Computers & Graphics 36 (2012) 241–249

Contents lists available at SciVerse ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

Applications of Geometry Processing

Visual storylines: Semantic visualization of movie sequence

Tao Chen [a,*], Aidong Lu [b], Shi-Min Hu [a]

[a] TNList, Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China
[b] Department of Computer Science, University of North Carolina at Charlotte, USA

ARTICLE INFO

Article history:
Received 20 August 2011
Received in revised form
13 February 2012
Accepted 16 February 2012
Available online 27 February 2012

Keywords:
Video summarization
Video visualization
Geometric layout

ABSTRACT

This paper presents a video summarization approach that automatically extracts and visualizes movie storylines in a static image for the purposes of efficient representation and quick overview. A new type of video visualization, *Visual Storylines*, is designed to summarize video storylines in an image composition while preserving the style of the original videos. This is achieved with a series of video analysis, image synthesis, relationship quantification and geometric layout optimization techniques. Specifically, we analyze the video contents and quantify the video story unit relationships automatically through clustering video shots according to both the visual and audio data. A multi-level storyline visualization method then organizes and synthesizes a suitable amount of representative information, including both the locations and interested objects and characters, with the assistance of arrows, according to the relationships between the video story units and the temporal structure of the video sequence. Several results have demonstrated that our approach is able to abstract the main storylines of professionally edited video such as commercial movies and TV series, though some semantic key clues might be missed in the summarization. Preliminary user studies have been performed to evaluate our approach, and the results show that our approach can be used to assist viewers to understand video contents when they are familiar with the context of the video or when a text synopsis is provided.

1. Introduction

In recent years, both the quality and quantity of digital videos have been increasing impressively with the development of visual media technology. A vast amount of movies, TV programs and home videos are being produced every year for various entertainment or education purposes. Under such circumstances, video summarization techniques are desperately required for video digestion and filtering processes by providing viewers with an efficient tool to understand video storylines without watching entire video sequences.

Currently, existing video summarization methods mainly focus on news programs or home videos, which usually contain simple spatiotemporal structures and straightforward storylines. Those methods cannot successfully handle professionally edited movies and TV programs, where directors tend to use more sophisticated screen techniques. For example, a movie may have two or several storylines alternately depicted in an irregular sequence. Also, technically, many existing methods summarize a video sequence with collections of key frames or regions of interest (ROIs) without high-level information such as location and sequence of events. We believe that this information should be carefully embedded in the video analysis and summarization process.

Our goal is to present a visually pleasing and informative way to summarize the storylines of a movie sequence in one static image. There are many advantages of using a still image to summarize a video sequence [1–5] because an image is generally much smaller and easier for viewers to understand. The methods that use still images to visualize video clips can be classified into two types according to their applications. One is to visualize a short video clip, mainly focusing on one or two characters and their spatial motion, e.g., [6,7]; the other is to visualize a related longer video clip that is capable of telling a semantic story, e.g., [1–3,5]. Our method belongs to the latter. A common problem with this type of method is that due to the highly compact form and losses of information (e.g., audio, text and motion), it is nearly impossible for viewers to extract the underlying stories without being aware of the context of the video or appropriate text descriptions. Even with this information provided, using previous methods, it is still very difficult to recover sophisticated storylines because there is a lack of analysis of scene relations. We believe that by properly considering vision and audio features and carefully designing the form of visualization, such a semantically difficult problem can be tackled to some extent for many professionally edited movies and TV programs.

In this paper, we present a new *Visual Storylines* method to assist viewers to understand important video contents by

* Corresponding author.
E-mail address: chent@cg.cs.tsinghua.edu.cn (T. Chen).

0097-8493/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cag.2012.02.010

revealing essential information about video story units and their relationships. Our approach can produce a concise and visually pleasing representation of video sequences, which highlights most of the important video contents and preserve the balanced coverage of original sequences. Accompanying the original text description of videos (plots), these results assist viewers to better understand the video topics before actually watching the videos. They can also help the viewers to understand video contents when they are familiar with the context of the video. Specifically, we first present an automatic video analysis method to extract possible video storylines by clustering video shots according to both visual and audio data. We also design a multi-level visual storyline method to visualize both abstract story relationships and important video segments. We have designed and performed preliminary user studies to evaluate our approach and have collected some encouraging results.

The main contribution of our approach is a series of automatic video analysis, image synthesis and relationship quantification and visualization methods. We have seamlessly integrated techniques from different fields to produce a compact summary of video storylines. Both the results and evaluation demonstrate that our approach can highlight important video contents and storylines from professionally edited movies and TV programs better than some previous methods.

The remainder of this paper is organized as follows. We first summarize related video summarization, analysis and representation approaches in Section 2. Section 3 presents our automatic approach to analyzing video structures and extracting storylines. Section 4 describes our multi-level storyline visualization method, which significantly enriches abstract storylines through a series of video analysis and image synthesis methods. We describe and discuss our user studies to evaluate our approach and provide experimental results in Section 5. Finally, Section 6 concludes the paper.

2. Related work

Our work is closely related to video summarization, which has been an important research topic in the fields of Computer Vision, Multimedia and Graphics. Video summarization approaches often focus on content summarization [8]. A good survey of both dynamic and static video summarization methods has been provided by Huet and Merialdo [9], who have also presented a generic summarization approach using Maximum Recollection Principle. More recently, Correa et al. [6] proposed dynamic video narratives, which depicted motions of one or several actors over time. Barnes et al. [10] presented *Video Tapestries*, which summarized a video in the form of a multiscale image through which users can interactively view the summarization of different scales with continuous temporal zoom. These two methods represent the state of the art of dynamic summarization.

