Frame Wise Video Editing based on Audio-Visual Continuity

Tatsunori Hirai *

Abstract

In this paper, we describe a method for freely changing the length of a video clip, leaving its content almost unchanged, by removing video frames considering both audio and video transitions. In a video clip that contains many video frames, there are less important frames that only extend the length of the clip. Taking the continuity of audio and video frames into account, the method enables us to change the length of a video clip by removing or inserting frames that do not significantly affect the content. Our method can be used to change the length of a clip without changing the playback speed. Subjective experimental results demonstrate the effectiveness of our method in preserving the video content. We also present a video authoring application using the proposed method.

Key words: Video processing, Video summarization, Video stretching, Authoring support

Introduction

Video content exists everywhere, on televisions, Blu-rays, the Internet, and personal devices. Such video content are already completed content, and people watch them as it was created. Therefore, the time consumed watching video content is driven by the video clip length. On the other hand, spare time to watch such video clips is limited, and the spare-time and the clip length rarely match. Here, we propose a method that enables changing the length of a video clip rather than adjusting the watching time to the length by adding flexibility to the clip length by frame wise editing.

In this paper, we propose a novel video editing method that edit video clip in frame-by-frame manner, which can be applied to watching video and authoring video. The idea of our video editing method is inspired

Figure 1  The main idea of our method.

* Komazawa University
Author contact: thirai@komazawa-u.ac.jp
by video summarization method. The objective of video summarization is to shorten a video clip while preserving its essence. It is to maximize the amount of information per unit time. The summarization amount and an understandable degree are trade-off. Our method achieves this objective by thinning out video frames. Focusing on audio and video frame transitions, our method removes frames that do not affect the content in a frame-wise manner. Since video clip is composed of a collection of similar frames, video clip of \(N\) frames is almost the same as video clip of \(N-1\) frames. We choose the one frame focusing on the transition of content and continue the thinning out process until the clip reaches the desired length. Thus, the length of a clip can be reduced, while the content and continuity of audio and video frames are preserved. Figure 1 shows the main idea of our approach.

Our method is not only to shorten the length of a video clip. Using the same logic, our method can also stretch the length of a video clip by inserting video frames that do not significantly affect the content. In this manner, the length of a video clip can be flexible.

We propose frame-wise video editing that enables both video summarization and video stretching. Video summarization method is an important technology for an efficient video browsing experience. Although video stretching method does not benefit video browsing experience, there is usefulness in the process of video authoring. Our method supports video authoring by enabling changes to the video clip length. In the video editing process, an editor must use trial and error to determine shot change points to fit the resulting video clip to a fixed time. Some editors try to edit videos on a frame-by-frame basis. This is one of the most troublesome work in creating final version of output video clip. If the length of the video fits the fixed time perfectly, editors can take more time in the other editing process. Our method helps such editors by enabling changes to the length of a video clip without requiring changes to its content. We aim to help people edit video clips by increasing the flexibility of a video clip in the aspect of the clip length.

This paper presents the detail of our method and subjective experiment which shows the effectiveness of the method. We also present a video authoring application using the method.

Related Work

Video summarization is one of the most important and well-established video processing research topics. There are many video summarization approaches in various domains [1].

In particular, summarization methods for sports video are actively pursued. For example, Kawamura et al. [2] proposed a method for summarizing racquet sports video by extracting rally scenes that are important for understanding the game. Tjondronegoro et al. [3] proposed a method for detecting and summarizing sports video highlights from whistle sounds, audience cheers, and textual information.

There are many other video summarization method which do not limit to sports video. DeMenthon et al. [4] proposed a method for summarizing videos by expressing a video as a curve in multi-dimensional feature space and simplifying the curve. Their method partly enables frame-wise thinning out of video frames but does not consider audio information which also takes important role in video content. Smith et al. [5] designed an indicator for thinning out video segments by integrating scene changes, camera motions, object recognition, and keywords in the audio. This method eliminates video segments that are less important for understanding the content. However, the elimination of video segments results in video sequence discontinuity, and the original content cannot be preserved. Therefore, the summarized result is equivalent to that of edited videos that do not preserve the original information.

