Experiments in computer-assisted annotation of audio

George Tzanetakis	Perry R. Cook
Computer Science Dept.	Computer Science and Music Dept.
Princeton University	Princeton University
35 Olden St.	35 Olden St.
Princeton, NJ 08544 USA	Princeton, NJ 08544 USA
+1 609 258 4951	+1 609 258 4951
gtzan@cs.princeton.edu	prc@cs.princeton.edu

ABSTRACT

Advances in digital storage technology and the wide use of digital audio compression standards like MPEG have made possible the creation of large archives of audio material. In order to work efficiently with these large archives much more structure than what is currently available is needed. The creation of the necessary text indices is difficult to fully automate. However, significant amounts of user time can be saved by having the computer assist the user during the annotation process.

In this paper, we describe a prototype audio browsing tool that was used to perform user experiments in semi-automatic audio segmentation annotation. In addition to the typical sound-editor functionality the system can automatically suggest time lines that the user can edit and annotate. We examine the effect that this automatically suggested segmentation has on the user decisions as well as timing information about segmentation and annotation. Finally we discuss thumbnailing and semantic labeling of annotated audio.

Keywords

Audio information retrieval, audio segmentation, semi-automatic audio browsing, audio thumbnailing, user experiments

INTRODUCTION

Advances in digital storage technology and the wide use of digital audio compression standards like MPEG have made possible the creation of large archives of audio material. In order to work efficiently with these large archives much more structure than what is currently available is needed. One way to approach the problem is through the creation of text indices and the use of traditional information retrieval techniques [1] familiar from the popular Web search engines. An excellent overview of the current state of the art in audio information retrieval is given in [2].

Work in audio annotation has mostly centered on speech and the creation of indices using speech recognition techniques. In the Informedia project [3] a combination of speech recognition, image analysis and keyword searching techniques is used to index a terrabyte video archive. SoundFisher is a content-aware sound browser that has been developed by Musclefish [4, 5]. Users can search for and retrieve sounds by perceptual and acoustical features, can specify classes based on these features, and can ask the engine to retrieve similar or dissimilar sounds.

This paper focuses on the task of annotating audio data and especially music. An example would be structuring hours of archived radio broadcasts for audio information retrieval. Annotation of simple cases like musical instruments or music vs speech can be performed automatically using current classification systems. Based on these techniques, a completely automatic annotation system for audio could be envisioned. Although not impossible in theory, there are two problems with such an approach. The first is that current systems are far from perfect and, therefore, annotation errors are inevitable. This problem has to do with the current state of the art, so it is possible that in the future it will be solved. There is a second problem, however, that is more subtle and not so easy to address. Audio, and especially music, is heard and described differently by each listener. There are, however, attributes of audio that most listeners will agree upon, like the general structure of the piece, the style, etc. Ideally a system for annotation should automatically extract as much information as it can and then let the user edit it.

This leads to a semi-automatic approach that combines both manual and automatic annotation into a flexible, practical user interface. A prototype of such a semi-automatic audio annotation tool has been created mainly for the purpose of collecting experimental data about what humans do when asked to segment and annotate audio. The main addition to the typical sound-editor functionality is the ability of the system to automatically suggest time lines that the user can edit and annotate. We examine the effect that this automatically suggested segmentation has on the user decisions. In addition timing information about the task of segmenting and annotating audio has been collected. Some analysis of user thumbnailing and the semantic labeling of annotated audio was performed.

SEGMENTATION

In this work segmentation refers to the process of breaking audio into regions in time based on what could be called "texture" of sound. Some examples are a piano entrance after the orchestra in a concerto, a rock guitar solo, a change of speaker etc. There are no assumptions about the type of audio and no statistical class model of the data is made. For segmentation we follow the methodology described in [6]. First a time series of multi-dimensional feature vectors is computed. Then a Mahalanobis distance signal is calculated between successive frames of sound. The peaks of the derivative of this distance signal correspond to texture changes and are used to automatically determine segmentation boundaries. In order to capture most of the desired boundaries the algorithm is parameterized to oversegment into more regions than the desired number (up to 16 segments for a 1 minute soundfile). This is called the *best effort* condition for the peak picking heuristic.