In this paper, we concentrate on approaches of static visual representations, which require the synthesis of image segments extracted from a video sequence. For example, the video booklet system [1] proposed by Hua et al. selected a set of thumbnails from the original video and printed them out on a predefined set of templates. Although this approach achieved a variety of forms, the layout of the predefined booklet templates was usually not compact. Stained-glass visualization [2] was another kind of highly condensed video summary technique, in which selected key frames with interesting areas were packed and visualized using irregular shapes, such as in a stained-glass window. In contrast to this approach, the approach presented in this paper synthesizes images and information collected from video sequences to produce smooth transitions between images or image ROIs. Yeung et al. presented a pictorial summary of video content [3] by arranging video posters in a timeline, which summarized the dramatic incident in each story unit. Ma and Zhang [4] presented a video snapshot approach that not only analyzed the video structure for representative images but also used visualization techniques to provide an efficient pictorial summary of the video. These two approaches showed that key-frame-based representative images were insufficient to recover important relations from a storyline. Among all forms of video representations, Video Collage [5] was the first to provide a seamlessly integrated result. In contrast to this technique, our approach reveals information about locations and relations between interesting objects and preserves important storylines.

This paper is also related to the analysis of video scene structure and detection of visual attention. For example, Rui et al. [11] and Yeung et al. [12] both presented methods to group video shots and used a finite state machine to incorporate audio cues for scene change detection. Because these approaches are either bottom-up or top-down methods, they cannot easily achieve a global optimization result. Ngo et al. [13] solved this problem by adopting normalized cut on a graph model of video shots. Our work improves on their method by relying on audio similarity between shots. Zhai and Shah [14] provided a method for visual attention detection using both spatial and temporal cues. Daniel and Chen [15] visualized video sequences with volume visualization techniques. Goldman et al. [7] presented a schematic storyboard for visualizing a short video sequence and provided a variety of visual languages to describe motion in video shots. Although this method was not suitable for exploring relations between scenes in a long video sequence, their definition of visual languages inspired our work.

Our *Visual Storylines* approach first clusters video shots according to both visual and audio data to form semantic video segments, which we call sub-stories. The storylines are revealed by their similarities. Next, the program calculates and collects the most important background, foreground and character information into composite sub-story presenters. A multi-level storyline visualization method that optimizes information layout is designed to visualize both abstract story relationships and important video segments. The details are introduced in the following two sections.

3. Automatic storyline extraction

It is necessary to extract the storylines from a video sequence before generating any type of video summaries. Automatic approaches are desirable, especially for tasks such as video previewing, where no user interaction is allowed. We achieve an automatic storyline extraction method by segmenting a video into multiple sets of shot sequences and determining their relationships. Our approach considers both visual and audio features to achieve a meaningful storyline extraction.

Our storylines are defined as important paths in a weighted, undirected graph of sub-stories (video segments). To generate a meaningful storyline, it is crucial to segment a video into a suitable number of video segments, which are sets of video shots. A shot is a continuous strip of motion picture film that runs for an uninterrupted period of time. Because shots are generally filmed with a single camera, a long video sequence may contain a large number of short video shots. These video shots can assist us to understand video contents; however, they do not reflect the semantic segmentation of the original videos well. Therefore, they should be clustered as meaningful segments, which are called video events.

Automatic shot clustering is a very challenging problem [11–13] because in many movie sequences several characters talk alternately in similar scenes or scenes may change greatly

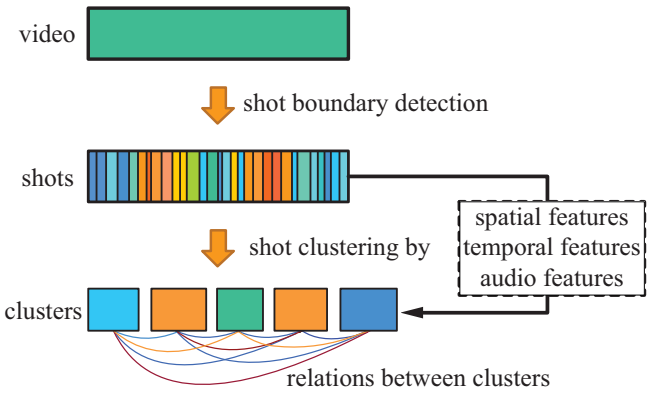

Fig. 1. Our video shot clustering algorithm combines both visual and audio features to generate a meaningful storyline.

while a character is giving a speech. Previously, Rui et al. [11] and Yeung et al. [12] presented methods to group video shots by using thresholds to decide whether a shot should belong to an existing group. Because a single threshold is usually not robust enough for a whole sequence, these approaches may lead to over segmentation. Ngo et al. [13] used normalized cut to cluster the shots. In their work, the similarities between shots contain color and temporal information. However, none of the existing approaches are sufficiently robust for movie sequences.

We believe that combining both the visual and audio features of a video sequence can improve the results of shot clustering, leading to more meaningful segmentations for visual storylines. Fig. 1 illustrates our video shot clustering algorithm, where we integrate several important video features to cluster video shots and determine their relationships. Although audio features have been utilized in video analysis [16–18], we are the first to use them as features for graph modeling of video shot clustering.