Most video summarization methods either extract important video segments or eliminate unimportant segments. However, such methods extract or eliminate video segments using their own criteria, preventing users from knowing what types of scenes they missed. Therefore, a fast-forwarding video browsing approach that does not eliminate information but enables efficient video watching has been proposed. Kurihara et al. [6]
proposed a fast-forwarding method that changes the playback speed in speech and non-speech segments to enable understanding of the speech content while fast forwarding a video. This method does not take visual information into account and is thus not well suited to videos, such as sports videos, in which motion is important. Moreover, this method changes the playback speed at a fixed rate regardless of the content. There are methods for changing the playback speed depending on factors such as motions in a clip [7], important events [8], or user preferences [9]. However, those methods do not take audio information into account and are thus not suitable for videos, such as videos including conversation, in which audio plays an important role. On the other hand, there is a method for changing the speed of audio media to enhance its intelligibility. Imai et al. [10] proposed a speech rate conversion method, for slowing the speech speed by shrinking the silent sections in audio media. In this manner, the entire audio length does not change but intelligibility improves. This approach might be useful in changing the length of audio content but our goal is to change the length of video clips including both video and audio. In summary, fast forwarding and speed conversion techniques for video and audio have been proposed separately, and method considering both modalities simultaneously has not been proposed.

In this paper, we present a method for removing and inserting video frames considering both video and audio, while preserving the content. Although the video frame removal and insertion can be regarded as a video playback speed conversion, we focus on the cost of frame thinning rather than on changing the playback speed on the basis of specific criteria. What we focus on to thin out video frames is the continuity of video frames with respect to video and audio. Because the time continuity in video frames play a great role, it is important to preserve the continuity between frames. Therefore, our method removes a video frame that does not affect transition, thereby preserving continuity in the resulting frames. At this point, before removing a frame that does not affect transition, we also consider the continuity after removing the frame. In other words, we consider both the transition of a video frame and continuity between anteroposterior video frames. Consequently, our method can preserve more continuity than by focusing only on the transition of corresponding video frames. Another difference between our method and fast forwarding is that our result can be used in video authoring applications. This is because our resulting video clips retain both their content and their continuity, thereby ensuring the naturalness of the results while preserving the content. To distinguish our method from fast forwarding, we call our video frame thinning out method a “frame-wise video summarization” method.

The advantage of video summarization and fast-forwarding methods is primarily related to the efficiency of browsing video content. However, our goal is not restricted to efficient browsing and includes video authoring support as well. In the video editing workflow, the original video clip material is cut and pasted by an editor to match a fixed time. The lengths of material clips and the fixed time rarely match without editing them. However, the editing process is time consuming because there are many possibilities for cutting and pasting, and the editor must consider the length and the content at the same time, which requires substantial trial and error. There are methods for resizing the length of background music to fit with the corresponding video. Sato et al. [11] proposed a method for reconstructing bars of music and resizing them to match a user’s preference, and Wenner et al. [12] proposed a method for resizing music by calculating natural jumping points within a musical piece. These methods help a video editor by providing background music with flexibility. Using flexible music reduces a restriction in video editing because the editor does not need to consider the background music length. This type of flexibility is also helpful for the editor if we can change the video length as well as the background music.

Berthouzoz et al. [13] proposed a system for seamlessly cutting and pasting interview video clips that an editor wants to use. The quality of edited video is sufficiently natural that people who watch the interview video cannot detect the cut point easily. However, the target of this method is limited. This method can only be applied to a video that contains minimal transitions between camera and objects, such as interview video,
therefore the method focuses on supporting video editing of interview video. We aim to support video authoring more extensively by achieving flexible video editing, frame-wise video summarization and stretching.

