MARSYAS system were extracted on two testbed collections of about 1.000 and 10.000 mp3 songs, respectively. Evaluating the performance of the two feature sets in different scenarios revealed different strengths and weaknesses of both approaches, both regarding the local stability of the extracted features over different segments of a piece of music, as well as their performance characteristics within different styles of music. Evaluation was performed both on a numeric basis, comparing recall and precision values for different answer set sizes, both for retrieving segments of a specific piece of music, as well as for genre-oriented retrieval. Last, but not least, a user study highlights the importance of incorporating users into the evaluation of any user-oriented retrieval system, particularly when the target values are highly subjective, as it is definitely and particularly the case in the domain of music retrieval.

References

[1] D. Bainbridge, C.G. Nevill-Manning, H. Witten, L.A. Smith, and R.J. McNab. Towards a digital library of popular music. In E.A. Fox and N. Rowe, editors, *Proceedings of the ACM Conference on Digital Libraries (ACMDL'99)*, pages 161–169, Berkeley, CA, August 11-14 1999. ACM. `http://www.acm.org/dl`.

[2] W.P. Birmingham, R.B. Dannenberg, G.H. Wakefield, M. Bartsch, D. Bykowski, D. Mazzoni, C. Meek, M. Mellody, and W. Rand. MUSART: Music retrieval via aural queries. In *Proceedings of the 2nd Annual Symposium on Music Information Retrieval (ISMIR 2001)*, Bloomington, ID, October 15-17 2001. `http://ismir2001.indiana.edu/papers.html`.

[3] J.S. Downie. *Annual Review of Information Science and Technology*, volume 37, chapter Music information retrieval, pages 295–340. Information Today, Medford, NJ, 2003. `http://music-ir.org/downie_mir_arist37.pdf`.

[4] H. Fast. Fluctuation strength and temporal masking patterns of amplitude-modulated broad-band noise. *Hearing Research*, 8:59–69, 1982.

[5] B. Feiten and S. Günzel. Automatic indexing of a sound database using self-organizing neural nets. *Computer Music Journal*, 18(3):53–65, 1994.

[6] J. Foote. An overview of audio information retrieval. *Multimedia Systems*, 7(1):2–10, 1999. `http://www.fxpal.com/people/foote/papers/index.htm`.

[7] J. Haitsma and T. Kalker. A highly robust audio fingerprinting system. In *Proceedings of the 3rd International Conference on Music Information Retrieval*, pages 107–115, Paris, France, October 13-17 2002. http://www.ifs.tuwien.ac.at/ifs/research/publications.html.

[8] C.-C. Liu and P.-J. Tsai. Content-based retrieval of mp3 music objects. In *Proceedings of the 10th International Conference on Information and Knowledge Management (CIKM 2001)*, pages 506 – 511, Atlanta, Georgia, 2001. ACM. http://www.acm.org/dl.

[9] Marsyas: A software framework for research in computer audition. Website. http://www.cs.princeton.edu/~gtzan/wmarsyas.html.

[10] F. Pachet and D. Cazaly. A taxonomy of musical genres. In *Proceedings of the International Conference on Content-Based Multimedia Information Access (RIAO 2000)*, Paris, France, 2000.

[11] G.P. Premkumar. Alternate distribution strategies for digital music. *Communications of the ACM*, 46(9):89–95, September 2003.

[12] A. Rauber and M. Frühwirth. Automatically analyzing and organizing music archives. In *Proceedings of the 5th European Conference on Research and Advanced Technology for Digital Libraries (ECDL 2001)*, Springer Lecture Notes in Computer Science, Darmstadt, Germany, Sept. 4-8 2001. Springer. http://www.ifs.tuwien.ac.at/ifs/research/publications.html.

[13] A. Rauber, E. Pampalk, and D. Merkl. Using psycho-acoustic models and self-organizing maps to create a hierarchical structuring of music by musical styles. In *Proceedings of the 3rd International Conference on Music Information Retrieval*, pages 71–80, Paris, France, October 13-17 2002. http://www.ifs.tuwien.ac.at/ifs/research/publications.html.

[14] A. Rauber, E. Pampalk, and D. Merkl. The SOM-enhanced JukeBox: Organization and visualization of music collections based on perceptual models. *Journal of New Music Research*, 32(2):193–210, June 2003. http://www.extenza-eps.com/extenza/loadHTML?objectIDValue=16745&type=ab%stract.

[15] Rauber, A. SOMeJB: The SOM-enhanced Jukebox. Website. http://www.ifs.tuwien.ac.at/~andi/somejb.

[16] M.R. Schröder, B.S. Atal, and J.L. Hall. Optimizing digital speech coders by exploiting masking properties of the human ear. *Journal of the Acoustical Society of America*, 66:1647–1652, 1979.

[17] C. Spevak and E. Favreau. Soundspotter - a prototype system for content-based audio retrieval. In *Proceedings of the 5. International Conference on Digital Audio Effects (DAFx-02)*, Hamburg, Germany, September 26-28 2002.

[18] G. Tzanetakis and P. Cook. Marsyas: A framework for audio analysis. *Organized Sound*, 4(30), 2000. http://www-2.cs.cmu.edu/~gtzan/work/pubs/tsap02gtzan.pdf.

[19] G. Tzanetakis and P. Cook. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302, July 2002. http://www-2.cs.cmu.edu/~gtzan/work/publications.html.

[20] E. Wold, T. Blum, D. Keislar, and J. Wheaton. Content-based classification search and retrieval of audio. *IEEE Multimedia*, 3(3):27–36, Fall 1996.

[21] H.J. Zhang and D. Zhong. A scheme for visual feature based image indexing. In *Proceedings of the IS&T/SPIE Conference on Storage and Retrieval for Image and Video Databases*, pages 36–46, San Jose, CA, February 4-10 1995. http://www.ee.columbia.edu/~dzhong/work.html.

[22] E. Zwicker and H. Fastl. *Psychoacoustics, Facts and Models*, volume 22 of *Series of Information Sciences*. Springer, Berlin, 2 edition, 1999.