Automatic Music Video Summarization Based on Audio-Visual-Text Analysis and Alignment

Changsheng Xu[1], Xi Shao[1, 2], Namunu C. Maddage[1, 2], Mohan S. Kankanhalli[2]

[1]Institute for Infocomm Research
21 Heng Mui Keng Terrace
Singapore 119613
65-68748248
{xucs, shaoxi, maddage}@i2r.a-star.edu.sg

[2]School of Computing
National University of Singapore
Singapore 117543
65-68746738
mohan@comp.nus.edu.sg

ABSTRACT
In this paper, we propose a novel approach for automatic music video summarization based on audio-visual-text analysis and alignment. The music video is separated into the music and video tracks. For the music track, the chorus is detected based on music structure analysis. For the video track, we first segment the shots and classify the shots into close-up face shots and non-face shots, then we extract the lyrics and detect the most repeated lyrics from the shots. The music video summary is generated based on the alignment of boundaries of the detected chorus, shot class and the most repeated lyrics from the music video. The experiments on chorus detection, shot classification, and lyrics detection using 20 English music videos are described. Subjective user studies have been conducted to evaluate the quality and effectiveness of summary. The comparisons with the summaries based on our previous method and the manual method indicate that the results of summarization using the proposed method are better at meeting users' expectations.

Categories and Subject Descriptors
H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing – *abstract methods, indexing methods.*

General Terms
Algorithms, Performance, Experimentation

Keywords
Music video, summarization, chorus, shot, lyrics, alignment

1. INTRODUCTION
The music video genre began to have wide popularity and influence in the early 1980s and it has attracted an increasingly large viewer-ship of different age levels shortly after its introduction. The style and content of music videos have strongly influenced advertising, television, film, and popular culture as a whole. With the rapid development of various technologies for multimedia content capture, storage, high bandwidth/speed transmission and compression standards, the production and distribution of music videos have increased rapidly and become more available. Nowadays, many music content providers and companies are putting their music videos on the websites and customers can purchase these music videos through the websites. However, an average customer would prefer to watch the highlights first before deciding their purchase. Although music video summaries are available at most music websites, they are generated manually, which is a very laborious process. Therefore, how to automatically create a concise and informative summary of an original music video is a challenging task and it is commercially relevant to come up with an automatic summarization approach for music videos.

Automatic music summarization and video summarization have attracted research activity in the past few years. Automatic music summarization approaches can be classified into machine learning based approaches [1,2,3] and pattern matching based approaches [4,5,6]. The challenge in music summarization is to determine the relevant features and make the final summary boundaries correspond to the meaningful music section (e.g. chorus) boundaries. Automatic video summarization approaches have been successfully applied to sports video [7], news video [8], home video [9] and movies [10], but relatively little work focused on music video analysis and summarization. We proposed a music video summarization method [11], which generated music summary and shot clusters separately. The final music video summary is created by aligning the music summary and clustered video shots. An obvious drawback of this method is the boundaries of music segments in the final summary are discontinuous. A recent method [12,13] has also proposed a music video summarization system, which is based on high-level metadata such as titles, artists, lyrics, etc. However, these metadata, especially the lyrics, are not easy to be obtained directly from the music video content. Therefore, assumption of availability of these metadata makes the problem easier and is not applicable to automatic summarization based on music video content only.

In this paper, we propose a novel automatic music video summarization approach, which combines the complementary strengths of low-level features and high-level music knowledge. We believe that the combination of bottom-up and top-down approaches is powerful to analyze and summarize music video content. The proposed method automatically extracts the metadata

ACM SIGIR'05, August 15–19, 2005, Salvador, Brazil.

from low-level audio/visual/text features and music knowledge instead of assuming the availability of these metadata as in [12,13]. Figure 1 illustrates the workflow of the proposed music video summarization approach. The music video is separated into music track and video track. For the music track, choruses are detected based on music structure analysis. For the video track, the video shots are segmented and classified into close-up face shots and non-face shots. Then the lyrics are detected and recognized from the shots. The most repeated lyrics are further identified. The music video summary is created by the alignment of the boundaries of the detected chorus, shot class and repeated lyrics.

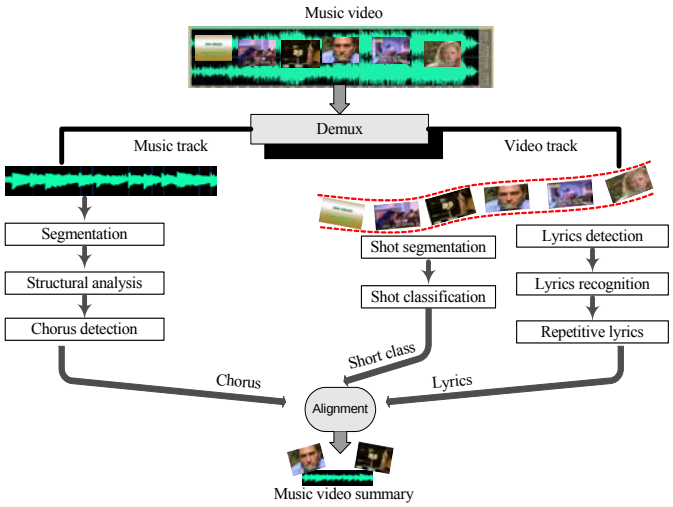


Figure1. Workflow of the music video summarization

The rest of the paper is organized as follows. Analysis of music, video and text extracted from the music video content is described in section 2, 3, and 4 respectively. Music video summarization scheme is discussed in section 5. Experimental and evaluation results are reported in section 6. We conclude the paper with hints for future work in section 7.