Frame-Wise Video Summarization

Digital video media display multiple images called frames from moment to moment and express motion and dynamics. The general video displays 30 frames per second (FPS), with an interval between frames of approximately 0.033 second. The number of FPS is called the frame rate, which is 29.97 FPS for general video content of NTSC, such as that on TV broadcasting in the U.S. Actually, video media do not capture images uninterruptedly but the human cognitive system interpolates the visual information between the frames such that a human perceives the content as consecutive visual information. This phenomenon is called apparent movement. The frame rate corresponds to the temporal resolution, and it need not always be 30 FPS or more. For example, most movie films and animations are 24 FPS, and some video content uploaded on the Web is 15 FPS to reduce the file size. Although a higher frame rate achieves smoother motion and a lower rate results in jumpy motion, the human brain interpolates the motion as long as a minimum frame rate is maintained.

Removing one frame per consecutive frame pair in a 30 FPS video clip doubles the playback speed. Consequently, the amount of visual information received from the video clip will be equivalent to that of a video recorded at 15 FPS. Here we focus on transitions in a video clip. For example, if all objects appearing in a video remains stationary during capture, thinning out of video frames has no effect. While the amount of visual information that can be received from the clip does not change, the clip length becomes shorter. In the extreme, keeping only one frame and eliminating all the other frames will not change the visual information as long as the objects are stationary, but only the time required for its playback. Thus, transition in video frames corresponds to the amount of visual information, and lower transition frames have less affect after thinning out is performed.

The main idea of this research is to focus on transition of video frames and to remove low transition frames in a frame-wise manner in order to reduce the length of a video clip, while preserving the amount of information (i.e., remaining the content). The same can be said with regard to the audio portion. If the same sound occurs in consecutive frames, removing some frames has little effect on the content. Using this feature, we propose a method that thins out video frames considering audio-visual transition.

Frame-wise Thinning out based on a Visual Transition

Human eyesight is sensitive to sudden change, hence, temporal continuity between video frames is important. The significance of video frame thinning depends on transitions. If the frame to be removed contains drastic change, the resulting video clip will cause discomfort compared with one resulting from removing a stable frame. For example, if the object to be captured by a camera is moving fast and can be captured for only a few frames, one frame plays an important role in the aspect of the visual information. Removing such a frame destroys object motion continuity. In contrast, if an object is moving very slowly, the moving distance during one frame is sufficiently short such that the loss of continuity from removing it is lesser than that from removing a frame containing a fast-moving object.

Consequently, we focus on transitions that can handle object motion comprehensively. In particular, we use the sum of squared differences (SSD) corresponding to the difference between adjacent video frames. The SSD value $s(t)$ at frame $t$ can be calculated as

$$s(t) = \sum_{y=1}^{\text{height}} \sum_{x=1}^{\text{width}} (f(x, y) - f_{t-1}(x, y))^2$$ (1)
where $f(x, y)$ is the pixel value at the coordinates $(x, y)$ in frame $t$. Figure 2 shows the SSD transition in a video of a person walking past a stable camera twice. In this graph, the SSD value increases sharply at an important event in the video (viz., a person walking by).

In this type of video, frames with stable objects will be preferentially removed when the lower value frames are thinned out. Thus, thinning out frames with low SSD values shortens a video clip, while preserving its content.

To some extent, this method is effective. However, only thinning out the frames with low SSD values sometimes creates new discontinuities. Therefore, new SSD values to be inserted after thinning out must also be considered. To take new SSD values into account, we set the thinning cost $C_{\text{video}}$ as the summation of SSD values of the frame and the new SSD value of anteroposterior video frames. Here the cost $C_{\text{video}}$ is normalized to enable combining it with an audio thinning cost later. Our method thins out video frames according to the cost, preserving the continuity of the entire video clip as much as possible. To achieve this aim, SSD values of removed frames must be recalculated after the first round of thinning. At this point, the new SSD value of thinned out frames is not normalized, creating a difference between normalized costs. To avoid this difference, we retain the value for the normalizing process and apply the same parameter to rescale the new SSD value. We prohibit thinning out the first or last frame of a video clip to ensure that the thinning cost can always be calculated.

