Query Music Database with Video by Synesthesia Observation

Ruiduo Yang and Michael S. Brown

Department of Computer Science
Hong Kong University of Science and Technology
Clear Water Bay, Kowloon, Hong Kong
{yangrd,brown}@cs.ust.hk

Abstract

In this paper, we proposed a novel framework to retrieve a music database by using a query of MPEG video sequence. In particular, music and video are considered as 2 time series data. We selected corresponding features in the music and video sequence. For example: the tempo in the music and the motion in the video. A novel similarity measure for matching these features is advocated to capture the synesthesia effect in music-video.

1. Motivations and related work

Multimedia information retrieval is becoming an attractive technique and hot research topic. Generally, multimedia information retrieval system can be divided into two parts: audio retrieval and video/image retrieval. During the retrieval process, those audio/video/image candidates, who are similar to the query audio/video/image in terms of either feature similarity or semantic similarity or both, will be returned as results.

Various techniques have been invented by people to do both visual and aural information retrieval, nevertheless, theses techniques have been designed separately for visual and audio information. The matching between visual information and aural one has been far more ignored while we believe technique as this will have great application for professional/amateur multimedia editing such as animation design, advertisement design, music video design etc., which are not provided even by the commercial multimedia editing tools like Macromedia Flash [1] and Adobe Premier [2].

Through the information mapping from audio to visual, there is synesthesia among them. Synesthesia is a condition in which one type of stimulation evokes the sensation of another, as when the hearing of a sound produces the visualization of a color. Forerunners in art and computer animation have been aware of synesthesia for a long time. In [3], John Whitney exploited the relationship between music and simple geometry animation. He also invent the computer technique called digital harmony, which has been widely used in today's computer techniques, for example the MP3 player winamp [4] and Microsoft Media player [5]. Digital harmony tries to firstly detect the underline features inside the music such as transcription, genre, tempo etc, after which the simple animation will be made in terms of dot, line, shape and their motion so that the synesthesia can be made automatically.

Similar efforts for making synesthesia between audio and video can be found in the state-of-art work. Hiraga et al. [6] researched technique to do music performance visualization, by which the personal performance of an instrument can be visualized as 3D animation. While Foote et al. tries to resize the video so that it can be perfectly matched to a given music [7]. Their work can be considered as extension of digital harmony.

Previous works have also been revealing and summarizing the interesting relationship between aural and visual information. Table 1. [8] Shows us a particular correspondence between these two.

	Hue	Saturation	Value	Shape
Pitch	Color Scales		Dark is deep	Size to pitch
Amplitude		Loud or Muted		
Overtones	Color Tone & overtones			Point or line
Tempo		Modulation to nuance		Fast is sharp
Interval	Contrast intervals			
Mode	Mode to color shade			

Table 1: Correspondence between music feature and visual feature in HSV model [8].

Besides the synesthesia, great efforts have been made in recent years to do music information retrieval. This is due

to that the development in internet technology have made a large volume of music audio data available to the general public while the music search technique is not as successful as their counterpart for text search. Yang et al. has given an excellent review for the state-of-art's music retrieval technique in [9]. In particular, Yang et al. have divided the music data into 3 categories: symbolic music (music with transcriptions, for example MIDI), monophonic music and polyphonic music. We extended Yang's classification as in Table 2. In the cell marked S1, both the query and the underlying database are in symbolic formats, in the cell marked by QBH, the monophonic acoustic query with a symbolic database represents a problem which is known as Query by humming, or QBH [10]. Finally the cell marked with S2 represents the problem that Yang et al. want to solve in [9]. In particular they use the spectrograms of the music to process and match music.

	Symbolic	Acoustic	Visual
Symbolic	S1	QBH	
Acoustic		S2	**A**
Visual			Video/image retrieval

Table 2: Classification of multimedia retrieval, the different column indicates different query data, while different row indicates different database.

We extend the last column and row on the base of [9]. The cell marked as "A" indicates the problem that we want to attack in this paper. In particular, in our system the incoming query will be visual such as a video sequence or an image, with the corresponding music returned as the result to match it to make a music video / animation. By doing this, we expect the synesthesia can be automatically build for those professional/amateur computer animation designer.

This paper is organized as follows: section 1 show the motivation and related work discussion. In section 2, framework for our system will be overviewed. Section 3 discusses the proposed similarity measure for matching music and video followed by the experiment results in section 4. Finally Section 5 concludes this paper with future work.

2. Music retrieval system framework

In the proposed framework, we will use a generic MPEG video sequence as input query. The database consists of large volumes of music data. Following [9], we considered this music data as polyphonic because this is the usual case for computer music video/animation design. The system is outlined at figure 1. We propose to track the features of the music sequences and store them as the time series. The input video sequence needs also to be processed for extracting the corresponding features which can be referenced as in table 1. A new correlation measure will be set up (see section 3) specifically for music-video data matching in our system and the corresponding music sequence with higher correlation score will be returned as results.

