Visualizing and Analyzing Video Content With Interactive Scalable Maps

Cui-Xia Ma, Yong-Jin Liu, Guozhen Zhao, and Hong-An Wang, *Senior Member, IEEE*

***Abstract*—Visualizing and communicating insights through maps offers an intuitive and familiar way to explore large-scale dynamic relational data. In this paper, we present *VideoMap,* which is a novel approach for presenting and interacting with relational video content by taking advantage of the map metaphor. VideoMap employs a metaphor to visualize video content by elements of a map with the aim of enabling exploration of video content as if reading a map. Video content is visualized in a hierarchal structure from a very large scale to a small scale of finely detailed representation. VideoMap recognizes a small set of sketch gestures for semantic zooming in and out, annotating the map, and automatically completing path navigation. To achieve this, VideoMap synthesizes map-derived visuals and binds them to the underlying data by operating the map with sketch interaction to facilitate interactive exploration. Extensive user studies were conducted to evaluate VideoMap, and the results demonstrated the effectiveness of VideoMap for facilitating the exploration and understanding of large video content.**

***Index Terms*—Interaction, map metaphor, multi-scale representation, video visualization and analysis.**

I. Introduction

THE explosive growth of video resources has accentuated the need for efficient analytical and interactive tools. Interactive analysis of large-scale dynamic video data is used to help people gain insights into finding underlying patterns and relationships hidden in the raw data. Conventional video visualization and analysis techniques mainly focus on low-level feature extraction and video content compression representation, which lack the immediacy of interactive feedback. Very little attention is paid to relationship visualization in video content and multi-scale browsing and interaction [1], [2]. The interaction path representing the evolution of a story and character relationships is important to aid users in understanding and communicating with digital video. Many works have focused on visualizing dynamic relational data, such as social media for music and TV viewing trends [3], streaming text data [4], and Web trends[1]. Maps are one of the typical methods that visualize large-scale dynamic relational data by preserving the mental map to meet user perception [3]. In general, videos consist of a sequence of scenes (indicating the locations where the story takes place) and events (where characters interact with each other), and these entities are represented in a hierarchical structure. At the semantic level, videos can be regarded as a sequence of temporally and causally related story units. The complexity of video content and latent relationships poses great challenges for existing visualization approaches.

Many researchers have focused on summarizing or visualizing the video content and provided solutions to reduce the time needed for understanding videos and to assist viewers in gaining insight and making decisions in a cost-effective manner. Although there has been much work in the area of video summarization and interaction, most of it is based on low-level image/video features and usually focuses on extracting salient frames and creating video summaries, such as video thumbnails [5], panorama excerpts [6], video storyboarding framework [7], video booklet [8], and video skimming [9], [10]. Ma *et al.* [11] proposed sketch annotation and a sketch based interactive video authoring tool with sketch summarization for video content. However, this method is limited to visualizing the content and relationships using a line drawing format [36]. Some existing video sites manually attach labels with clues of main plots to the timelines of videos to satisfy viewers' diverse browsing requirements. In this way, viewers are able to selectively watch part of the video according to their own interests. However, this method provides only limited timeline information, and viewers cannot obtain a comprehensive picture of the entire video content. In particular, most conventional approaches focus on depicting a specific event and did not consider providing an overview of video content and latent relationships.

Maps offer a promising way to visualize relational data with a hierarchical structure, and it is convenient for viewers to interact with a digital map via panning, zooming and sketching freehand lines. Maps also offer an intuitive way to interact with and navigate within a video. Temporal data, such as the events in a video in our application, can be visualized as paths through some space; e.g., temporal data have been previously visualized as 2D

Manuscript received May 9, 2016; revised August 7, 2016; accepted September 21, 2016. Date of publication September 28, 2016; date of current version October 19, 2016. This work was supported in part by the National Key Research and Development Plan under Grant 2016YFB1001200, in part by the Natural Science Foundation of China under Grant U1435220, Grant 61232013, Grant 61661130156, Grant 61322206, and Grant 61272228, and in part by the Royal Society-Newton Advanced Fellowship. The guest editor coordinating the review of this manuscript and approving it for publication was Dr. David Gotz. *(Cui-Xia Ma and Yong-Jin Liu contributed equally to this work.) (Corresponding author: Yong-Jin Liu.)*

C.-X. Ma is with the State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences, Beijing, China, and also with the Engineering Research Center of Digital Media Technology, Ministry of Education, Jinan 100190, China (e-mail: cuixia@iscas.ac.cn).

Y.-J. Liu is with the Tsinghua National Laboratory for Information Science and Technology, Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China (e-mail: liuyongjin@tsinghua.edu.cn).

G. Zhao is with the State Key Laboratory of Brain and Cognitive Science, Institute of Psychology, Chinese Academy of Sciences, Beijing 100101, China (e-mail: zhaogz@psych.ac.cn).