Specifically, our shot clustering algorithm integrates the following visual and audio features: shot color similarity, shot audio similarity, and temporal attraction between shots. Shots are obtained using the approach proposed in [19], which can handle complex scene transitions, such as hard cut, fade and dissolve. The color similarity and temporal attraction are defined the same way as in [11], and the shot audio similarity is defined as an MFCC feature distance [20]. The Mel-frequency cepstral coefficients (MFCC) derived from a signal of a short audio clip approximate the human auditory system's response more closely than the linearly spaced frequency bands used in the normal cepstrum. MFCC can be used as a good measure of audio similarity for speaker diarisation. For each shot, we calculate the mean vector and covariance matrix of all the MFCC feature vectors in the shot;, the audio similarity between two shots is then defined as one minus the Mahalanobis distance between the shots.

Thus, we define the overall similarity between two shots x and y as follows:

$$ShtSim_{x,y} = Attr_{x,y} \times (W_C * SimC_{x,y} + W_A * SimA_{x,y}),$$

where $Attr_{x,y}$ is the temporal attraction between shots and W_C and W_A are the weights for color and audio measures $SimC_{x,y}$ and $SimA_{x,y}$. Because we know that greater similarity is more reliable, we define the weights as follows:

$$W_C = \frac{\omega_c}{\omega_c + \omega_a}, \quad W_A = \frac{\omega_a}{\omega_c + \omega_a},$$

where

$$\omega_c(x,y) = \begin{cases} e^{\lambda_c(x,y)} & \text{if } SimC_{x,y} > \mu_c + \frac{\sigma_c}{2}, \\ e^{-1} & \text{otherwise,} \end{cases}$$

$$\omega_a(x,y) = \begin{cases} e^{\lambda_a(x,y)} & \text{if } SimA_{x,y} > \mu_a + \frac{\sigma_a}{2}, \\ e^{-1} & \text{otherwise,} \end{cases}$$

$$\lambda_c(x,y) = -\frac{(1-SimC_{x,y})^2}{\left(1-\mu_c-\frac{\sigma_c}{2}\right)^2},$$

$$\lambda_a(x,y) = -\frac{(1-SimA_{x,y})^2}{\left(1-\mu_a-\frac{\sigma_a}{2}\right)^2}.$$

μ_c and σ_c are the mean and variance of color similarities, μ_a and σ_a are the mean and variance of audio similarities.

After calculating pairwise similarities, we build a weighted, undirected graph and adopt normalized cut to cluster the shots. An adaptive threshold is used for the termination of recursive partition, as in [13]. The incorporation of audio features improves the clustering result. For example, when clustering the movie sequence in Fig. 5(a), the second sub-story (represented in the upper right corner of the result image) has an outdoor/indoor change; using similarity, as defined in [13], will improperly partition it into two clusters due to the significant change in appearance, but because the same character is speaking, the audio similarity is relatively large. Therefore, our similarity measure provides a more semantic clustering.

We use each cluster to represent a sub-story. We denote clusters as $S = \{Sub\text{-}story_1, Sub\text{-}story_2, \ldots, Sub\text{-}story_m\}$. Those sub-stories are usually not independent from each other, especially in professionally edited movies. Some sub-stories may be strongly related even though they are not adjacent. For example, some movies often contain more than one story thread and different sub-stories occur at different locations synchronously. To demonstrate this, filmmakers may cut two stories into multiple sub-stories and depict them alternately. To capture this important information, we calculate the relations between two sub-stories. They are defined as follows:

$$ER_{i,j} = W_C * Avg_{x \in E_i, y \in E_j} SimC_{x,y} + W_A * Avg_{x \in E_i, y \in E_j} SimA_{x,y}.$$

To handle the situation that some shots are mis-clustered, we empirically throw the first and last five shots in a sub-story when calculating the average described above. We further check all the shot clustering results that are generated. The video events with larger similarity values are viewed as being more related. We will integrate the relation information during the generation process of visual storylines in Section 4.

In all five video sequences, we manually labeled 43 story cuts; the shot clustering with audio similarity provided 33 correct story cuts, while the number of cuts was reduced to 21 without audio similarity ("correct" means a story cut is detected within a distance of five shots from ground truth). This proves that the use of audio similarity greatly increases the accuracy of shot clustering.

4. Generation of visual storylines

With the extracted storylines, we further visualize a movie sequence in a new type of static visualization. This is achieved using a multi-level visual storyline approach, which selects and synthesizes important story segments according to their relationships in a storyline. Our approach also integrates image and information synthesis techniques to produce both semantic and visual appealing results.

Previously, static summarization of a video was usually achieved by finding a key frame in a sequence [3,1,4] or a ROI (region of interest) in the key frame [2,5]. Obviously, one single key frame or

ROI is insufficient to represent many important pieces of information about a story, such as time, location, characters and occurrence. Simply "stacking" all the images together, like "VideoCollage", is still not enough to reveal a storyline or roles of different characters due to the lack of relationships and emphasis.

Our design of the visual storyline approach is based on the observation that complicated stories usually consist of multiple simple stories, while simple stories only involve a few key factors, such as characters and locations. Generally, while commercial movies contain multiple sub-stories, the major storylines are rather straightforward. Therefore, we can design a visual storyline as an automatic poster to visualize various movies.

To handle complicated storylines, such as commercial movies, a multi-level approach is necessary to visualize various movies due to the following reasons.

- First, because one still visualization can only provide a limited amount of information, we need to control the details of visual storylines such that they are presented at a suitable scale for viewers to observe.
- Second, it is important to describe major events and main characters instead of details that are only relevant to some short sub-stories. Therefore, we always need to include the top levels of storylines and generate visual summaries at different scales.