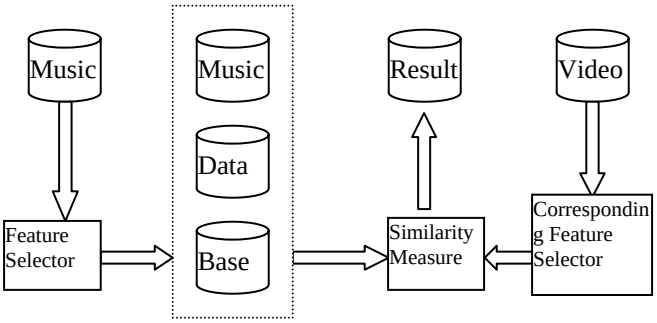

Figure 1: Framework for our multimedia retrieval system.

3. A novel correlation measure by synesthesia observation

In our Retrieval System, both the Video Sequence and the Music Sequence has been considered as a time series. A time series is a sequence of real numbers, representing the measurements of a real variable at equal time intervals [11]. There is several similarity measures set up for matching 2 time series besides Euclidean Distance. Goldin and Kanellakis et. al. suggests normalization to capture the variance and mean [12]. Jagadish et. al. use transformation rules.[13] . Rafiei et al. use moving averages to smooth data [14]. And Berndt proposed the dynamic time warping approach [15]. Nevertheless, none of these methods take the considerations of the synesthesia effect when matching a music sequence to a video sequence. A user can have a several times stronger feeling when synesthesia between a music and video happens than they occur separately. Based on this fact, a new similarity measure is used in our system:

We denote the Music and video as:

$\text{Music}: M_i$ i = 1,..., n is the time index

$\text{Video}: V_i$ i = 1,..., n is the corresponding time index

In which $M_i =< x_1,...,x_k >$, $V_i =< y_1,...,y_k >$, with the x_1 to x_k to be the feature of music, which can be tempo, pitch, amplitude etc. while y_1 to y_k is the corresponding

visual features of the video, which could be motion, hue, shape etc.

For any observation $< M_i, V_i >$ we generate the corresponding synesthesia observation which is:

$< SM_{i*2-1}, SV_{i*2-1} >$ and $< SM_{i*2}, SV_{i*2} >$

In which:

$$SM_{i*2-1} = M_i, \quad SV_{i*2-1} = \arg\min_{V_j(|j-i|<\delta t)} dist(M_i, V_j)$$

$$SV_{i*2} = V_i, \quad SM_{i*2} = \arg\min_{M_j(|j-i|<\delta t)} dist(M_j, V_i)$$

Here, δt is a reasonable window size in which the synesthesia can happen. *dist* is the normalized distance function between any pair of music-video data. Hence for the time series M and V, we generate their corresponding synesthesia time series as:

$$\text{Music}: SM_i \; i = 1,...,n \text{ is the time index}$$

$$\text{Video}: SV_i \; i = 1,...,n \text{ is the corresponding time index}$$

At the last step, the similarity measure score is computed by the correlation of the transformed synesthesia time series:

$$Score = \frac{\sum_{i=1,...,2n} (SM_i - \overline{SM}) \bullet (SV_i - \overline{SV})}{\sqrt{\sum_{i=1,...,2n} (SM_i - \overline{SM})^2 \bullet \sum_{i=1,...,2n} (SV_i - \overline{SV})^2}}$$

4. Tempo tracking for music and motion vector synthesis for video

In our system, the assumption is made that the tempo of music will be related to the motion of the video frame by frame. Tempo is defined for the speed of the music. Scheirer et.al has proposed the excellent tempo tracking method and the test of large amount of music has proved its reliability [16].

We followed this tempo tracking method. Suppose the original music signal is m_i. After tempo tracking, we have $T(m_i)$. In order to match this data with the video, $T(m_i)$ will be smoothed and windowed as the input video's frame rate. Let's suppose the input video has a frame rate of f (fps). The smoothing function is $S(x)$ and the windowing function is $W(x, f)$. We have the music data to be:

$$M_i = W(S(T(m_i)), f);$$

For video data, since we assume it is MPEG compressed the motion vector inside each frame can be directly extracted from the compressed stream. For each frame I, we define the motion energy as:

$$ME_i = \frac{\sum_{j \text{ for all block}} |mv_j|}{N_{1i}} + \frac{\sum_{j \text{ for all block}} |residual_j|}{N_{2i}}$$

Here, *mv* and *residual* can be directly extracted from the MPEG Video. N_{1i} is defined as the number of non-stational (non-zero motion vector) block, while N_{2i} is defined as the number of non-stational (non-zero residual) block. Both N_{1i} and N_{2i} are used to capture the local motion energy.

