Music Analysis and Retrieval Systems for Audio Signals

George Tzanetakis
Computer Science Department, Faculty of Engineering, University of Victoria, P.O. Box 3055 STNCSC, Victoria, BC V8W 3P6, Canada. E-mail: gtzan@cs.uvic.ca

Perry Cook
Computer Science and Music Department, Princeton University, Princeton, NJ 08544. E-mail: prc@cs.princeton.edu

The constantly increasing amount of audio available in digital form necessitates the development of software systems for analyzing and retrieving digital audio. In this work, we describe our efforts in developing such systems. More specifically, we describe the design philosophy behind our approach, the specific problems we try to solve, and how we evaluate the performance of our algorithms. Automatic music analysis and retrieval of non-speech digital audio is a relatively new field, and the existing techniques are far from perfect. To improve the performance of the developed techniques, two main techniques are used: (1) integration of information from multiple analysis and retrieval algorithms and (2) the use of graphical user interfaces that enable the user to provide feedback to the design, development, and evaluation of the algorithms. All the developed algorithms and user interfaces are integrated under MARSYAS, a software framework for research in computer audition.

Introduction

Although the manipulation and storage of sound using computers is not new, only recently have developments in compression technology, network bandwidth, and storage capacity made possible, for the average user, the creation of large collections of digital audio and especially music. Currently, the main way users interact with these increasingly large audio collections is by conventional browsing tools that employ only the file name information and occasionally some manually annotated metadata. Another important characteristic of the developed systems is that they work directly on raw audio signals without attempting to perform polyphonic transcription. More sophisticated analysis and retrieval software systems for music are required to handle the increasing size and complexity of these collections. It is very likely that in the near future most of the recorded music in human history will be available on the Web, and most of the recording companies are currently investigating business models of how this is going to be accomplished.

This article is an overview of research in music information retrieval (MIR) for audio signals conducted in the last four years in the Computer Science Department of Princeton University. A main goal of this project is to take advantage of the human user during the design, development, and evaluation of the proposed software systems. The goal of this article is to provide a broad overview of the design, development, and evaluation of music analysis and retrieval systems for audio signals by summarizing the results of several conference and journal publications. Additional technical details can be found in the references.

Overview—Terminology

Feature Extraction is the process of computing a numerical representation that can be used to characterize a segment of audio. This numerical representation is called the *feature vector*, and is subsequently used as the fundamental building block of various types of analysis algorithms. This vector typically has a fixed dimension and therefore can be thought as a point in a high dimensional *feature space*. For music—and audio in general—the computed features are typically calculated based on some Time-Frequency Decomposition Technique. Some examples are the Short Time Fourier Transform (STFT), the Discrete Wavelet Transform (DWT), and Linear Prediction Coefficients (LPC). All these signal processing techniques calculate how the energy of the signal is distributed in time and frequency. When using feature vectors to represent music audio files two main approaches are used. In the first approach the audio file is broken into small segments in time and a feature vector is computed for each segment. The resulting representation is a time series of feature vectors, which can be thought of as a trajectory of points in the feature space. In the second

 ● Published online 25 May 2004 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/asi.20060