2. MUSIC ANALYSIS

The unique characteristic of the music video genre is that the music plays the dominant role in a music video, while in other genres such as sports video and home video the dominant part is the visual track. Therefore, the music video summarization should be based on music structure analysis, which is different from the existing approaches of video summarization. Based on the music knowledge [14], the chorus is melodically stronger than other parts and can be used as a thumbnail for the music content. The earlier approaches on music structure analysis [15,16] have not fully exploited music knowledge and addressed how to estimate the boundaries of music sections. To tackle this issue, we have developed a music structure analysis method for popular music with 4/4 time signature [17]. We will discuss how to detect the chorus from the music in the following subsections.

2.1 Segmentation

Typical audio including music segmentation approaches use fixed length intervals (20~40 ms) to segment audio signals. It works well for speech [18] but may not be appropriate for music. Compared to speech, music signals are heterogeneous because the signal sources change when the music score progresses with time. Thus it is difficult to judge the size of the signal section which can be considered as quasi-stationary unless the music domain knowledge is applied. According to music composition theory [14] the ideal segmentation for more accurate vocal/instrumental boundary detection and melody contour extraction is to segment the music based on the length of individual music notes. In order to perform such segmentation, it is required to have an accurate onset detector to find all the note onsets in a music song. However it is very difficult to detect all the onsets because of the polyphonic nature of the music signals.

From theory of music [14], we know that usually smaller length notes (eighth or sixteenth note) are played in the bars to align the melody with the rhythm of the lyrics and fill the gap between lyrics. Thus we propose a novel segmentation scheme to detect the length of the smallest note (eighth or sixteenth note) and segment the music into the smallest note length frames. Figure 2 illustrates the steps to detect the smallest note length.

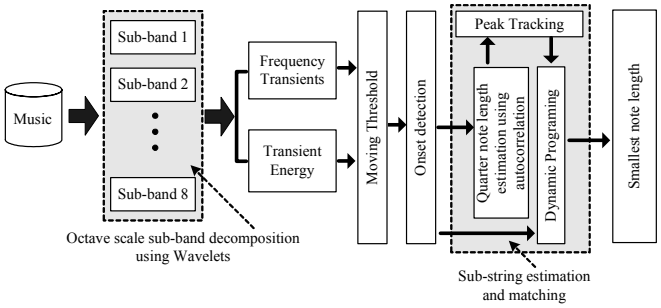


Figure2. The smallest note length detection

Table 1. Octave frequency ranges of sub-bands

Sub-band No	01		02	03	04	05	06	07	08
Octave scale	~ B1	C2 ~ B2	C3 ~ B3	C4 ~ B4	C5 ~ B5	C6 ~ B6	C7 ~ B7	C8 ~ B8	Higher Octaves
Freq-range (Hz)	0 ~ 64	64~128	128~256	256~512	512~1024	1024~2048	2048~4096	4096~8192	(8192 ~ 22050)

We first decompose the music signal into 8 sub-bands, whose frequency ranges are in octave scale (Table 1), based on the consideration that music harmonic structures are in octaves [18]. Both the frequency and energy transients are analyzed for each sub-band. An energy-based detector is used for the upper sub-bands (05-08) to detect the strong transient note onset, while a frequency-based distance measure is used for the lower sub-bands (01-04), because fundamental frequencies (F0s) and harmonics of music notes are strong in these sub-bands. In order to detect the rhythm progression in different note level, we take the weighted summation (Equation (1)) of onsets detected in each sub-band, where *On(t)* is the sum of onsets detected in all eight sub-bands $Sb_i\ (t)$ at time '*t*' in the music signal. In our experiments, it is noticed that hard onsets generated from bass drums, bass guitar and bass notes of piano are found in sub-band 01 and 02. The timing of snares and side drums are highlighted in sub-band 06 to 08. These onsets indicate the bar timing. The soft onsets (treble clef notes) are typically found in sub-band 03 to 05. Thus the weight matrix w = {0.6, 0.9, 0.7, 0.9, 0.7, 0.5, 0.8, 0.6} is empirically found to be the best set for calculating hard and soft onsets to extract the inter-beat time lengths.

$$On(t) = \sum_{i=1}^{8} w(i) \cdot Sb_i(t) \quad (1)$$

The initial inter-beat length is estimated by taking the autocorrelation over the detected onsets. We employ the dynamic programming approach to check for patterns of equally spaced strong and weak beats among the detected onsets and compute both inter-beat length and the smallest note length. Figure 3 (a) is a 10-second song clip. The detected onsets are shown in Figure 3 (b). The autocorrelation of the detected onsets is shown in Figure 3 (c). The sixteenth note level segmentation is shown in Figure 3 (d). The sixteenth note length is 112.10625 ms.

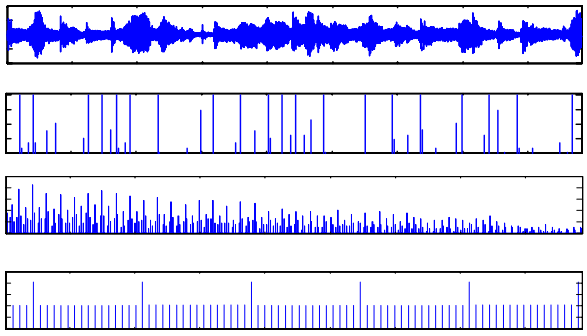


Figure3. 10 seconds clip of the song

2.2 Structure Analysis and Chorus Detection

Music structure is important information for music semantic understanding. Its components, e.g. Introduction (Intro), Verse, Chorus, Bridge, Instrumental and Ending (Outro), construct the melody-based similarity regions and content-based similarity regions. We define melody-based similarity regions as similar pitch contours constructed from the chord patterns and content-based similarity regions as the regions which have both similar vocal content and similar melody. For example, Verse sections in a song can be considered as melody-based similarity regions while Chorus sections as content-based similarity regions.