We have developed several methods to synthesize images and information collected from a video sequence. The following section first introduces how to extract essential image segments by selecting background and foreground key elements, then describes our design of a sub-story presenter, storyline layout and storyline visualization.

4.1. Background image selection

This step aims to find a frame that can best describe the location (or background) of a sub-story. Typically, it should be the image with the largest scene in the video sequence. Although detecting the scale from a single image is still a very difficult problem in the areas of computer vision and machine learning, we can simplify this problem by making the following assumptions based on our observations, which are summarized as follows:

Shots containing larger-scale scenes usually have smoother temporal and spatial optical flow fields. This is because these background scenes are usually captured by static or slow-moving cameras. In this case, if the optical flow fields indicate a zooming-in or zooming-out transition, the first or the last frame should be selected, respectively, because they represent the scenes of the largest scale.

We can remove the frames with good responses to face detection to avoid the violation of characters' feature shots because they are not likely to be background scenes.

Very often, a shot containing this kind of frame appears at the beginning of a video sequence; this is called the establishing shot. The establishing shots mostly occur within the first three shots of a sub-story.

Therefore, we can detect the image with the largest scale automatically using additional information collected from a video sequence. We run a dense optical flow calculation [21] and face detection algorithms [22] through the video sequence and discard shots with stable face detection response. The remaining shots are sorted in the ascending order *optical flow discontinuity* defined as follows.

Optical flow discontinuity for $Shot_i$ from a video event (i is shot index in the video event):

$$Discont(i) = \frac{1}{numFrm_i} * \sum_{j=1}^{numFrm_i - 1} (DscS_j + DscT_j)$$

Here, $numFrm_i$ is the frame number of $Shot_i$, $DscS_j$ is The spatial optical flow discontinuity of frame j, and $DscT_j$ is the temporal optical flow discontinuity between frame j and $j+1$. They are measured in the same way as in [21].

After sorting by this discontinuity value, a proper frame from each of the top ten shots is selected (according to zooming order) as the background candidate of a video sequence. To achieve this, we run a camera zoom detection for the shot according to [23] and choose the frame with the smallest zoom value. We sequentially check the selected ten frames; if any of them belongs to the first three (in temporal) shots of the video sequence, it will be chosen as the background image of the sub-story because it has a great chance of being the establishing shot. Otherwise, we simply choose the one that ranks first. A selected background image is demonstrated in the top-left corner of Fig. 2.

4.2. Foreground ROIs selection

There are three kinds of objects that are good candidates for foreground regions of interest (ROIs) in drawing visual attention:

Character faces: Characters often play major roles in many commercial movies, where more than half of the frames contain human characters.

Objects whose motion is distinguished from that of the background often draw temporal attention.

Objects with high contrast against the background often draw spatial attention.

Therefore, we propose a method that integrates the detection algorithms of human faces and spatiotemporal attention. We reuse the per-frame face detection result from Section 4.1 and only preserve those stably detected in temporal space (detected in continuous five frames). Then, we define a face-aware spatiotemporal saliency map for each frame as follows:

$$Sal(I) = \kappa_T \times SalT(I) + \kappa_S \times SalS(I) + \kappa_F \times SalF(I).$$

Here, the spatiotemporal terms are exactly the same as those described in [14], though a more advanced approach such as [24] could also be used. We add the face detection result to the saliency map with the last factor. Specifically, for pixels falling in the detected face regions, we set their saliency value $SalF(I)$ as 1, if the pixels fall outside the detected face regions, their saliency value is set to zero. κ_F is the weight for $SalF(I)$. Without violating the dynamic model fusion (which means the weights are dynamically changed with the statistic value of $SalT(I)$), we set $\kappa_F = \kappa_S$.

Next, we automatically select ROIs for each video sequence. To prevent duplicate object selection, we create the restriction that only one frame can be used for ROI selection in each shot. This frame is the one with the largest saliency value in the shot. Then, for a newly selected ROI, we check the difference between its local histogram and those of the existing ROIs. If it is smaller than a certain threshold (0.1 Chi-square distance), only the ROI with the larger saliency value will be preserved. Those ROIs are then sorted by their saliency value per pixel. Different kinds of selected ROIs are shown in Fig. 2.

4.3. Sub-story presenter

We design a method to generate a static poster to present simple sub-stories. Our approach is inspired by popular commercial movie

Fig. 2. Synthesis process of the sub-story presenter.

Fig. 3. Storyline geometric layout. The right figure is a synthesized visual storyline for a video sequence lasting 30 min, which is clustered into 10 sub-stories. For limited spaces, the figures on the left show six sub-story presenters.

posters, which usually have a large stylized background and feature character portraits, along with several of the most representative (relatively smaller) film shots. This layered representation not only induces the user to focus on the most important information, but also provides state-of-the-art visual appearance.

Our sub-story presenter contains at least three layers. The bottom layer is the background image frame extracted in Section 4.1. The layer next to the bottom contains ROIs with no face detected, while other layers are composed of other ROIs extracted in Section 4.2. The higher layer contains ROIs of higher order, i.e., with higher saliency values. We use a greedy algorithm to calculate the layout, as illustrated in Fig. 2.

We start from the bottom layer, i.e., the background image. We initialize the global saliency map with the saliency map of the background image. Then, we add each layer overlapping on the presenter, from the lowest layer to the top layer. For each layer, we add the ROIs from the one with the highest saliency value to that with the lowest. For each ROI, we first resize it according to its saliency degree, then search for a position that minimizes the global saliency value of the presenter covered by the ROI. After adding a new ROI, the global saliency map is updated by replacing the covered region's saliency with newly added ROI's.