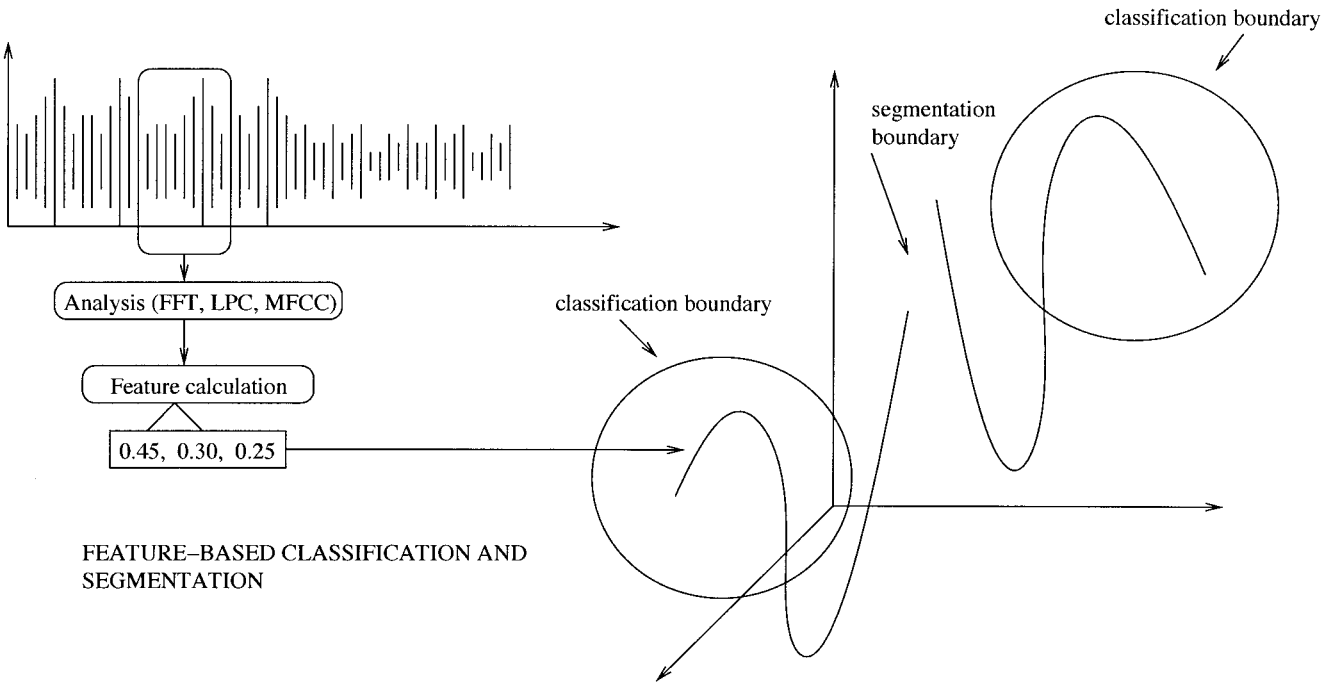

FIG. 1. Feature extraction, segmentation, and classification.

approach a single feature vector that summarizes information for the whole file is used.

Beat extraction and analysis is a special case of feature extraction where the extracted features attempt to represent the rhythmic structure of the music. Although typically an estimate of the main beat of the music and of its strength are computed, more detailed information can be used for MIR.

Content-based *similarity retrieval* is arguably the most basic analysis that can be performed using the feature vector representation. In this technique, the user provides a sample music audio file as a query and the system returns a list of "similar" music audio files ranked by their similarity. Using the single feature vector approach, similarity retrieval can be performed by ranking the returned files based on their distance from the query feature vector. Using the feature space analogy the query can be thought of as a point, and the points that are closest to it are the ones that have similar content. More complicated techniques outside the scope of this article are required for similarity retrieval using the feature vector trajectory approach.

Another technique that is based on feature extraction is *classification*, where a music audio file or segment is automatically assigned to set of predefined class labels. Examples of classification in MIR are vocal vs. instrumental music, musical instrument classification, and genre classification. Classification can be thought of as a partitioning of the feature space into regions such that all points in a region belong to the same class. The shape and number of those regions for each class depend on the specific classification method used.

In many cases a variety of music "textures" are used throughout a music work. For example, a piano concerto might contain parts where only the piano is playing, parts where the orchestra and the piano are playing , and parts where only the woodwind section of the orchestra is playing. A rock piece might consist of an instrumental introduction, chorus, and a guitar solo. Similarly a jazz piece might contain improvised solos by individual instruments in addition to the chorus of the song where all the instruments are playing. *Segmentation* refers to the process of detecting where in time those music "texture" changes occur. One way of doing this is by detecting abrupt changes in the trajectory of feature vectors. Figure 1 is a schematic overview of feature extraction, segmentation, and classification.