The first step for music structure analysis is to segment the music into frames with the smallest note length using the method proposed in the previous section. Then the melody-based and content-based similarity regions are detected. Finally, the music structure is formulated and choruses are extracted based on detected melody-based and content-based similarity regions and music knowledge.

2.2.1 Melody-based similarity region detection

The melody-based similarity regions have the similar chords patterns. Therefore, in order to detect the melody-based similarity regions, the chords of each segment are detected and sub-chord patterns are matched with the whole music song using dynamic programming [19].

A chord is constructed by playing 3 or 4 music notes simultaneously. Thus the key idea to identify the chord is to detect the fundamental frequencies (F0s) of notes which comprise the chord. We use a method similar to the one described in [20] for chord detection. The Pitch Class Profile (PCP) features, which are highly sensitive to the F0s of notes, are extracted from training samples to model the chord with Hidden Markov Model (HMM) [17]. We use 48 HMMs to model 12 Major, 12 Minor, 12 Diminished and 12 augmented chords. Each model has 5 states and 3 Gaussian Mixtures (GM) for each hidden state. The mixture weights, means and covariance of all GMs and initial and transition state probabilities are computed using Baum-Welch algorithm [21]. The Viterbi algorithm [21] is applied to find the efficient path from starting to end state in the models.

In our experiments, we find that sometimes the observed final state probabilities of HMMs corresponding to the chord pairs are high and close to each other. This may lead to wrong chord detection. Thus we apply heuristic rules based on music composition to correct the detected chords and the time alignment of the chords [17].

2.2.2 Content-based similarity region detection

The melody-based similarity regions which have similar vocal content are defined as content-based similarity regions. Therefore, after melody-based similarity regions are detected, it is important to decide which regions have similar vocal content.

Singing voice boundary detection is the first step to analyze the vocal content. We use the "*Octave Scale*" to calculate Cepstral coefficients [22] to represent the music content because the sung vocal lines always follow the instrumental line such that both pitch and harmonic structure variations are in octave scale. In our approach we divide the whole frequency band into 8 sub-bands corresponding to the Octaves in music. Cepstral coefficients are extracted from the Octave Scale [22]. Singular value decomposition is applied to find the uncorrelated Cepstral coefficients for Octave scale. We use the order range of 10-16 coefficients for Octave scale. Support vector machine [23] with radial based kernel function (RBF) is used to identify the instrument and vocal frames. Then the similarities are measured between the similar melody-based similarity regions, and the regions with high similarity are defined as the content-based similarity regions [17].

2.2.3 Chorus detection

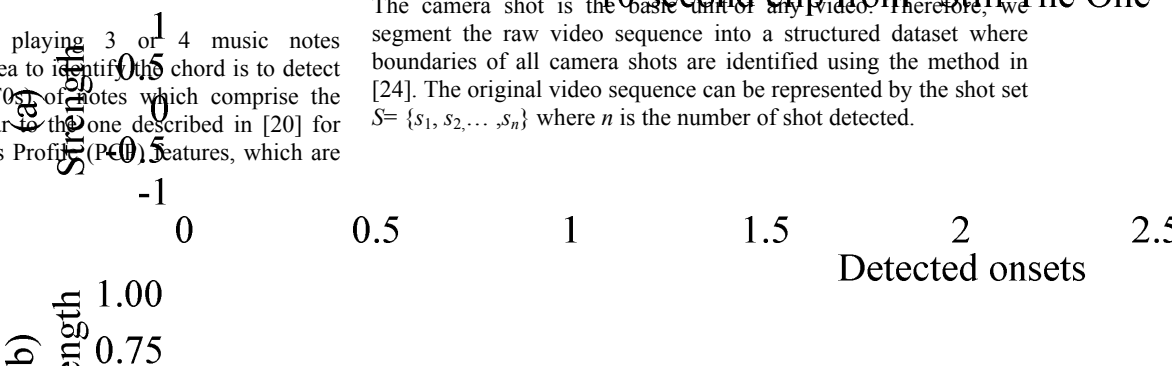
Based on detected melody-based and content-based similarity regions and music knowledge [17], music structure can be identified. The choruses are detected from the content-based similarity regions. The details can be found in [17]. The detected *i*th chorus in a music song is represented as: $chorus_i$=<Start-B_i, End-B_i>, where Start-B_i and End-B_i denote the start and ending boundaries of the chorus.

3. VIDEO ANALYSIS

The purpose of video analysis is to detect and classify video shots as well as help detect lyrics from the video frames so as to align the detected music chorus to create a continuous and meaningful music video summary. Video analysis includes camera shot boundary detection and semantic shot classification.

3.1 Shot Segmentation

The camera shot is the basic unit of any video. Therefore, we segment the raw video sequence into a structured dataset where boundaries of all camera shots are identified using the method in [24]. The original video sequence can be represented by the shot set $S = \{s_1, s_2, \ldots, s_n\}$ where n is the number of shot detected.