In this process, we use a threshold φ, which we call the level-of-detail controller, to control the number of ROIs presented. In other words, when adding a new ROI, every object in the presenter (including the background image) must preserve at least φ portion of its original saliency value in the global saliency map (the detected face region has the exception that it should never be covered to prevent displaying half faces). When this is violated, the ROI with the lowest saliency value will be removed from the presenter, and the layout will be recalculated. With this "LOD" control, when the video sequence we represent becomes more complex, we can ensure each presented part still provides sufficient information.

After adding each layer, we use graph cut to solve the labeling problem, followed by α-poisson image blending [25]. To emphasize the importance of foreground objects, we stylize each layer as shown in Fig. 2. We compute the average hue value of the background image and use this value to tint each layer; the lower layers will be tinted to greater degrees. Fig. 3 shows six basic event presenters synthesized by our approach. They are able to

represent the most important information of the video event, such as locations and characters, and also preserve the original video style.

4.4. Storyline geometric layout

Now, the remaining problem is how to arrange sub-story presenters in the final visual storylines to reveal their relationships. We prefer to preserve the style of movie posters so that visual storylines are intuitive for general users to understand. Here, we present an automatic algorithm that generates a geometric storyline layout by utilizing all the information extracted from video analysis.

Given n sub-story presenters $\{R_1,R_2,\ldots,R_n\}$ for n sub-stories and their relations and a canvas of size $l \times m$, we first resize all the sub-story presenters:

$$size(R_i) = \max\left(0.25, \frac{L(R_i)}{L_{max}}\right) \times \frac{l*m}{1.5n},$$

where $L(R_i)$ is the length (in frame) of the $Sub\text{-}story_i$ and L_{max} is the maximal duration of all the sub-stories. Let (x_i,y_i) denote the shift vector of the sub-story presenters R_i on the canvas; then, we minimize the following energy function:

$$E = E_{ovl} + w_{sal}*E_{sal} + w_{rela}*E_{rela} + w_{time}*E_{time}.$$

The overlay term $E_{ovl} = -A_{ovl}$ is the negative of the overlay area of all the basic event presenters on the canvas; saliency cost E_{sal} is the negative saliency value of the constructed saliency map; The relation term is defined as follows:

$$E_{rela} = \sum_{i=0}^{n} \sum_{j=i+1}^{i+3} \left(Dist(i,j) - \frac{\sqrt{lm}(ER_{max} - ER_{i,j})}{ER_{max} - ER_{min}} \right)^2,$$

where $ER_{i,j}$,ER_{max} and ER_{min} are relationships measured between $Sub\text{-}story_i$ and $Sub\text{-}story_j$, maximal relationship and minimal relationship, respectively. $Dist(i,j)$ is the distance between the centers of two basic event presenters. This term attempts to position sub-story presenters with stronger relationships closer to each other along the x coordinate; the temporal order term is defined as

$$E_{time} = \sum_{i=0}^{n-1} \delta_i, \quad \text{where } \delta_i = \begin{cases} 0 & \text{if } y_i + \epsilon < y_{i+1} < y_i + h_i - \epsilon, \\ 1 & \text{otherwise,} \end{cases}$$

h_i is the height of resized R_i, and $\epsilon = 30$. This term attempts to position sub-story presenters with respect to temporal order along the y coordinate while preserving some overlapping. We set $w_{sal} = 0.15, w_{rela} = 0.1, w_{time} = 0.1$.

Minimizing the energy function above will maximize the overlay area of all basic event presenters, which visualize temporal order along the y coordinate and visualize relations along the x coordinate. We use a heuristic approach to solve this layout problem. We start from the first sub-story presenter; when each new presenter is put in, the algorithm calculates the position that minimizes the current energy function. Because this method cannot ensure that all pixels are covered, we can choose those obsolete ROIs from adjacent basic event presenters to fill the hole. One alternative is to adopt the layout optimization method described in [26]. Then, the overlapped region will be labeled by graph cut and α-poisson image blending. Because overlapping may violate the LOD control, it is necessary to recalculate the layout for sub-story presenters. Fig. 3 shows the events layout and the LOD control effect. As shown, when the represented video sequence becomes complicated, our results will not be cluttered, as observed using other methods, and still provide essential video information.

4.5. Storyline visualization

The final visual storylines are enriched with a sequence of arrow shapes to represent key storylines. This is achieved by building a storyline graph, which uses video sub-stories as nodes. If the relationship between two adjacent video sub-stories in the visual storylines is greater than a certain threshold, we add an edge between the two. After traversing all the nodes, circles will be cut off at the edge between the two nodes with the largest temporal distance. Thus, each branch in this acyclic graph represents a storyline. We add an arrow at the intersection between any two connected sub-story presenters, with the restriction that no ROI should be covered. The directions of arrows illustrating the same storyline are calculated according to a B-spline, which is generated by connecting all the arrow centers and saliency-weighted centers of the sub-story presenters involved in this storyline. This produces the smoothest and most natural illustration of the storyline. The arrow bottom is reduced to disappear in the previous event to emphasize the directions of storylines. Different storylines are distinguished by arrow color.

5. Experiments and evaluations

5.1. Experimental results

Fig. 5 shows the sample results of visual storylines. Their computation times on a Core 2 Duo 2.0-GHz machine and LOD thresholds (φ) are shown in Table 1.