In *audio thumbnailing* or music summarization the goal is to create a short summary version that captures the essential characteristics of a music audio file. Audio thumbnailing is typically used for the short presentation of many audio files as, for example, in browsing and similarity retrieval ranked lists. Another technique used for browsing and retrieval lists is *audio visualization*, where 2D or 3D graphics are used to represent music audio files or collections.

Polyphonic transcription is the process of converting a raw music audio file to a high level symbolic representation typically consisting of notes and their durations. The most common format for this symbolic representation is MIDI. This high level representation can then be used for symbolic MIR. Unfortunately, a polyphonic transcription system that works robustly with arbitrary raw audio signals has not yet been developed.

The term *computer audition* is used in this article to refer to any technique that tries to extract information from raw audio signals. Other terms for this technique used in the

published literature are Computational Auditory Scene Analysis (CASA) and Machine Listening.

Related Work

Foote (1999) offers an early overview of research in AIR including speech and symbolic MIR. One of the earliest published Audio Information Retrieval (AIR) systems is described in Wold, Blum, Keislar, and Wheaton (1996). In this system spectral features and statistical pattern recognition techniques are used to classify and retrieve from a database of short isolated sounds—mainly musical instruments and sound effects. The collection of isolated sounds used in Wold et al. has been used as a testbed to compare other methods (Foote, 1997; Welsh, Borisov, Hill, von Behren, & Woo, 1999; Li, 2000; Li & Khokar, 2000).

Audio feature extraction for AIR has been based on techniques developed for speech recognition. Probably the most common form of analysis front-end for feature extraction is the short time Fourier Transform (STFT) (Rabiner & Gold, 1975). Mel-frequency cepstral coefficients (MFCC) are a more perceptually accurate representation typically used in speech recognition applications (Davis & Mermelstein, 1980). Another technique originating from speech research is linear prediction coefficients (LPC) (Makhoul, 1975). Audio features can also be directly calculated from compressed audio data like MP3 files (Pye, 2000; Tzanetakis & Cook, 2000b).

Scheirer and Slaney (1997) describe the construction and evaluation of a classifier for discriminating music from speech. The classification of music instrument sounds has been explored in Brown (1999), Fujinaga (2000), Martin (1999), and Eronen and Klapuri (2000). Tzanetakis and Cook (2001) describe automatic musical genre classification. Techniques for automatic audio segmentation are described in Aucouturier and Sandler (2001), Foote (2000), Kimber and Wilcox (1996), and Tzanetakis and Cook (2000a). A classification and segmentation system for audio signals from movies or TV programs is described in Zhang and Kuo (2001). Logan (2000) describes a method for automatic music summarization (audio thumbnailing) and describe user experiments that show that it performs better than random summarization.

Methods for automatic beat and tempo detection are presented in Foote and Uchihashi (2001), Scheirer (1998), Goto and Muraoka (1998), and Smith (1999). The use of a beat histogram for genre classification is described in Tzanetakis and Cook (2001). Perceptually motivated music listening systems are described in detail in Scheirer (2000), Ellis (1996), and Smaragdis (2001) deal with more general problems in computational auditory scene analysis.

Design Guidelines

The field of Audio Information Retrieval (AIR) is relatively new and still developing. As a result, most of the existing algorithms are not perfect. Because the goal of this project is to build robust prototype music analysis and retrieval systems, several ways of dealing with the imperfections of the existing techniques are used. To handle the uncertainty and noise of the calculated features, statistical pattern recognition and machine learning techniques are used. Integrating the results of different algorithms is another way of improving their performance. For example, when classifying a piano concerto, the classification results can be improved by first automatically detecting the piano and orchestra segments, and then classifying those segments separately. In similarity retrieval the results can be improved by restricting the returned results to files of a particular musical genre that is automatically detected using classification algorithms. The order in which the algorithms are integrated can be either predefined or controlled by the user. The user can interact with the system using traditional and novel graphical user interfaces. Having all the algorithms integrated under a common interface allows the user to edit imperfect results and to provide feedback to the algorithms. In addition, the use of visualization takes advantage of the strong pattern recognition abilities of the human visual system to enhance the results. For example, the AABA (where A and B are different sections and A repeats) structure of a particular song can be clearly displayed using appropriate automatic audio visualization techniques, although it might be difficult to detect automatically. Finally, the graphical user interfaces can be used in user studies for evaluating the developed algorithms.