For each shot s_i, we choose a key frame f_i as the representative frame of the shot. To detect the most salient lyrics appearing stably in the shot, the representative frame f_i is selected in the middle of the shot instead at the two ends of the shot boundary, because the shot boundaries commonly contain transition frames which will blur the lyrics caption. The lyrics detection will be discussed in section 4.

3.2 Shot Classification

In order to better represent the semantic meaning of the shots, we further classify the detected shots into two categories: close-up face shot and non-face shot. Face is an important characteristic in music videos, which may indicate the singer or actor/actress in the music video. Therefore, the music summary should contain the face shots. The face and non-face shots alternatively appear in the music video, when the semantic meaning changes for the video content. The most salient difference between the close-up face shot and non-face shot is camera motion and the features of the object (i.e. face). This motivates us to use following features for shot classification:

(1) Camera motion: As the camera always follows the movement of the object, the camera motion provides a useful cue to represent the activity and characteristic of the object. In our approach, we use "average motion magnitude", "motion entropy", "dominant motion direction", "camera pan parameter", "camera tilt parameter" and "camera zoom parameter". These features are computed using the Motion Vector Field extracted from the compressed video.

(2) Face: Face is an important characteristic of the close-up shot. If faces are detected in the shot, this shot should be close-up shot. We use a skin color based method [25] for face detection.

To accurately classify the shot candidates, the above features from individual shots are fed into a classifier. We use support vector machine (SVM) here since SVM is a useful statistical machine learning technique that has been successfully applied in the pattern recognition area [23]. In our approach, the SVM kernel function is a Gaussian Kernel.

The *i*th shots in the music video can be represented as: $shot_i$=<Start-B, End-B, Class>, where Start-B and End-B denote the start and ending boundaries of the shot and Class indicates that the shot is close-up or non-close-up.

4. TEXT ANALYSIS

The purpose of text analysis is to make use of lyrics appearing in the music video to help align the chorus to create music video summary. Note that many music videos do have the lyrics of the video appear as visual text superimposed on the video frames. Lyrics are good cues indicating the structure information of the music video. Here we extract the lyrics directly from music video frames. The text analysis includes three steps. Firstly, for each frame in the representative frame set, we detect whether the frame contains lyrics or not. Secondly, the lyrics recognition is applied to those frames with the lyrics. Finally, repeated lyrics are grouped together to find the most repeated lyrics.

4.1 Lyrics Detection

Given the representative frame set $F= \{f_1,f_2,\ldots,f_n\}$, text detection is applied to each representative frame f_i, using the method proposed in [26].

Several heuristic rules related to lyrics of the music video are used to facilitate the lyrics detection.

a) Lyrics always appear in the lower half part of the frame.

b) Lyrics caption is a bar whose width is larger than height.

4.2 Lyrics Recognition

The frames containing the lyrics are used to generate the lyrics frame set F', where $F'=\{f'_1,f'_2,\ldots,f'_m\}\subseteq F$.

For each frame in the lyrics frame set F', the content of each lyrics is recognized. The low resolution of video (typically 72 dpi) is a major source of problems in text recognition. OCR (Optical Character Recognition) systems have been designed to recognize text in documents, which were scanned at a resolution of at least 200dpi to 300dpi resulting in a minimal text height of at least 40 pixels. In order to obtain good results with standard OCR system, it is necessary to enhance the resolution of segmented text lines. In our experiment, we use cubic interpolation to rescale the text height (normally about 20 pixels) into 40 pixels while preserving the aspect ratio.

It should be noted that although there is no OCR software can achieve 100% accuracy, it will not affect the final result much, as the error can be supplemented by the following approximate string matching operation.

After text recognition, the recognition results are saved in a lyrics set $C=\{c_1,c_2,\ldots,c_m\}$. Each element c_i in this set corresponds to the text content of frame f'_i in lyrics frame set F'.

4.3 Repeated Lyrics Detection

The aim of repeated lyrics detection is to find the most salient part of a music video. We assume that the most salient part of a music video happens in the most salient music part (i.e. chorus). Although what makes a music part distinguished among a music work is not clear, current research typically assumes it to be the most repeated part.

Generally, chorus of a song contains the most repeated music phrases. In this paper, a music phrase is defined as a short musical passage, which is similar to linguistic sentence in the speech.

Considering the lyrics set C obtained in the previous step. Since a music phrase lasts for several shots which may correspond to several continuous lyrics in the lyrics set C, we need to merge these continuous lyrics into one to represent the music phrase corresponding to it. After the merging process, the music phrase set $P= \{p_1,p_2,\ldots,p_t\}$ is formulated.

Given the music phrase set P, we use dynamic programming [19] to match each lyrics (i.e., p_i) with the lyrics sequence starting from this lyric (i.e.$p_i\ p_{i+1}\ldots p_t$), as it has been proven efficient for string matching that allows errors, or called approximate string matching. Suppose we need to match the lyric p_i (denoted as X) with the lyrics sequence starting from this lyric (denoted as Y), we should fill a edit distance matrix $\boldsymbol{D_i}$ (X,Y), which is defined as minimum cost of a sequence of modification (insertion, deletions and substitution) that transforms X into Y. In the matrix, the element $D_i(k,l)$ represents the minimum number of modifications that are needed to match $X_{1..k}$ to $Y_{1..l}$. The algorithm can be described as following:

Initial: $D_i\,(k,0)=k\ ; D_i\,(0,l)=0;\ 1\le k\le|X|,\ 1\le l\le|Y|$

Recurrence:

$$D_i(k,l) = \min\begin{cases} D_i(k-1,l-1)+\delta(X_k,Y_l) & 1 \le k \le |X| \\ D_i(l-1,k)+1 & 1 \le l \le |Y| \\ D_i(l,k-1)+1 \end{cases} \quad (2)$$

where $\delta(X_k, Y_l)$ =0 if X_k=Y_l and 1 otherwise. $|X|$ and $|Y|$ denote the length of string X and Y respectively.