The video sequence used in Fig. 5(a) is a classic movie clip that features two scenes (different locations and characters) alternately. Our approach successfully extracts the two storylines. Note that the movie title in the result is a manually added ROI, which replaces the correspondence part in Fig. 3.

Fig. 5(b) and (c) visualizes two fast-paced TV programs. They both feature multiple storylines, which is popular among modern TV series. Our approach extracts the main storylines for each program. Although one storyline (threaded by the pink arrows) in (c) has merged two semantic scenes together due to very similar scenes, our most recent user studies show that viewers can still understand the plot from our visual stories. It should be noted that a user can adjust the LOD threshold φ to generate multi-level results. The multi-level visual storylines generated by different thresholds in Fig. 5(b) and (c) are demonstrated in the supplementary file.

Fig. 5(d) visualizes a movie clip that alternately features two groups of characters, which ultimately meet each other. Our visual storylines reveal this important feature using two merging storylines.

In summary, our approach for generating visual storylines is suitable for visualizing movie scenes with salient visual attributes, such as those of a desert, meadow, sky and other outdoor scenes, or indoor scenes with artistic stylized illumination. Changes in characters may also help the system distinguish different scenes.

One failed case is shown in Fig. 4. The commercial movie *Lock, Stock, and Two Smoking Barrels* is famous for its fast scene changes

Table 1
Computation times for each representation result.

Video clip	Length (min)	Time cost (min)	φ (%)
Fig. 5(a): Star Wars	30	125	40
Fig. 5(b): Lost	20	80	60
Fig. 5(c): Heroes	22	90	70
Fig. 5(d): Crazy	15	62	40

Fig. 4. A failed case of our system when representing 25 min video sequence from the commercial movie *Lock, Stock, and Two Smoking Barrels*. User studies show this summary cannot reveal the true storylines of the movie sequence.

and techniques of expressing multiple storylines. In this movie, most of the scenes in those different storylines are indoor scenes with indistinguishable color models. Moreover, character groups in different scenes share complex interactions. Therefore, our approach cannot accurately extract the storylines. The extracted storylines are captured with respect to the temporal order of the sub-story presenter.

5.2. User studies and discussion

We have designed three user studies to evaluate our approach. The first user study is designed to check the aesthetic measure and representative measure relative to other methods.

Twenty subjects are invited for this user study, including 14 graduate students and six undergraduate students (majoring in computer science, architecture and art) who are unaware of our system. Four kinds of video summaries (Booklet, Pictorial, Video Collage and Visual Storylines) are created for sequences shown in Fig. 5. After watching the video sequences, users have been asked to answer the following questions on a scale from 1 (definitely no) to 5 (definitely yes), as used in [25,5]. Here, we list our questions and provide the average scores and standard deviations for each method after their names.

- Are you satisfied with this summary in general?
  Visual Storylines (4.10, 0.62), Video Collage (3.50, 0.67), Pictorial (2.30, 0.90), Booklet (2.45, 0.97)
- Do you believe that this result can represent the whole video sequence?
  Visual Storylines (4.20, 0.68), Video Collage (3.65, 0.65), Pictorial (3.30, 0.64), Booklet (3.15, 0.57)
- Do you believe this presentation is compact?
  Visual Storylines (4.00, 0.71), Video Collage (3.90, 0.70), Pictorial (2.60, 0.49), Booklet (2.35, 0.57)
- Would you like to use this result as a poster of the video?
  Visual Storylines(4.65, 0.48), Video Collage (3.70, 0.71), Pictorial (1.4), Booklet (3.1)
- Do you believe that this presentation produces the correct storylines?
  Visual Storylines (4.85, 0.36), Video Collage (2.25, 0.70), Pictorial (2.5, 0.74), Booklet (1.75, 0.83)

The results demonstrate that our approach achieves the highest scores in all categories; therefore, it is the most representative and visually appealing summary among these four approaches. It also shows that the Visual Storylines method performs better in extracting and visualizing video storylines.

The other two user studies are designed to determine whether our results can help users quickly grasp major storylines without watching a video. It should be noted that it is generally very difficult for someone to understand the semantic storylines of a movie or TV program from a single image without any context. In the second user study, subjects are asked to watch some video clips related to the test video. Fifteen more subjects are invited and confirm that they have not seen any of the movies or TV programs shown in Figs. 4 and 5 before. Ten of them were assigned to a "test group", the other five were assigned to an "evaluation group". We showed the test group the five movies/TV programs used in our paper but skipped the parts that were used to generate the video summaries. The evaluation group was allowed to watch the full movies or TV programs. Then, in the test group, half of the subjects were provided with five summaries generated by our method, while the other half were provided with five summaries generated by "Video Collage" (because it is the most competitive in the first user study). Then, these ten subjects were asked to write text summaries for the five video clips they did not watch. These text summaries were shown to the evaluation group and evaluated on a scale from 1 (very bad summaries) to 5 (very good summaries). The average score for each video evaluated by different methods is shown in Table 2.

In the third user study, we invited ten more subjects. They were asked to read text synopses for the five videos tested in our paper. They were also provided with the summaries (Visual Storylines for half, Video Collage for the other half). Then, they were asked to circle the corresponding regions in the summaries for some previously marked keywords in the synopses, which included locations, objects and character names. We manually checked the correctly circled regions and list the results in Table 2.

Table 2 shows that when viewers know the context of the video, for example the main characters and their relationships and the preceding and succeeding stories, they can easily understand the stories using our visual storylines. It also shows that viewers can quickly perceive correct connections between the text synopses and our summaries. It should be noted that the two statistical results of *Lock, Stock, and Two Smoking Barrels* are lower than 3 and lower than 60%.