Evaluation Results

The performance of a MIR system is defined by what human users expect from it rather than some easy to calculate, objective measure. To evaluate the performance of the developed MIR systems, several user studies have been conducted. For each of these user studies the main goal has been to answer the following questions:

1. What is the average human performance for the task we are trying to automate?
2. How consistent are the human responses between subjects?
3. How do human users judge the performance of the automatic system?
4. Does the existence of an automatic system influence the user responses?

In addition to these questions typically the manual results of the user study are used to further tune the parameters of the automatic algorithms. For classification purposes a supervised learning paradigm is used. In this approach the goal is to derive a statistical model of the distribution of feature vectors for a particular class. This is achieved through the use of a training set of labeled feature vectors extracted from a large collection of an audio file representative of that particular class. More details regarding this process can be found in standard Pattern Recognition text-

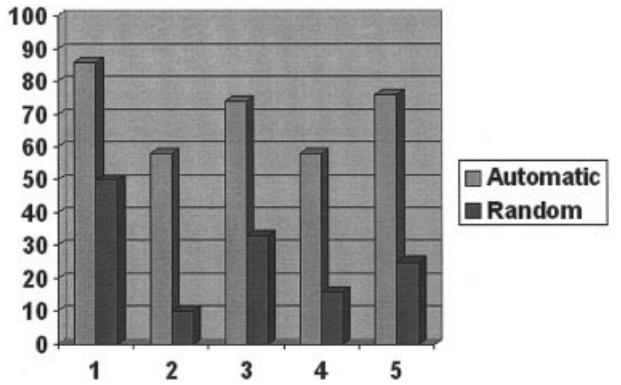

FIG. 2. Automatic accuracy classification results.

books like Duda, Hart, and Stork (2000), Fukunaga (1972), and Schalkoff (1992).

Figure 2 summarizes the results of various types of automatic classification for audio signals. For each data set the automatic classification accuracy is compared with chance classification. To calculate these results, a tenfold evaluation paradigm was used. In this paradigm, many different random partitions of the data to training and testing data are used and the classification results averaged 100 iterations. This way the classification results are representative of how the algorithm would perform with real world data provided that the labeled data set we use is representative of the class in which we are interested. For each data set the following classes were used:

1. MusicSpeech: Music, Speech.
2. Voices: Male Voice, Female Voice, Sports announcing.
3. Genres: Classical, Country, Disco, HipHop, Jazz, Rock, Blues, Reggae, Pop, Metal.
4. Jazz: Bigband, Cool, Fusion, Piano, Quartet, Swing.
5. Classical: Choir, Orchestra, Piano, String Quartet.

The performance of humans in classifying musical genre has been investigated in Perrot and Gjerdingen (1999). Using a ten-way forced choice paradigm, college students were able to accurately judge (53% correct) after listening to only 250-millisecond samples and (70% correct) after listening to 3 seconds (chance would be 10%). Although direct comparison of these results with the automatic musical genre classification is not possible due to different genres and datasets, it is clear that the automatic performance is not far from the human performance. It should be noted that perfect performance in genre classification is probably impossible to achieve because of the fuzzy nature of genre definitions. Figure 3 shows the confusion matrix for the automatic genre classification. The columns correspond to the actual genre and the rows to the predicted genre. For example, the cell of row 6, column 3, with value 16, means that 16 percent of Country music (column 3) was misclassified as Rock music (column 7). The percentages of correct classifications lie in the diagonal of the confusion matrix.