The rationale for above formula can be explained as follows. D_i (k,0) and D_i (0,l) represent the edit distance between a string of the length k or l and the empty string. For D_i (k,0), clearly k deletions are needed on the non-empty string. While for D_i (0,l), because we allow that any text position in Y can be the potential start matching point, we set the first row of the matrix to zeros, which means the empty pattern matches with zero errors at any text position.

The last row of Matrix $\boldsymbol{D_i}(X,Y)$ is defined as function $h_i(r)$,r=1.. $|Y|$. It measures how well the string X matches with different locations shifted by r in the string Y. Figure 4 plots out one of the lyrics repetition detection results for the music video "Yesterday Once More".

It can be seen from Figure 4 that except for p_i itself (the first local minimum denoted with circle in Figure 4), there are other three matching points, also denoted with circles. These three matching points are not equal to zero (the best possible) because of the OCR errors. We can set a threshold to find the local minimum of function $h_i(r)$. In our implementation, the threshold is set to 2*(1-OCR accuracy) multiplying the length of text p_i.

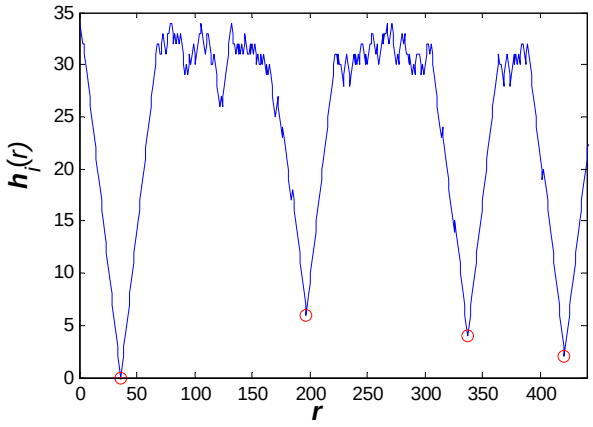


Figure4. One lyrics repetition detection result

Thus, the task to find the salient part of the music can be converted to the task to find the most repeated music phrase in the set P. The detailed algorithm is described below:

1) Take the first element in set P, and use dynamic programming to find the repeated music phrases in set P.

2) Select the first element in set P, together with its repeated music phrases found to construct a subset R_j. Meanwhile, delete these music phrases in set P. Increase j.

Repeat step 1) and 2) until P is empty.

The set R=$\{R_1,\cdots R_j,\cdots,R_k\}$ contains the k subsets, each subset R_j represents a cluster containing the same music phrase in set P.

By counting the number of element for each subset R_j in set R, we can find the subset containing the most repeated music phrase, denoted as R^*_{opt}. The ith repeated lyrics in the music video can be represented as: lyrics$_i$=<LyricStart-B$_i$, LyricEnd-B$_i$ >, where LyricStart-B$_i$ and LyricEnd-B$_i$ denote the start and ending boundaries of the lyrics in time line.

5. SUMMARY GENERATION

The final music video summary is created based on the most salient part detected from both music track and visual track. For the music track, the Chorus is the strongest and most repeated part of the song [14]; while for the visual track, the most salient part contains the most repeated lyrics. However, since the chorus detected in the music track is not always consistent in time line with the most repeated lyrics detected in the visual track, we need to align the music and visual part to make the final music video summary meaningful and smooth.

5.1 Music-Visual-Text Alignment

The purpose of music-visual-text alignment is to synchronize the most salient parts detected from the music track and visual track so as to make the final music video summary meaningful and smooth. Assume the i-th chorus in a music song is represented as: chorus$_i$=<Start-B$_i$, End-B$_i$>, and the corresponding lyrics are represented as: lyrics$_i$=<LyricStart-B$_i$, LyricEnd-B$_i$ >. Generally, the time line of Start-B$_i$ is not equal to LyricStart-B$_i$, neither is End-B$_i$ equal to LyricEnd-B$_i$ due to two reasons. The first reason is that the lyrics in the music video generally appear earlier and last longer than the corresponding singing voice in time line. This will result in LyricStart-B$_i$ less than Start-B$_i$ and LyricEnd-B$_i$ bigger than End-B$_i$. The second reason is that the shots which are considered as unstable (last less than 0.3 seconds) are discarded in our approach. This will result in LyricStart-B$_i$ bigger than Start-B$_i$ and LyricEnd-B$_i$ less than End-B$_i$. In light of this, we create the music video summary based on the following rules:

1) Construct a dataset Φ containing all choruses and its corresponding lyrics in the song as the candidates set, denoted by Φ ={(chorus$_1$ lyrics$_1$),⋯,(chorus$_i$ lyrics$_i$), ⋯, (chorus$_n$, lyrics$_n$)}, where n is the number of choruses detected. For each chorus (i.e. the i-th chorus), we represent it using two time lines (i.e. chorus$_i$=<Start-B$_i$, End-B$_i$>) and each corresponding lyrics in the music video can be represented as: lyrics$_i$=<LyricStart-B$_i$, LyricEnd-B$_i$ >.