The user studies reveal two potential applications for our approach. First, if a viewer misses an episode of a TV show or a part of a movie, visual storylines can be synthesized to help the viewer quickly grasp the missing information. Second, when presenting our results together with the text synopsis of a video, viewers obtain a visual impression of the story described in the synopsis. Therefore, our automatically generated results can be easily integrated into TV guides, newspapers, movie review magazines and movie websites as illustrations of text synopses.

In addition to comparing with other methods used to generate static summaries for long video sequences, we would also like to discuss and compare our methods with state-of-the-art video summarization methods. Because [6,7] mainly focus on one or two characters and their respective spatial motion, their summarization is very suitable for visualizing one or several shots. However, they cannot deal with long video sequences like we can using our method. However, if we incorporate their static representations of character motion into our sub-story representation, the visual storylines can be

Fig. 5. Visual storylines of (a) a 30-min sequence from the commercial movie *Star Wars: Attack of the Clones*, (b) a 20-min sequence from the TV program *Lost*, (c) a 30-min sequence from the TV program *Heroes*, (d) a 20-min sequence from the commercial movie *The Gods Must Be Crazy 2*.

Table 2
The statistical results for user study 2 and 3.

User studies	Star Wars	Lost	Heroes	Crazy	Lock
User study 2 (Scores)					
Our method	4.52	3.28	4.08	4.12	2.76
Video Collage	2.64	2.08	1.84	3.48	1.64
User study 3 (Correct/All)					
Our method	26.6/28	34.6/39	21.2/27	34.4/36	21/37
Video Collage	20/28	16.4/39	13.2/27	26.6/36	17.4/37

more compact and less visually repetitive. The *Video Tapestries* [10] provides a static summarization method similar to ours, though their shot layout is purely sequential. However, when a multiscale summarization is interactively viewed by a user, it can provide more information than that afforded by our method. However, our static result is more suitable for traditional paper media.

Here, we discuss some limitations of our approach and possible improvements. As the failure case indicated, our approach generates limited results for indistinguishable scenes. In addition, because it selects important candidates according to low-level features such as visual saliency and frequency, the visualization may still miss crucial semantic information. For example, the coffin, which plays an important role in the resulting collage of *Lost* sequence, is barely recognizable. Another issue regarding our approach is that even with LOD control, our results may still suffer from repetitively showing main characters, as in other methods. One solution, as mentioned above, is to adopt the character motion representations described in [6,7] or generate motion photography in static image similar to that described in [27]. We may also try to recognize repeating characters or foreground objects from their appearance and segmentation silhouettes by the boundary band map matching method introduced in [28]. A recently reported candid portrait selection approach [29], which involves learning a model from subjective annotation, could also help us to find more visually appealing character candidates. The α-poisson image blending we adopted to construct the visualizations sometimes generates undesirable cross-fading, which could be resolved using recently developed blending methods such as hybrid blending [30] or environment-sensitive cloning [31]. Finally, our preliminary user study could also be improved. The questions in the first user study are too general and subjective, which may bias the evaluation due to the various degrees of understanding experienced by different individuals when viewing video sequences. The second user study is too complex in design and may bias the results according to the writing skills of individuals.

6. Conclusion

This paper presents a multi-level visual storyline approach to abstract and synthesize important video information into succinct

still images. Our approach generates visually appealing summaries by designing and integrating techniques of automatic video analysis and image and information synthesis. We have also designed and performed preliminary user studies to evaluate our approach and compare with several classical video summary methods. The evaluation results demonstrate that our visual storylines can extract storylines better than previous approaches.

The techniques of video visualization and summary are important additions in handling the enormous volume of digital videos because they can help viewers better grasp the main storylines of a video quickly without watching the entire video sequence, especially when they are familiar with the context of the video or a text synopsis is provided. With the efficiency provided by video visualization techniques, we believe that these methods can also be used to assist other video operations, such as browsing and documentation for entertainment and educational purposes.

Acknowledgements

This work was supported by the National Basic Research Project of China (Project Number 2011CB302205) and the Natural Science Foundation of China (Project Number 61120106007, 61033012). Aidong Lu was supported by grant DOE DE-FG02-06ER25733.

Appendix A. Supplementary data

Supplementary data associated with this article can be found in the online version at doi:10.1016/j.cag.2012.02.010.