To evaluate the automatic segmentation methodology proposed in Tzanetakis and Cook (1999), two user studies were conducted. In these experiments the subjects were asked to segment ten 1-minute-long sound files. A variety of textures and styles are represented. In particular there were two excerpts from radio broadcasts with speech and music, three classical music excerpts, two jazz excerpts, one funk excerpt, and two rock-pop music excerpts. Twenty subjects were used for two user studies. In the first study a standard sound editor was used to segment the files whereas in the second study the users edited an automatically provided segmentation. The subjects were asked to segment each sound file in three ways. The first way, which we call "free," is breaking up the file into any number of segments. The second and third way constrain the users to a specific budget of total segments 8 ± 2 and 4 ± 1. Without going into details, there was significant agreement between subjects regarding segmentation marks. The percentage of total segmentation marks that more than 10 of the 20 subjects agreed upon is 73% (in the case of free segmentation). Furthermore, 87% of these segmentation marks were automatically detected by our algorithm. Finally, in the second user study where an automatically suggested segmentation was used, 70% of the automatically suggested segmentation marks were retained, and the automatic segmentation did not bias the segmentation results. More details about these user

	Classical	Country	Disco	Hiphop	Jazz	Rock	Blues	Reggae	Pop	Metal
Classical	73	0	0	0	6	2	0	0	0	0
Country	0	43	1	0	1	6	2	4	3	1
Disco	0	6	43	10	0	4	9	3	3	2
Hiphop	0	4	9	49	0	3	2	17	10	2
Jazz	21	5	1	0	71	6	5	0	2	3
Rock	4	16	6	1	8	41	11	5	11	17
Blues	2	18	2	1	7	7	61	5	1	2
Reggae	0	2	12	30	2	13	6	63	6	0
Pop	0	3	25	8	3	7	0	3	64	2
Metal	0	3	1	1	1	11	4	0	0	71

FIG. 3. Automatic genre confusion matrix.

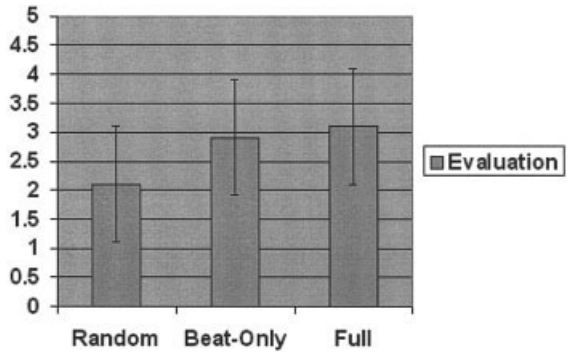

FIG. 4. Evaluation of rock-song music retrieval.

studies can be found in Tzanetakis and Cook (1999, 2000a). In addition to segmenting the sound files, users had to provide a thumbnail and text annotation for each segment. Most of the thumbnails were around the segmentation boundaries, suggesting that thumbnails can be created based on the automatic segmentation. User experiments to evaluate this method for audio thumbnailing are planned for the future. A preliminary examination of the text annotations showed that about 60% of all words fit into three categories: sound source descriptions (like saxophone solo or orchestra), structural music theoretic description (like introduction, chorus), and basic acoustic parameters (like loud or soft). Creating a system that automatically suggests text annotations is planned for the future.

A user study of content-based music information retrieval of rock songs was conducted. The large size (1000 files) of the collection made the calculation of recall difficult so only precision was examined. The relative uniformity of the collection (only rock songs) made the retrieval task more challenging. Because the 30-second snippets used for evaluation did not have many texture changes, the single vector approach for representing audio files was used. Seven subjects were asked to give a relevance judgement from 1 (worse) to 5 (best) for each file returned. There were 12 queries, five matches returned for each query and three algorithms (random, beat-only, beat and texture) giving a total of $7 \times 5 \times 12 \times 3 = 1{,}260$ collected data points. The beat detection was performed using the algorithm described in Scheirer (1998).

Figure 4 shows the mean and standard deviation of the three retrieval methods. The standard deviation is due to the different nature of the queries and subject differences and is about the same for all algorithms. Although it is clear that the system performs better than random and that the full approach is slightly better than using only beat detection, more work needs to be done to improve the scores.