ANNOTATION TOOL

The annotation tool used for the user experiments consists of a graphical user interface looking like a typical sound-editor. Using a waveform display, arbitrary regions for playback can be selected and annotated time lines can be loaded and saved. Each segmented region is colored differently and the user can move forward and backward through those regions. In addition to the typical sound editor functionality the system can automatically segment the audio to suggest a time line. The resulting regions can then be edited by adding/deleting boundaries until the desired segmentation is reached. Finally, the plug-in architecture of the system easily allows the use of segmentation results from other analysis tools such as a speech recognition system.

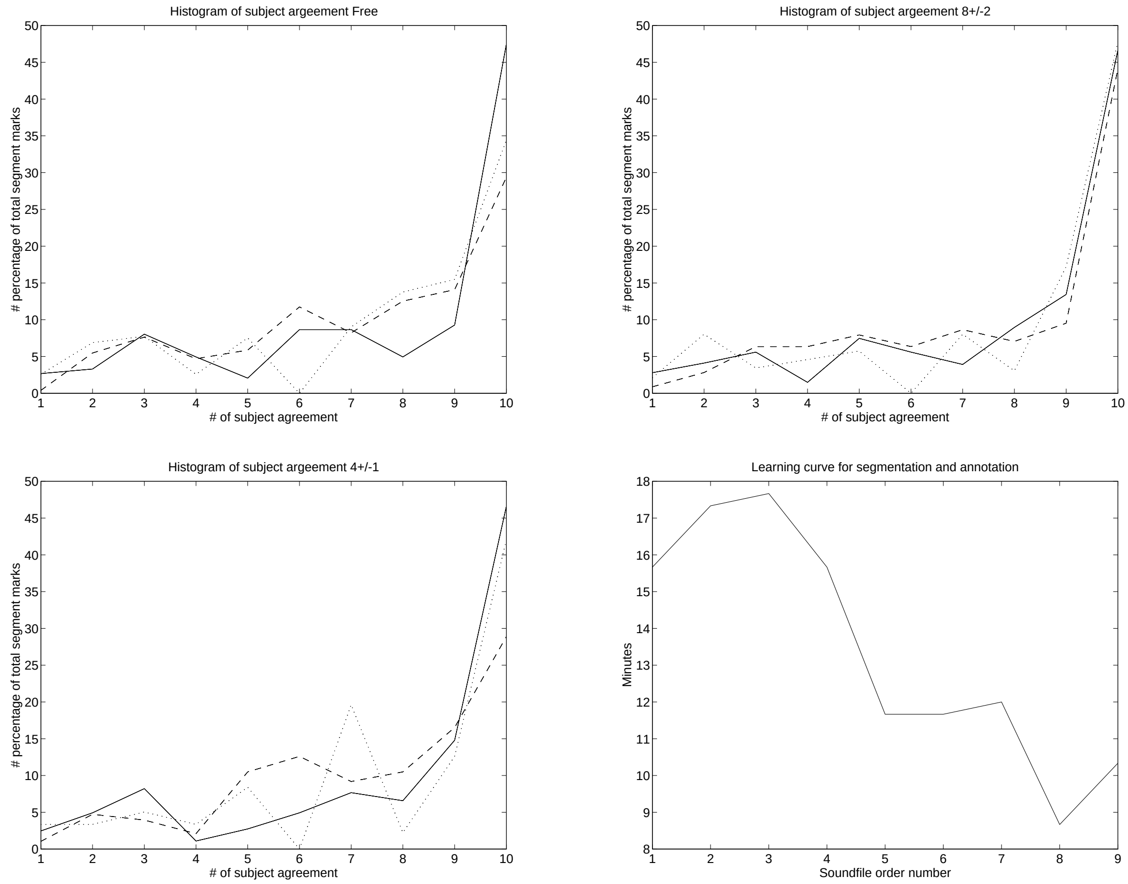

Figure 1: A,B,C: histograms of subject agreement and D: learning curve for mean segment completion time

USER EXPERIMENTS

In [6] a pilot user study was conducted trying to answer two main questions: are humans consistent when they segment audio and if their performance can be approximated by automatic means. In that study subjects were allowed to use the sound editor of their choice, therefore no timing information was collected. In this work a new set of user experiments was performed to re-confirm the results of [6] and answer some additional questions. The main question we tried to answer was if providing an automatic segmentation as a basis would bias the resulting segmentation. In addition information about the time required to segment and annotate audio was collected. It is our belief that the use of automatically segmented time-lines can greatly accelerate the segmentation and annotation process. Further, significant parts of this process can also be automated.