References

[1] Hua XS, Li S, Zhang HJ. Video booklet. In: ICME; 2005. p. 4–5, doi: 10.1109/ICME.2005.1521392.
[2] Chiu P, Girgensohn A, Liu Q. Stained-glass visualization for highly condensed video summaries. In: IEEE international conference on multimedia and expo, vol. 3; 2004. p. 2059–62.
[3] Yeung M, Yeo BL. Video visualization for compact presentation and fast browsing of pictorial content. IEEE Trans Circuits Syst Video Technol 1997; 7(5):771–85. doi:10.1109/76.633496.
[4] Ma YF, Zhang HJ. Video snapshot: a bird view of video sequence. In: Proceedings of the 11th international multimedia modelling conference; 2005. p. 94–101, doi:10.1109/MMMC.2005.71.
[5] Wang T, Mei T, Hua XS, Liu XL, Zhou HQ. Video collage: a novel presentation of video sequence. In: IEEE international conference on multimedia and expo; 2007. p. 1479–82, doi: 10.1109/ICME.2007.4284941.
[6] Correa CD, Ma KL. Dynamic video narratives. ACM Trans Graph 2010; 29:88–9. doi:10.1145/1778765.1778825.
[7] Goldman DB, Curless B, Seitz SM, Salesin D. Schematic storyboarding for video visualization and editing. ACM Trans Graph (Proc SIGGRAPH) 2006; 25(3).
[8] Money AG, Agius H. Video summarisation: a conceptual framework and survey of the state of the art. J Visual Commun Image Represent 2008;19(2): 121–43. doi:10.1016/j.jvcir.2007.04.002.
[9] Huet B, Merialdo B. Automatic video summarization. In: Hammoud RI, editor. Interactive video. Signals and communication technology. Berlin, Heidelberg: Springer; 2006. p. 27–42. ISBN: 978-3-540-33215-2.
[10] Barnes C, Goldman DB, Shechtman E, Finkelstein A. Video tapestries with continuous temporal zoom. ACM Trans Graph 2010;29:89–90. doi:10.1145/1778765.1778826.
[11] Rui Y, Huang TS, Mehrotra S. Exploring video structure beyond the shots. In: Proceedings of IEEE conference on multimedia computing and systems; 1998. p. 237–40.
[12] Yeung M, Yeo BL, Liu B. Extracting story units from long programs for video browsing and navigation. In: Proceedings of the third IEEE international conference on multimedia computing and systems; 1996. p. 296–305, doi:10.1109/MMCS.1996.534991.
[13] Ngo CW, Ma YF, Zhang HJ. Video summarization and scene detection by graph modeling. IEEE Trans Circuits Syst Video Technol 2005;15(2):296–305. doi:10.1109/TCSVT.2004.841694.
[14] Zhai Y, Shah M. Visual attention detection in video sequences using spatiotemporal cues. In: MULTIMEDIA '06: Proceedings of the 14th annual ACM international conference on multimedia. ACM, New York, NY, USA; 2006. p. 815–24. doi: 10.1145/1180639.1180824, ISBN 1-59593-447-2.
[15] Daniel G, Chen M. Video visualization. In: VIS '03: Proceedings of the 14th IEEE visualization 2003 (VIS'03). Washington, DC, USA: IEEE Computer Society; 2003. p. 54–5. doi: 10.1109/VISUAL.2003.1250401, ISBN 0-7695-2030-8.
[16] Wang Y, Liu Z, Huang JC. Multimedia content analysis-using both audio and visual clues. IEEE Signal Process Mag 2000;17(6):12–36. doi:10.1109/79.888862.
[17] Sugano M, Nakajima Y, Yanagihara H. Automated MPEG audio-video summarization and description. In: Proceedings of the IEEE international conference on image processing, vol. 1; 2002. p. I956–9, doi: 10.1109/ICIP.2002.1038186.
[18] He L, Sanocki E, Gupta A, Grudin J. Auto-summarization of audio-video presentations. In: 7th ACM international conference on multimedia; 1999. p. 489–98.
[19] Lienhart RW. Comparison of automatic shot boundary detection algorithms. In: Yeung MM, Yeo BL, Bouman CA, editors. Proceedings of the SPIE, storage and retrieval for image and video databases VII, presented at the society of photo-optical instrumentation engineers (SPIE) conference, vol. 3656; 1998. p. 290–301.
[20] Rabiner L, Schafer R. Digital processing of speech signals. Englewood Cliffs: Prentice Hall; 1978.
[21] Black MJ, Anandan P. The robust estimation of multiple motions: parametric and piecewise-smooth flow fields. Comput Vis Image Underst 1996;63(1): 75–104. doi:10.1006/cviu.1996.0006.
[22] Lienhart R, Maydt J. An extended set of haar-like features for rapid object detection. In: IEEE ICIP 2002, vol. 1; 2002. p. 900–3.
[23] Wang R, Huang T. Fast camera motion analysis in MPEG domain. In: ICIP 99. Proceedings of the 1999 International conference on Image processing, vol. 3; 1999. p. 691–4, doi: 10.1109/ICIP.1999.817204.
[24] Cheng MM, Zhang GX, Mitra NJ, Huang X, Hu SM. Global contrast based salient region detection. In: IEEE CVPR; 2011. p. 409–16.
[25] Rother C, Bordeaux L, Hamadi Y, Blake A. Autocollage. In: SIGGRAPH '06: ACM SIGGRAPH 2006 papers. ACM, New York, NY, USA; 2006. p. 847–52, doi:10.1145/1179352.1141965. ISBN 1-59593-364-6.
[26] Huang H, Zhang L, Zhang HC. Image collage: arcimboldo-like collage using internet images. ACM Trans Graph 2011;30(6).
[27] Teramoto O, Park I, Igarashi T. Interactive motion photography from a single image. Visual Comput 2010;26:1339–48. doi:10.1007/s00371-009-0405-6.
[28] Cheng MM, Zhang FL, Mitra NJ, Huang X, Hu SM. Repfinder: finding approximately repeated scene elements for image editing. ACM Trans Graph 2010;29(4):83:1–8.
[29] Fiss J, Agarwala A, Curless B. Candid portrait selection from video. ACM Trans Graph 2011;30(6).
[30] Chen T, Cheng MM, Tan P, Shamir A, Hu SM. Sketch2photo: internet image montage. ACM Trans Graph 2009;28(5):124:1–10.
[31] Zhang Y, Tong R. Environment-sensitive cloning in images. Visual Comput 2011;27:739–48. doi:10.1007/s00371-011-0583-x.