We are currently conducting a user experiment to investigate how humans perceive music beat strength. Each subject has to rate 50 segments from a variety of musical styles into five beat strength groups (weak, medium-weak, medium, medium-strong, strong). The order of presentation is randomized to avoid any order effects. Preliminary results from the ten subjects who have completed the study so far indicate that there is large agreement between subjects regarding music beat strength. The results of this study will be used to evaluate and improve the beat histogram calculation method described in Tzanetakis and Cook (2001).

Graphical User Interfaces

Several content aware graphical user interfaces specifically designed for browsing and interacting with large audio collections have been developed. These are:

1. *Augmented sound editor*. This interface offers the same functionality as a traditional sound editor (waveform and spectogram displays, mouse selection, playback status bar, zooming, etc). In addition to these typical features, a sound file can be automatically segmented with each region displayed with a different color. For browsing the user can move by regions and each region can be annotated with text. Different classification schemes can be applied to each segmented region or to arbitrary selections. Retrieval and audio thumbnailing are also supported.
2. *Timbregrams* are a collection-dependent way of visualizing sounds. Each timbregram consists of a series of vertical color stripes where each stripe corresponds to a feature vector. The timbregram reveals sounds that are similar by color and time periodicity. For example, the ABA structure of a song will be reflected in an ABA structure in color. Because the mapping of the feature vectors to color depends on the specific data collection used, multiple visualizations with different meanings for the same file are possible.
3. *Timbrespace* is another collection-dependent way of visualizing collections of sounds for browsing. Each sound (feature vector) is represented as a single point in a 3D space. The mapping of sounds to points is performed automatically and is dependent on the specific data collection used. The Timbrespace reveals similarity of sounds based on their proximity in the space. Figure 5 shows a Timbrespace.
4. *GenreGram* is a dynamic real-time 3D graphics audio display targeted toward radio signals. The live radio signal is analyzed in real time and is classified into 11 categories: Male Voice, Female Voice, Sports Announcing, Classical, Country, Disco, Fuzak, HipHop, Jazz, Rock, and Static. For each of these categories a confidence measure, ranging from 0.0 to 1.0, is calculated and used to move up or down rotating cylinders corresponding to each category. Each cylinder is texture-mapped with a representative image of its corresponding category. The movement is also weighted by a separate classification decision of Music vs. Speech. In addition to being a nice demonstration for real-time automatic audio classification, the GenreGram gives valuable feedback both to the user and algorithm designer. Different classification decisions and their relative strengths are combined visually, revealing correlations and classification patterns. Since the boundaries between musical genres are fuzzy, a display like this is more informative