2) For each chorus in dataset Φ, we compare the start time and the ending time between chorus and its corresponding lyrics respectively as a matching factor to measure how well these two elements (chorus and lyrics) are matched. For example, if the start time and the ending time of the lyrics (actually the start time and the ending time of the lyrics corresponding the shot boundaries) fall in the ±1 second of start time and the ending time of its corresponding chorus, we consider that this pair matching is better than the pair with ±2 deviation. Then we order the dataset Φ according to this matching factor. In addition, for each chorus in dataset Φ, we need to find the corresponding shot type (close-up face shot or non-face shot). The first pair in the matching factor order that contains close-up face shot will be selected as the seed to generate the music video summary, we denote it as (chorus*, lyrics*). If all shots

corresponding to choruses in dataset Φ are non-face shots, then the first pair in dataset Φ in matching factor order will be selected as the seed (chorus*, lyrics*) to generate the music video summary.

3) Once the seed pair (chorus*, lyrics*) has been found, we can create music video summary based on it. For the seed pair (chorus*, lyrics*), we take chorus* as the stable element, and align the shots corresponding to it. To make the music and visual content synchronal, we select the visual content according to the time line of chorus.

Table 2: The music phrase length of the different note levels

	Bar Length	Music phrase length
Quarter note level	4*τ	4*4*τ (16τ)
Eighth note level	8*τ	4*8*τ (32τ)
Sixteenth note level	16*τ	4*16*τ (64τ)

* The smallest note length detected is τ

If the summary is shorter than the required length, the preceding or succeeding music phrases will be integrated into the selected chorus to satisfy the length requirement for the summary. According to music theory [27], one music phrase is usually four bars in length. Therefore, the rhythm information is useful for aligning music phrases such that the generated summary has a smooth melody. For example, with the assumption that time-signature of an input song is 4/4, if the smallest note length we detected is τ, the length of the music phrase can be calculated according to the different note level of this smallest note. Table 2 lists the different music phrase length calculation schemes corresponding to three different note levels to which the smallest note commonly belongs in popular songs.

Figure 5 illustrates how to include the music phrases anterior or posterior to the selected chorus to get the desired length of the final summary. Similarly, in order to make the music and visual content synchronal, the visual content is selected according to the time line of music part selected as the summary.

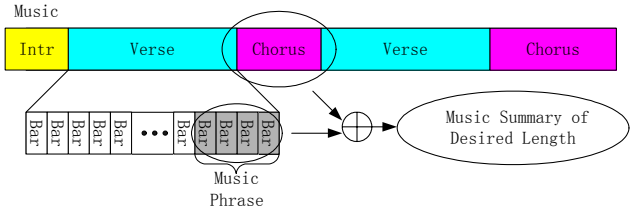


Figure 5: Music selection to meet the desired length

6. EXPERIMENTS & EVALUATION

The experiments include the accuracy of chorus detection, shot classification and lyrics detection. The evaluation is to conduct user study to evaluate the quality of music video summaries.

We use 20 popular English music videos (the details are listed in Table 3) for the experiments and evaluation. The lengths of the music videos range from 2m00s to 4m44s. All the music tracks in music videos are sampled at 44.1 kHz with 16 bits per sample and stereo format.

Table 3.The dataset of music video

No.	MTV Song Name	Duration	Singer/Band Name
1.	Five Hundred Miles	3m32s	Peter,Paul& Mary
2.	Only You	2m40s	Harry Connick Jr
3.	Yesterday Once More	4m02s	Carpenters
4.	Sound of Silence	3m05s	Paul Simon
5.	Let It Be	3m48s	Beatles
6.	Cloud Number 9	3m45s	Bryan Adams
7	Love Me Tender	2m48s	Elvis Presley
8	Evergreen Tree	2m40s	Cliff Richards
9	The End of the World	2m39s	Skeeter Davis
10	When A Child Is Born	2m34s	Richard Johns
11.	My Love	3m53s	Westlife
12.	25 Minutes	4m20s	MLTR
13.	You Are Still the One	3m33s	Shania Twain
14.	Here I Am	4m44s	Bryan Adams
15.	Daniel	3m54s	Elton John
16.	Take Me to Your Heart	3m58s	MLTR
17	Hero	4m20s	Mariah Carey
18	Island Girl	3m43s	Elton John
19	Fantasy	4m03s	Mariah Carey
20	Come On Over	2m52s	Shania Twain

6.1 Chorus Detection

We extracted music tracks from all music videos to detect choruses in these music tracks. We use chorus detection accuracy formulated in Equation (3) to evaluate the result of detected chorus.

$$chorus\ \ detection\ \ accuracy = \frac{No.\ of\ \ chorus\ \ correctly\ \ detected}{Total\ \ No.\ of\ \ Chorus} \times 100\% \quad (3)$$

Here correctly detected chorus means the start and ending boundaries of the detected chorus should fall into ± 2 seconds of the start and the ending boundaries of the original chorus in music. The results of chorus detection accuracy of each music video are tabulated in Table 4.

In Table 4, TnoC denotes the total number of the chorus contained in a music track, and CDA denotes the chorus detection accuracy. From Table 4, we can see that our music structure analysis algorithm is able to at least correctly detect one chorus in a music track. It partially validates our music-visual-text alignment algorithm, which takes chorus as a stable element and aligns the corresponding visual part to it.