Avidan et al. [14] proposed an image resizing (retargeting) method called seam carving. This method searches for a path that is not important and thins out pixels in a row- or column-wise manner. Rubinstein et al. [15] expanded this method and proposed an improved seam carving that considers insertion cost in addition to thinning cost. The improved seam carving can be applied to video retargeting. Our method can be considered as a form of seam carving applied to entire video frames for video summarization instead of video retargeting.

**Frame-wise Thinning out based on an Audio Transition**

The sampling rate of audio is generally 44,100 or 48,000 samples per second, which is very different from...
that of video frames. Generally, audio waveform is analyzed in segment called audio frame. We designed an audio frame for analysis to adjust the time step between audio and video. When the video frame rate is $r$, we set the audio frame length to $2/r$ seconds and the step length for the audio frame to $1/r$ seconds. Hence, the audio and video time steps can be synchronized.

As an audio feature expressing audio continuity, we use spectral flux, which represents local temporal transition of the audio spectrum. It takes a high feature value at the point when an audio transition occurs (e.g., sound onset or offset). The reason we chose spectrum-based feature is because thinning out of raw audio signal sample ends up with pitch change. Therefore, we focused on continuity of audio spectrum rather than raw audio signal. We extract spectral flux from an audio part of a video clip using MIRtoolbox 1.5, an audio analysis tool developed by Lartillot et al. [16].

![Transition of spectral flux](image)

**Figure 3** Transition of spectral flux value.

Figure 3 shows the transition of spectral flux for an audio sample that includes speech and hand claps. The audio sample is recorded in an indoor environment where particular sounds are not observed other than the speech and claps. In the graph in fig. 3, the spectral flux value reflects the auditory events. From this result, we can say that it is possible to detect a section in which an audio event occurs by focusing on the spectral flux value.

Thinning out audio frames with a low spectral flux value shortens the audio without losing the audio event content. Here we use audio frame thinning cost $C_{audio}$ and thin out frames exactly as in thinning out based on a visual transition, considering the new insertion cost in addition to the spectral flux value of a frame. This enables remaining audio continuity after the thinning out process. An audio can be shortened by calculating the spectral flux value per thinning round.

*Frame-wise Thinning out based on an Audio-Visual Transition*

We combine video and audio frame thinning. Our video and audio frame thinning consider visual and audio continuity, respectively. To consider audio-visual continuity, we design the audio-visual thinning cost $C(t)$ for removing frame $t$ as
Frame Wise Video Editing based on Audio-Visual Continuity

\[ C(t) = \alpha C_{\text{video}}(t) + (1-\alpha)C_{\text{audio}}(t) \] (2)

here the parameter \( \alpha \) is a weight for audio-visual balance. When \( \alpha = 0.5 \), audio and visual continuity is considered equally. Both \( C_{\text{video}} \) and \( C_{\text{audio}} \) are normalized in advance to have mean zero and variance one in order to achieve uniformity. Frame-wise video summarization can be achieved by removing a video frame with minimum cost using Eq. (2).

Figure 4 shows the transition of audio-visual thinning out cost calculated for the video with a person walking by used in fig. 2. In this video, the visual event is a human walking across the video twice, and the audio event is the sound of footsteps. Before and after the visually observable walking event, there are sections with only the sound of footsteps can be heard. In such a section, the cost remains high, indicating that the cost reflects not only visual but also audio events. The parameter \( \alpha \) is an important parameter to add weight to audio and video, which is highly dependent to the content. For example, auditory cost is not as important in action scenes as it is in conversation scenes. At the current implementation, this parameter is manually set by a user. Automatic optimization of \( \alpha \) for each video content is our future work.