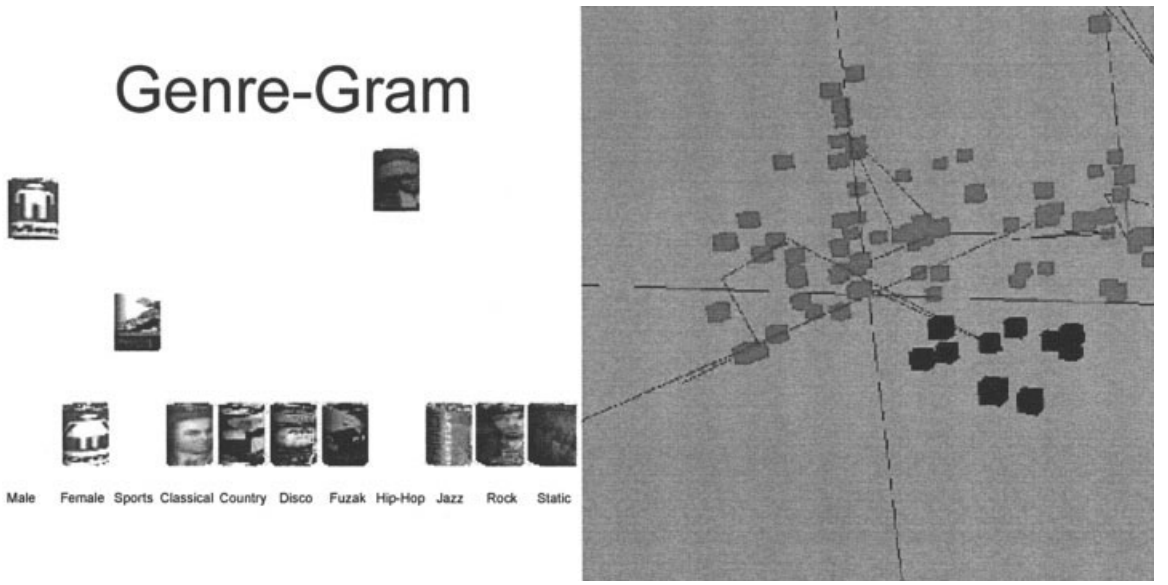

FIG. 5. Timbrespace and GenreGram.

than a single all or nothing classification decision. Figure 5 shows a static snapshot of a GenreGram.

Implementation

All the developed algorithms are integrated under MARSYAS, a software framework for research in computer audition. The framework follows a client–server architecture where the server, written in C++, performs all the numerically intensive computations and the client, written in JAVA, consists of the graphical user interface. Multiple clients, possibly in different computers and operating systems, can be connected simultaneously to the server. MARSYAS is mainly developed under Linux but also works under Solaris, Irix, and Windows systems. The software is freely distributed under the GNU public license and can be obtained from

Future Work

We are currently expanding the genre classification hierarchy to more genres, and we are investigating other semantic descriptions (instrumentation, emotion) as possible classification categories. Another interesting direction for future research is the identification of which instruments are playing in a sound mixture without necessarily separating them. Of course, more user experiments in musical beat strength perception, similarity, retrieval, and classification are planned for the future. Finally, a full prototype music information retrieval system that will be available for research purposes is planned for the future.

Summary

A series of systems for music analysis and retrieval of audio signals were presented. These systems work directly on raw audio files without attempting to perform polyphonic transcription. Novel content-aware graphical user interfaces are used to help the design, development, and evaluation of these systems. Most of the described systems are integrated under MARSYAS, a free software framework for research in computer audition which can be downloaded from http://www.cs.princeton.edu/~gtzan/marsyas.html

References

Aucouturier, J.-J., & Sandler, M. (2001). Segmentation of musical signals using Hidden Markon Models. Paper presented at the 110 Audio Engineering Society Convention, Amsterdam, The Netherlands.

Brown, J. (1999). Computer identification of musical instruments. Journal of the Acoustical Society of America, 105(3), 1933–1941.

Davis, S., & Mermelstein, P. (1980). Experiments in syllable-based recognition of continuous speech. IEEE Transactions on Acoustics, Speech and Signal Processing, 28, 357–366.

Duda, R., Hart, P., & Stork, D. (2000). Pattern classification. New York: Wiley.

Ellis, D. (1996). Prediction-driven computational auditory scene analysis. Unpublished doctoral dissertation, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA. Retrieved April 2004, from http://sound.media.mit.edu/~dpwe

Eronen, A., & Klapuri, A. (2000). Musical instrument recognition using cepstral features and temporal features. Paper presented at the International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 753–757), Istanbul, Turkey.

Foote, J. (1997). Content-based retrieval of music and audio. Multimedia Storage and Archiving Systems, II, 138–147.

Foote, J. (1999) An overview of audio information retrieval. ACM Multimedia Systems, 7, 2–10.

Foote, J. (2000). Automatic audio segmentation using a measure of audio novelty. Paper presented at the International Conference on Multimedia and Expo, IEEE (pp. 452–455).

Foote, J., & Uchihashi, S. (2001). The Beat Spectrum: A new approach to rhythmic analysis. Paper presented at the International Conference on Multimedia and Expo, IEEE.

Fujinaga, I. (2000). Realtime recognition of orchestral instruments. Paper presented at the International Computer Music Conference (ICMC) (pp. 141–143).

Fukunaga, K. (1972). Introduction to Statistical Pattern Recognition, 175–177. New York: Academic Press.