Table 4. Chorus detection accuracy

No.	TnoC	CDA %	No.	TnoC	CDA %
1	3	66.67	11	4	100
2	2	100	12	4	100
3	4	75	13	3	33.33
4	3	66.67	14	4	75.00
5	6	66.67	15	2	50.00
6	3	33.33	16	2	100.00
7	3	100	17	2	50.00
8	2	33.33	18	3	33.33
9	3	66.67	19	3	66.67
10	3	66.67	20	4	75.00

6.2 Shot Classification

The purpose of shot classification is to find the close-up face shots in the video track and use these shots to construct the music video summary. We investigate the accuracy of SVM classifier which is used for shot classification. We need to select training data for SVM before we use it for classification. In order to make training results statistically significant, training data should be sufficient and cover various music videos. We use 1000 shots (500 are Close-Up Face and 500 are Non-Face) manually selected from various music videos as the training set to train SVM (the previous 20 English music videos are not included in this training set), and we employ the radial basic function (RBF) with Gaussian kernel as the kernel function in SVM training. The radial basic function (RBF) with Gaussian kernel can be defined as following:

$$K(x, x_i) = \exp(-|x - x_i|^2 / c) \quad (4)$$

where x denotes the vector drawn from the input space, x_i represents training vectors (i=1…n), and c is the width of a Gaussian kernel. In our experiment, we set c=2.

After training SVM, we use it as the classifier to classify all shots detected in our test set which contains 20 popular English music videos listed in Table 3. Totally there are 532 Close-Up Face shots and 763 Non-Face shots. The classification results are show in Table 5.

Table5.**Shot classification results**

Shot Class	Total	Correct	False Alarm	Recall	Precision
Close-Up Face	532	491	58	92.29%	89.43%
Non-Face	763	667	83	87.42%	88.93%

6.3 Lyrics Detection

The lyrics detection accuracy is defined in Equation (5):

$$Lyrics\ detection\ accuracy = \frac{No.\ of\ lyrics\ correctly\ detected}{Total\ No.\ of\ lyrics} \times 100\% \quad (5)$$

Here the correctly detected lyrics means the actual number of lyrics detected using our proposed method and the start and ending boundaries of the detected lyrics should fall into ±*2* seconds of the start and the ending boundaries of the original lyrics chorus in the music video. Due to the limitation of shot detection algorithm, some lyrics in certain shot cannot be detected if one shot contains more than two different lyrics. In our experiment, the lyrics detection accuracy from music videos is 97.6%.

6.4 Subjective User Study

Since there is no objective measure available to evaluate the quality of a music video summary, we adopt a subjective user study [28] to evaluate the performance of our music video summarization method. The basic idea of this user study is to use appropriate attributes to access the users' perception of the proposed method. The following attributes are considered for music video summary.

a. *Clarity*: This pertains to the clearness and comprehensibility of the music video summary.

b. *Conciseness*: This pertains to the terseness of the music video summary and how much of the music video summary captures the essence of the music video.

c. *Coherence*: This pertains to the consistency and natural drift of the segments in the music video summary.

We have evaluated our proposed method on a test set of 20 music videos. The length of the summary for each sample is set to 20s.

We invited 12 participants with music experience to evaluate the music video summaries. Most of the participants are students of the School of Computing in National University of Singapore. Their ages ranged from 18 to 30 year old. Before the tests, the subjects could watch each testing sample for as many times as needed till he/she grasped the theme of the sample. Then the subjects watched summaries generated from test samples and rated the summaries in four categories (Clarity, and Conciseness, Coherence, and Overall Quality) on a scale of 1-5, corresponding to the worst and best respectively. We employ the overall quality of the video as an attribute to evaluate a summary because it pertains to the general perception of the users to the video summaries. The average grade of summaries from all subjects is the final grade. In order to make comparison, we also asked the subjects to rate the summaries generated using our previous summarization method [11] and the summaries manually generated by two music experts from our institute (they are not subjects). In order to avoid potentially biased evaluation results, we present the music video summaries generated by different methods in a random order so that the subjects do not know which method had been used to generate each summary before they rate them. Table 6 shows the average scores of the user evaluation to the summaries generated using proposed method, our previous method, and manual method respectively.

From the test results, it can be seen that the summaries using proposed method performed quite well, especially in the coherence attribute, compared with our previous method. This is because our previous method just focused on the most frequent music segments which may occur in different places in a song, and the summary is created by concatenating these segments together. As a result, the discontinuity will happen either in the summarized segment beginning from the middle of music phrase or in the boundary of two different summarized segments. These two problems are avoided in our proposed method as we made the music video summary based on the continuous music phrases.

It also can be seen that proposed method is comparable to the manual summarization method. It is quite surprising that the proposed method performs better than the manual summarization in terms of coherence. This is probably because the summaries generated by the proposed method contain the entire music phrases while music experts may sometimes break in the middle the music phrases at the beginning or ending of the music video summary for the purpose of not exceeding the desired length.

Table 6 Results of user evaluation

	Clarity	**Conciseness**	**Coherence**	**Overall Quality**
Proposed Method	4.6	4.4	4.8	4.6
Previous Method	4.2	4.0	3.9	4.1
Manual Method	4.7	4.4	4.6	4.7

7. CONCLUSION

We have presented a novel approach to create music video summary using audio-video-text analysis and alignment. The proposed approach has combined music knowledge with low-level feature analysis to provide a powerful tool for music video analysis and summarization. The user study has illustrated the created summaries using proposed approach are promising and superior to existing methods.

The future work will focus on three directions. Firstly, we need to improve and refine the algorithms for chorus detection, shot classification and lyrics detection. Secondly, we will test our approach on a large scale of music videos. Thirdly, we will explore more applications based on proposed approach. For example, some potential applications can be music video semantic indexing and retrieval, singer identification, etc.