The data used consists of 10 sound files about 1 minute long. A variety of styles and textures are represented. Ten subjects were asked to segment each sound file in 3 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 4 ± 1 and 8 ± 2. In [6] any standard audio editing tool could be used whereas in this study the previously described annotation tool was used.

In figure 1 histograms of subject agreement are shown. To calculate the histogram all the segmentation marks were collected and partitioned into bins of agreement in the following manner: all the segment marks within ± 0.5 seconds were considered as corresponding to the same segment boundary. This value was calculated based on the differences between the exact location of the segment boundary between subjects and was confirmed by listening to the corresponding transitions. Since at most all the ten subjects can contribute marks within this neighborhood of ± 0.5 seconds the maximum number of subject agreement is 10. In the figure the different lines show the histogram of the experiments in [6] (old), the results of this study (new) and the total histogram of both studies (total). The total histogram is calculated by considering the 20 subjects and dividing by two to normalize (notice that this is different than taking the average of the two histograms since this is done on a boundary basis). Therefore if in the two studies the boundaries were changed indicating bias because of the automatically suggested segmentation, the shape of the total histogram would drastically change. The histograms show that there is a large percentage of agreement between subjects and that the segmentation results are not affected by the provision of an automatically suggested segmentation time line. Finally the fact that the the old, new and total histograms have about the same shape in subfigures A,B,C suggests that constraining the user to a specific budget of segments does not affect significantly their agreement.

As a metric of subject agreement we define the percentage of the total segmentation marks that more than 5 of the 10 subjects agreed upon. This number can be calculated by integrating the histograms from 6 to 10. For the experiments in [6] this metric gives 79%, for this study 76% and, for the combined total 73%. In [6] (old), 87% of the segments that half the subjects agreed upon were in the set of the best effort automatic segmentation. Moreover for this study (new) 70% of the total segment marks by all subjects were retained from the best effort automatically suggested segmentation. All these numbers are for the case of free segmentation.

The mean and standard deviation of the time it took to complete (segment and annotate) a soundfile with duration of about 1 minute was 13 ± 4 minutes. This result was calculated using only the free segmentation timing information because the other cases are much faster due to the familiarity with the soundfile and the reusability of segmentation information from the free case. Subfigure D of figure 1 shows the average time per soundfile in order of processing. The order of processing was random therefore the figure indicates there is a significant learning curve for the task. This happens despite the fact that an initial soundfile that was not timed was used to familiarize the users with the interface. Therefore the actual mean time is probably lower (about 10 minutes) for an experienced user.

THUMBNAILING

An additional component of the annotation tasks of [6] was that of "thumbnailing." After doing the free, 8 ± 2, and 4 ± 1 segmenting tasks, subjects were instructed to note the begin and end times of a two second thumbnail segment of audio that best represented each section of their free segmentation sections.

Inspection of the 545 total user thumbnail selections revealed that 62% of them were chosen to be the first two seconds of a segment, and 92% of them were a selection of two seconds within the first five seconds of the segment. This implies that a machine algorithm which can perform segmentation could also do a reasonable job of matching human performance on thumbnailing. By simply using the first five seconds of each segment as the thumbnail, and combining with the results of the best-effort machine segmentation (87% match with human segments), a set containing 80% "correct" thumbnails could be automatically constructed.

ANNOTATION

Some work in verbal cues for sound retrieval [7] has shown that humans tend to describe isolated sounds by source type (what it is), situation (how it is made), and onomatopoeia (sounds like). Text labeling of segments of a continuous sound stream might be expected to introduce different description types, however. In this paper, a preliminary investigation of semantic annotations of sound segments was conducted. While doing the segmentation tasks, subjects were instructed to "write a short (2-8 words) description of the section..." Annotations from the free segmentations were inspected by sorting the words by frequency of occurrence.