Goto, M., & Muraoka, Y. (1998). Music understanding at the beat level: Real-time beat tracking of audio signals. In D. Rosenthal & H. Okuno (Eds.), Computational Auditory Scene Analysis (pp. 156–176). Mahwah, NJ: Lawrence Erlbaum Associates.

Kimber, D., & Wilcox, L. (1996). Acoustic segmentation for audio browsers. Paper presented at the Interface Conference, Sydney, Australia.

Li, G., & Khokar, A. (2000). Content-based indexing and retrieval of audio data using wavelets. Proceedings of the International Conference on Multimedia and Expo (pp. 885–888). IEEE.

Li, S. (2000). Content-based classification and retrieval of audio using the nearest feature line method. IEEE Transactions on Speech and Audio Processing, 8(5), 619–625.

Logan, B. (2000). Music summarization using key phrases. Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 749–752).

Makhoul, J. (1975). Linear prediction: A tutorial overview. Proceedings of the IEEE, 63, 561–580.

Martin, K. (1999). Sound source recognition: A theory and computational model. Unpublished doctoral dissertation, Massachusetts Institute of Technology, Cambridge, MA. Retrieved April 2004, from http://sound.media.mit.edu/~kdm

Perrot, D., & Gjerdingen, R. (1999). Scanning the dial: An exploration of factors in identification of musical style. Paper presented at the 1999 Society for Music Perception and Cognition (abstract) (p. 88).

Pye, D. (2000). Content-based methods for the managements of digital music. Proceedings of International Conference on Acoustics, Speech and Audio Processing (ICASSP) (pp. 2437–2440).

Rabiner, L., & Gold, B. (1975). Theory and applications of digital signal processing. Englewood Cliffs, NJ: Prentice Hall.

Schalkoff, R. (1992). Pattern recognition: Statistical, structural and neural approaches. New York: Wiley.

Scheirer, E. (1998). Tempo and beat analysis of acoustic musical signals. Journal of the Acoustical Society of America, 103(1), 588–601.

Scheirer, E. (2000). Music-listening systems. Unpublished doctoral dissertation, Massachusetts Institute of Technology, Cambridge, MA.

Scheirer, E., & Slaney, M. (1997). Construction and evaluation of a robust multifeature speech/music discriminator. IEEE Transactions on Acoustics, Speech and Signal Processing, 1331–1334.

Smaragdis, P. (2001). Redundancy reduction for computational audition, a unifying approach. PhD thesis, MIT Media Laboratory, Massachusetts Institute of Technology, Cambridge, MA.

Smith, L. (1999). A multiresolution time-frequency analysis and interpretation of musical rhythm. Unpublished doctoral dissertation, University of Western Australia, Perth, Australia.

Tzanetakis, G., & Cook, P. (1999). Multifeature audio segmentation for browsing and annotation. Proceedings of the Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA) (pp. 103–107), New Paltz, NY. New York: IEEE.

Tzanetakis, G., & Cook, P. (2000a). Experiments in computer-assisted annotation of audio. Proceedings of the International Conference on Auditory Display (ICAD) (pp. 111–116).

Tzanetakis, G., & Cook, P. (2000b). Sound analysis using MPEG compressed audio. Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 761–764), Istanbul, Turkey.

Tzanetakis, G., & Cook, P. (2001). Automatic musical genre classification. Proceedings of the International Symposium on Music Information Retrieval (ISMIR) (pp. 205–211).

Welsh, M., Borisov, N., Hill, J., von Behren, R., & Woo, A. (1999). Querying large collections of music for similarity (Tech. Rep. UCB/CSD00-1096). University of California, Berkeley Computer Science Division.

Wold, E., Blum, T., Keislar, D., & Wheaton, J. (1996). Content-based classification, search and retrieval of audio. IEEE Multimedia, 3(2) 27–36.

Zhang, T., & Kuo, J. (2001). Audio content analysis for online audiovisual data segmentation and classification. Transactions on Speech and Audio Processing, 9(4), 441–457.