**Video Stretching via Frame Insertion**

We also propose a method for stretching a video clip to provide it with flexibility. Video stretching can be achieved by using the same cost (Eq. (2)) as in the video summarization. Inserting video frames with low cost at the same point can stretch a video clip while preserving its content. There are two differences between video summarization and stretching. One is that we do not consider the new inserted cost after frame insertion because the inserted cost incurred from the insertion of the same frame is zero. Another difference is that we do not re-calculate the cost for each frame insertion round because the video frame with minimum cost remains the minimum despite frame insertion. Instead of re-calculating cost, we insert video frames in the ascending order of cost. Hence, we begin video frame insertion with the least-cost frame and then insert the frame with the second-least cost.

However, inserting video frames in an ascending order leads to disregarding content. If we double the
length of a video clip by inserting video frames in an ascending order, all the frames will be doubled and the playback speed will be halved. Although it is possible to preserve the clip content with slow playback, the motion in the clip will be slowed and changes its meaning. Therefore, we set a maximum percentage of video frames to insert and do not use all the frames for insertion. For example, if the maximum percentage is set to 50%, only half of the video frames will be used for frame insertion. This value corresponds to the percentage of frames that do not play important roles for content. The value depends on the content, and it must be set differently for each video clip. However, for the current implementation, we leave the setting of this value to the user. Automatic optimization of the percentage of insertion frame is our future work.

**Subjective Experiment**

We performed a subjective experiment to evaluate the results of our video summarization and stretching method with respect to the naturalness of a processed video.

We compared our results with video clips produced using playback speed conversion, i.e. fast forwarding and slow playback. Seven subjects were asked to watch video clips generated using our proposed method and the playback speed conversion method and to determine which were more natural. The subjects are all male and 6 of them are aged 21 to 23 years old students and one subject was 50s. The subjects watched a total of 20 video clips and scored them from 1 to 5 as shown in Table 1, by comparing the clips generated using each method. If the score was close to “3: the video is natural,” summarization or stretching succeeded in changing the length of the video clip without noticeable artifacts. We used two types of video clips and five clip lengths, with shrinking rates of 50%, 75%, 100%, 125%, and 150%. The video clips included “No.1: video of fireworks with narration included” and “No.2: speech video with gesture included.” The original length of clip no.1 was 22 seconds, and clip no.2 was 25 seconds.

For each video clip, the subjects watched five clips with different lengths produced using our method, followed by five clips produced using the comparison method. The clips were displayed to the subjects in random clip length order, and subjects were asked to assign each score independently, regardless of the previous clip displayed. The video clips included the original clip, with a shrinking rate of 100%, but the subjects were not told which clip it was. Although the 100% shrinking rate clips are exactly the same in the proposed and comparison methods, we displayed both clips in order to check the variance of the evaluation scores. We set the audio-visual balance parameter $\alpha$ in Eq. (2) to 0.5 and the maximum percentage for frame insertion to 20%.

Table 2 shows the evaluation score for each video clip. The score 3.00 indicates naturalness, and scores closer to that value reflect better results. The score in tab. 1 is the average of seven subjects. From this result, we can conclude that our proposed method scores more closely to 3.00 and is thereby shown to be more natural than the speed conversion method for shrinking rates of 75%, 125%, and 150%. However, the result
for the 50% shrinking rate is the same as that for the speed conversion method. This is because thinning out 50% of frames is excessive and causes the removal of meaningful frames. From the questionnaire provided to the subjects, we learned that discontinuity, particularly in speech video, was the major cause of unnaturalness. This experiment handled only two types of video clips and four shrinking rates, and we are planning to further evaluate the relation between shrinking rate and naturalness to understand the effective range of our method. In addition, the audio-visual balance parameter $\alpha$, and maximum percentage of insertion frames should be optimized in the future.

**Video Authoring Application using the Proposed Method**

Here we present a video authoring application which uses our frame-wise video editing method. Figure 5 shows the screen capture of the application. It looks same as an ordinary video editing software. The difference is that our application uses frame-wise video editing method so that the length of each clip can be freely changed. This avoids a video editor from a troublesome work which requires a lot of trial and error in fixing editing point of each scene.