The average annotation length was 4 words, resulting in a total of 2200 meaningful (words like *of, and, etc.* were removed) words, and 620 unique words. Of these, only 100 words occur 5 or more times and these represent 64% of the total word count. Of these "top 100" words, 37 are literal descriptions of the dominant source of sound (piano, female, strings, horns, guitar, synthesizer, etc.), and these make up almost 25% of the total words used.

The next most popular word type could be classed as music-theoretical structural descriptions (melody, verse, sequence, tune, break, head, phrase, etc.) 29 of the top 100 words were of this type, and they represent 23% of the total words used. This is striking because only 5 of the 20 subjects could be considered professional composers/musicians and structural descriptions (not always correctly) were used by many of the non-musicians.

Another significant category of words used corresponded to basic acoustic parameters (soft, loud, slow, fast, low, high, build, crescendo, increase, etc.). Most of such parameters are easy to calculate from the signal. 12 of these words represented about 10% of the total words used.

These preliminary findings indicate that with suitable algorithms determining basic acoustical parameters (mostly possible today), the perceptually dominant sound source type (somewhat possible today), and music-theoretical structural aspects of sound segments (much algorithmic work still to be done), machine labeling of a fair number of segments (60%) would be possible.

IMPLEMENTATION

The system has been implemented using MARSYAS [8] an object oriented framework for building audio analysis applications. A client-server architecture is used. The graphical user interface (written in JAVA) acts as a client to the server engine (written in C++) where all the signal processing is done. The system runs on Solaris, SGI, Linux and Windows (95,98 and NT) platforms.

FUTURE WORK

In the future we plan to collect more data on the time required to segment audio. Empirical evidence is required that the automatic segmentation reduces user time required. Further tests in thumbnailing will need to be devised to determine the salience of the human and machine selected thumbnails, and to determine their usefulness. For example, can thumbnails cause a speedup in location/indexing, or can thumbnails be concatenated or otherwise combined to construct a useful "caricature" of a long audio selection? We plan to make more detailed analyses of the text annotations. Finally the developed graphical user interface allows the collection of many user statistics like number of edit operations, detailed timing information,etc. that we plan to investigate further.

SUMMARY

A series of user experiments in computer assisted annotation of audio were performed. The results show that using automatic segmentation to assist the user does not bias the resulting segments. The average time required for completing the task was 13 minutes. By defining a segment thumbnail as the first five seconds after a segment boundary we can include 87% of human thumbnails. A preliminary examination of text annotation showed that about 60% of all words into three categories: sound source descriptions, structural music theoretic descriptions, and basic acoustic parameters.

ACKNOWLEDGMENTS

The authors would like to acknowledge support from Intel, Interval Research and Arial Foundation.

REFERENCES

1 C. van Rijsbergen, *Information retrieval*, Butterworths, London, 2nd edition, 1979.

2 J. Foote, "An overview of audio information retrieval," *ACM Multimedia Systems*, vol. 7, pp. 2–10, 1999.

3 A. Hauptmann and M. Witbrock, "Informedia: News-on-demand multimedia information acquisition and retrieval," in *Intelligent Multimedia Information Retrieval*, chapter 10, pp. 215–240. MIT Press, Cambridge, Mass., 1997, http://www.cs.cmu.edu/afs/cs/user/alex/www/.

4 E. Wold, T. Blum, D. Keislar, and J. Wheaton, "Content-based classification, search and retrieval of audio," *IEEE Multimedia*, vol. 3, no. 2, pp. 27–36, 1996.

5 E. Wold, T. Blum, D. Keislar, and J. Wheaton, "A content-aware sound browser," in *Proc. 1999 ICMC*, 1999, pp. 457–459.

6 G. Tzanetakis and P. Cook, "Multifeature audio segmentation for browsing and annotation," in *Proc.1999 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, WASPAA99*, New Paltz, NY, 1999.

7 S. Wake and Asahi.T, "Sound retrieval with intuitive verbal expressions," in *Proceeding of International Conference on Auditory Display*, Glaskow, 1997, ICAD.

8 G. Tzanetakis and P. Cook, "A framework for audio analysis based on classification and temporal segmentation," in *Proc.25th Euromicro Conference. Workshop on Music Technology and Audio Processing*, Milan, Italy, 1999, IEEE Computer Society.