In this representation, each pair of observation at a certain time is denoted as:

$$< M_i, ME_i >$$

After this, we will generate the synesthesia time series and compute the correlation score. The algorithm is showed below:

1. Track the tempo of each music data in the database, get M_i.
2. For the input MPEG video. Compute the ME_i
3. For each music sequence in the database, generate the synesthesia time series $< SM, SV >$.
4. Return a sorted list of music with the highest correlation score of SM and SV

In the experiment, we use several professional made music videos as the test example. Our music data base has 100 polyphonic music sequences, which includes the extracted music sequences for those professional made music-video. Table 3 shows us the matching result. In Table 3 column 1 is the name of the selected music videos. Column 2 returns the correlation score between the original video and their music part. Column 3 returns the highest correlation score between the original video and the music in the database while column 4 is the average correlation score. The 5th column returns the rank of the original music

Sequence	Original Score	Highest Score	Average Score	Rank
Basket Ball	0.23	0.23	0.08	1
Cell Phone Ad.	0.36	0.36	0.1	1
Computer Ad.	0.30	0.30	0.1	1
Company Ad	0.38	0.38	0.12	1
Flash	0.33	0.33	0.07	1

Table 3. Experiment Result for 5 professional made music videos. In the experiments, all the video has a frame rate of 30fps. Sequence "basketball" has 2000 frames while the other sequence has 900 frames. We set δt to be 0.05 seconds.

From the results we can see, the original music always has the best correlation with the input video. This is because the music-video is intended to maximize the synesthesia effect. The other music, however, may have the boundary mismatching or tempo mismatching with the original video, hence penalties the correlation scores.

5. Conclusions and future work

In this paper, we propose a novel framework to do automatic music retrieval using a video query. We also provided a novel similarity measure to effectively match the video and music data. We implemented the idea based on the assumption that the motion in video can be related to the tempo in music. The coming work can include more corresponding futures between video and music. A learning based approach can also be taken with a large database of professionally made music video. The efficiency of the indexing and retrieval algorithm should also be taken consideration.

6. REFERENCES

[1] Macromedia Flash MX. Visit: July 26, 2003. http://www.macromedia.com/software/flash

[2] Adobe Premier. Visit: Nov 14, 2003. http://www.adobe.com/products/premiere/main.html

[3] J. Whitney, "Digital Harmony." McGraw-Hill, Peterborough, NH, 1980

[4] Winamp. Visit: Nov 14, 2003. http://www.winamp.com/

[5] Media Player. Visit: Nov 14, 2003. http://www.microsoft.com/windows/windowsmedia/default.aspx

[6] R. Hiraga, R. Mizaki, I. Fujishiro, "Performance visualization – a new challenge to music through visualization.", in Proc. ACM Multimedia, 2002

[7] J. Foote, M. Cooper, A. Girgensohn, "Creating Music Videos using Automatic media Analysis", in Proc. ACM Multimedia, 2002.

[8] Correspondence. Visit: July 26 2003, http://rythemlight.com

[9] C. Yang, "Efficient Acoustic Index for Music Retrieval with Various Degrees of Similarity", in Proc. ACM Multimedia, 2002.

[10] A. Ghias, J. Logan, D. Chamberlin and B. Smith, "Query by humming – Musical Information Retrieval in an Audio Database", in Proc. ACM Multimedia, 1995.

[11] D. Gunopulos and G. Das. "Time Series Similarity Measures.", In Tutorial Notes of the Association for Computing Machinery Sixth International Conference on Knowledge Discovery and Data Mining, pages 243--307, 2000

[12] D.Q. Goldin, P.C. Kanellalds. On Similarity Queries for Time-Series Data: Constraint Specification and Implementation. CP 1995

[13] H. V. Jagadish , A. O. Mendelzon , T. Milo, "Similarity-based queries,", Proceedings of the fourteenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems, p.36-45, May 22-25, 1995, San Jose, California, United States

[14] D. Rafiei , A. Mendelzon, "Similarity-based queries for time series data,", Proceedings of the 1997 ACM SIGMOD international conference on Management of data, p.13-25, May 11-15, 1997, Tucson, Arizona, United States

[15] D. J. Berndt , J. Clifford, "Finding patterns in time series: a dynamic programming approach,", Advances in knowledge discovery and data mining, American Association for Artificial Intelligence, Menlo Park, CA, 1996

[16] E.D. Scheirer, "Tempo and beat analysis of acoustic musical signals,' J. Acousti. Sot. Am., 1998, vol. 103, no. 1, pp. 588-601.