By dragging and dropping a video clip that a user wants to add for the final edition of an output video, the clip can be added to the timeline. The user connects video clips as a video editor does in ordinary video editing software. In a final step of video editing, it is necessary to adjust the length of the output video. In this step, all the thing that the user need to do is to drag the back of the clip and adjust the length. Thus the length can be freely changed while preserving its content.

In addition, there is automatic time adjustment function. When the user set a fixed time of the output video and press “Auto” button, system automatically adjust the length of all clips and automatically make the final output into the user specified length. In “Auto” button, system equivalently thin out or insert video frames from each clip.

**Applications of the Proposed Method**

There are many possible application that can be realized using the proposed method. In this section, we introduce some further applications of our proposed method that we are exploring beyond summarization and stretching.

In our video summarization method, video frames are thinned out in an ascending order. Here we consider the thinning out of video frames in a descending order and the elimination of moving objects in a video clip. It is possible to eliminate moving objects in a video clip by thinning out video frames at a high cost if all the
other objects are stable. However, if the camera is moving or the scene is complex and many objects are mov-
ing, it is difficult to eliminate frames using this approach. In such a condition, pixel-wise elimination rather
than frame-wise elimination is more effective, and we are investigating that possibility.

Another possible application is audio-visual synchronization for a music video clip.

In previous sections, we considered audio and visual continuity simultaneously as in Eq. (2). In contrast,
if we consider them separately, it may be possible to synchronize music and video. Sato et al. [11] proposed
a method to synchronize music to a video clip by recomposing a musical piece. Similarly, our method has the
potential of synchronizing music and video by resizing video. When a user specifies the points of video and
audio to be synchronized, our method enables shrinking and stretching the video frames between those
points. Because synchronizing audio and video is difficult in video editing process, it is effective for a user
to support audio-visual synchronization at the frame level.

Our video summarization method is effective for content-based video retrieval as well. Since the method
shortens the video while preserving the content, content-based video retrieval can be more efficient and fast
without considering verbose video frames. In addition, data storage can be used more efficiently if the con-
tent can be reversible. We can discuss the reversibility of our summarization method by referring the theory
of information compression. Our method is related to the theory of information compression, and we will
further explore the relation in the future research.

**Figure 5** The video authoring application.
(The film used in this figure is “Roman Holiday”, 1953, public domain)
Conclusion

In this paper, we have presented a novel method for summarizing and stretching video clips by focusing on frame-wise transition and continuity. The subjective experimental results demonstrated effectiveness in a reasonable manner.

Our method thins out video frames regardless of the meaning of silence or stability.

Therefore, if a silence is meaningful, our method may have a negative effect. The method summarizes video considering the content. However, if the silence or stable state is the content, our method eliminates the content. It is a limitation of our method that it is difficult in the current implementation to consider the semantics of the content. Most but not all the semantics included in audio-visual transition can be considered using our method. A further limitation is that our method is not well suited to apply to a video that includes music. Because every audio frame in music has meaning, our method must not remove any frames if there is background music. In such a case, resizing music using the method proposed by Wenner et al. [12] may be effective, or perhaps, simply thinning out the frames equally would produce a better result.

There are limits to the number of frames that can be removed, while still preserving the content. We will explore the limit further and enable adaptive and automatic setting of parameters such as $\alpha$ in Eq. (2) by quantitatively defining an amount of information for a video clip. The idea of our method is similar to that of video compression. We are planning to apply those theories used in video compression technique to achieve effective frame thinning.

The time we can assign to watching the video content is limited. We can watch the video content efficiently by scheduling what and when to watch. However, we would like to enjoy freely without letting such an annoying schedule restrain our daily lives. Instead of adjusting our schedules to the video content, the video content should be adjusted to our schedules. Our method has the possibility of enabling a user to watch a video regardless of how much time he or she has available for video watching, and we will further explore that possibility.

References