8. REFERENCES

[1] Logan B and Chu S , Music Summarization Using Key Phrases, In *Proc. IEEE International Conference on Audio, Speech and Signal Processing*, Istanbul ,Turkey, 2000, vol.2, II749 - II752.

[2] Xu C, Zhu Y and Tian Q, Automatic music summarization based on temporal, spectral and cepstral features, In *Proc. IEEE International Conference on Multimedia and Explore*, Lausanne, Switzerland, 2002, 117-120.

[3] Lu L, and Zhang H , Automated Extraction of Music Snippets, In *Proc. ACM International Conference on Multimedia*, Berkeley, CA, 2003, 140-147.

[4] Bartsch M A and Wakefield G H, To Catch a Chorus: Using Chroma-based Representations for Audio Thumbnailing, In *Proc. Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA),* New Paltz, New York , 2001, 15 – 18.

[5] Cooper M and Foote J, Automatic Music Summarization via Similarity Analysis, In *Proc. International Conference on Music Information Retrieval*, Paris, France, 2002, 81-85.

[6] Chai W and Vercoe B, Music Thumbnailing via Structural Analysis, In *Proc. ACM international conference on Multimedia ,* Berkeley, CA, 2003, 223-226.

[7] Yow, D., Yeo, B.L., Yeung, M., and Liu, G., Analysis and presentation of soccer highlights from digital video, In *Proc. of Asian Conference on Computer Vision*, Singapore, 1995, vol. II, 499-503.

[8] Gong, Y., Liu, X., and Hua, W., Creating motion video summaries with partial audio-visual alignment**,** In *Proc. of IEEE International Conference on Multimedia and Expo*, Lausanne, Switzerland, 2002, vol.1, 285–288.

[9] Foote, J., Cooper, M., and Girgensohn, A., Creating Music videos using automatic media analysis. In *Proc. ACM International Conference on Multimedia*, Juan-les-Pins, France, 2002, 553-560.

[10] Pfeiffer, S., Lienhart, R., Fischer, S., and Effelsberg, W., Abstracting digital movies automatically, *Journal of Visual Communication and Image Representation*, 7, 4, (1996), 345-353.

[11] Shao, X., Xu, C., and Kankanhalli, M.S., Automatically generating summaries for musical video, In *Proceedings of IEEE International Conference on Image Processing,* Barcelona, Spain, 2003, Vol.2, 547-550.

[12] Agnihotri, L., Dimitrova, N., Kender, J., and Zimmerman, J., Music videos miner, In *Proc. of the ACM International Conference on Multimedia,* Berkeley, CA, 2003, 442-443.

[13] Agnihotri, L., Dimitrova, N., and Kender, J., Design and evaluation of a music video summarization system, In *Proc. of IEEE International Conference on Multimedia and Expo*, 2004, Taibei, Taiwan.

[14] Ten Minute Master No 18: Song Structure. *MUSIC TECH magazine.* www.musictechmag.co.uk (Oct. 2003), 62 – 63.

[15] Goto M A, Chorus-section detecting method for musical audio signals, In *Proc. IEEE International Conference on Acoustics Speech and Signal Processing*, Hong Kong, 6-10 April, 2003.

[16] Dannenberg R B and Hu N, Discovering music structure in audio recording, In *Proc. 2nd International Conference on Music and Artificial Intelligence*, Scotland, UK, 2002, 43-57.

[17] Maddage C. N, Xu.C, Kankanhalli M.S , Shao X, Content-based music structure analysis with the applications to music semantic understanding, *In Proc. ACM International Conference on Multimedia*, New York, NY, 2004, 112-119.

[18] Rossing, T.D., Moore, F.R., and Wheeler, P.A., *Science of Sound.* Addison Wesley, 3rd Edition 2001.

[19] Navarro, G. A guided tour to approximate string matching, *ACM Computing Surveys*, Vol.33, No.1, March 2001, 31-88.

[20] Sheh, A., and Ellis, D.P.W., Chord Segmentation and Recognition using EM-Trained Hidden Markov Models. In *Proc. ISMIR* 2003.

[21] Young S *et al*., *The HTK Book*, Dept. of Engineering, University of Cambridge, Version 3.2, 2002.

[22] Deller, J. R., Hansen, J.H.L., and Proakis, H. J. G. *Discrete-Time Processing of Speech Signals*, IEEE Press (1999).

[23] Collobert, R., and Bengio, S. SVMTorch: Support Vector Machines for Large-Scale Regression Problems. *Journal of Machine Learning Research*. Vol 1, 2001, 143-160.

[24] Smoliar, S., and Zhang, H., Content-based video indexing and retrieval, *IEEE Multimedia*, vol. 1, pp. 62–72, 1994.

[25] Li S. Z., Zhu L. , Zhang Z. Q. , Blake A. , Zhang H. J. and Shum H. , "Statistical learning of multi-view face detection", *European Conference on Computer Vision,* Denmark, May 2002.

[26] Hua X. S., , Chen X. R., Liu W., Zhang H. J. , Automatic location of text in video frames. *3rd International Workshop on Multimedia Information Retrieval*, Ottawa, Canada, 2001.

[27] *Rudiments and Theory of Music.* The Associated Board of the Royal Schools of Music, 14 Bedford Square, London, WC1B 3JG, 1949.

[28] John .P.C. , Virginia A. Diehl and Kent L.N, Development of an instrument measuring user satisfaction of the human-computer interface,*Proceedings of SIGCHI'88*, pp.213-218 , New York,1988