H.-A. Wang is with the State Key Laboratory of Computer Science and the Beijing Key Laboratory of Human-Computer Interaction, Institute of Software, Chinese Academy of Sciences, Beijing 100190, China (e-mail: hongan@iscas.ac.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2016.2614229

[1][Online]. Available: https://ia.net/know-how/ia-trendmap-2007v2

tracks [24]. From the viewpoint of a video, that space consists of scenes. In our study, these scenes are collocated based on their similarity. Characters and events occur in scenes and are visually connected with edges. Putting all of this together takes the form of a map.

Considering the advantages of a map in both visualization and interaction with video data, we propose a novel approach, called *VideoMap*, based on a map metaphor for visualizing video content, which can be used as a tool for video navigation and analysis. We focus on the legible and pleasing layout of various video content including characters, events, scenes and various relationships among them instead of developing techniques for automating the process of extraction. Each type of video data has a corresponding symbol in our pre-established graphics library. Furthermore, we make use of the Katz centrality [32] to quantitatively measure the importance of characters, which determines their sizes in the map. Events with several attributes are visualized as a collection of points in two-dimensional space through multidimensional scaling [Fig. 1(a) and (b)]. In addition to the representation of the original datasets, additional statistical information hidden in the relationship can also be visualized on the map. VideoMap provides different types of roads to represent various relationships of these datasets for viewers [Fig. 1(c)]. In order to reduce visual clutter and maintain the map's clarity as much as possible, the algorithm Simplified Memory Bounded A* [33] is applied during the process of road visualization. Sketch gestures by drawing freeform and editable sketches are provided to support operations on VideoMap.

II. Related Work

Video visualization and analysis can help users browse video faster, facilitating better understanding of video content. Visual analysis of video content is important to reduce the burden of users' understanding and to reuse the content. Benfold and Reid [13] presented a multi-target tracking system that can correct data association errors and fill in gaps of the missed observations by visualizing and performing data association on a sliding frames window in different views. Hu [14] proposed a semantic-based video retrieval framework for visual surveillance. The retrieval framework supports various queries for video analysis including queries of key words, multiple object queries, and queries by sketch. Piciarelli *et al.* [15] presented a trajectory clustering method suited for video surveillance and monitoring systems. The clusters are dynamic because the trajectory data are acquired in real time, without the need for an offline processing step. This method can be used both to give proper feedback to the low-level tracking system and to collect valuable information for the high-level event analysis modules. Klein *et al.* [16] proposed a novel and accurate approach to motion detection for automatic video surveillance systems. This system consists of three modules: a background modelling (BM) module, an alarm trigger (AT) module, and an object extraction (OE) module. In the BM module, a unique two-phase background matching procedure is performed using rapid matching followed by accurate matching in order to produce optimal pixels for the background model. The AT module eliminates the unnecessary examination of the entire background region, allowing the subsequent OE module to process only blocks containing moving objects. These aforementioned works are useful for tracking targets. However, little progress has been made to facilitate relationship analysis between objects. Furthermore, most studies focus on analysing surveillance videos with some specific events in fixed scenes, and they are inappropriate for general movies in which a dramatic story is depicted in variable scenes.

There have also been many studies on video summarization and visualization. Those studies enable users to understand video content and save time when viewing videos. Yueng and Yeo [17] proposed a method to analyse video and build a compact pictorial summary for visual presentation. They condensed video sequence into a few images, each of which summarizes a dramatic incident taking place in a meaningful segment of the video. They proposed a graphic layout pattern according to the relative dominances, and created a set of video posters, which is a compact, visually pleasant and intuitive representation of the story content. Uchihashi [18] presented a method for automatically creating pictorial video summaries that resemble comic books. They computed the relative importance of video segments according to their length and novelty, using image and audio analysis to automatically detect and emphasize meaningful events. The result is a compact and visually pleasing summary that captures semantically important events and is suitable for printing or Web access. These works are more concerned about presenting video content briefly but do not support analysis for video associations or provide interaction with users. Zhu *et al*. [19] presented a novel approach to extract tactical information from the goal event in a soccer broadcast video and present the goal event in a tactical mode to coaches and sports professionals. Parry *et al*. [20] presented a framework for hierarchical event representation and an importance-based selection algorithm for supporting the creation of a video storyboard from a video. They used Snooker video visualization as a case study to demonstrate the concepts and algorithms. Xie *et al*. [21] presented algorithms for parsing the structure of produced soccer programmes. They defined two mutually exclusive states of the game, play and break, based on the rules of soccer, using the dominant colour ratio and motion intensity within a set of hidden Markov models to segment the game into two states and then gathered the playing segmentation to represent a soccer game. Their work focused on the analysis of sport videos. Tapaswi *et al.* [22] presented a novel approach to automatically represent the characters' interactions in TV episodes in the form of storyline. Jing *et al.* [23] proposed a novel method to convert conversation-based video into manga-style layout comics. This method uses a speaker detection technique to extract key frames that contain the portraits of speakers and puts word balloons near the corresponding speaker. An initial layout is optimized to achieve the optimal display. However, this approach applies only to conversation-based video and relies extensively on the subtitles. Videos can also be regarded as a type of temporal series data. Bach *et al.* [24] proposed a generic way to visualise time series data called time curves. They first constructed a distance matrix, which represents the similarity between time points, and then mapped these points to a plane space in the form of a curve

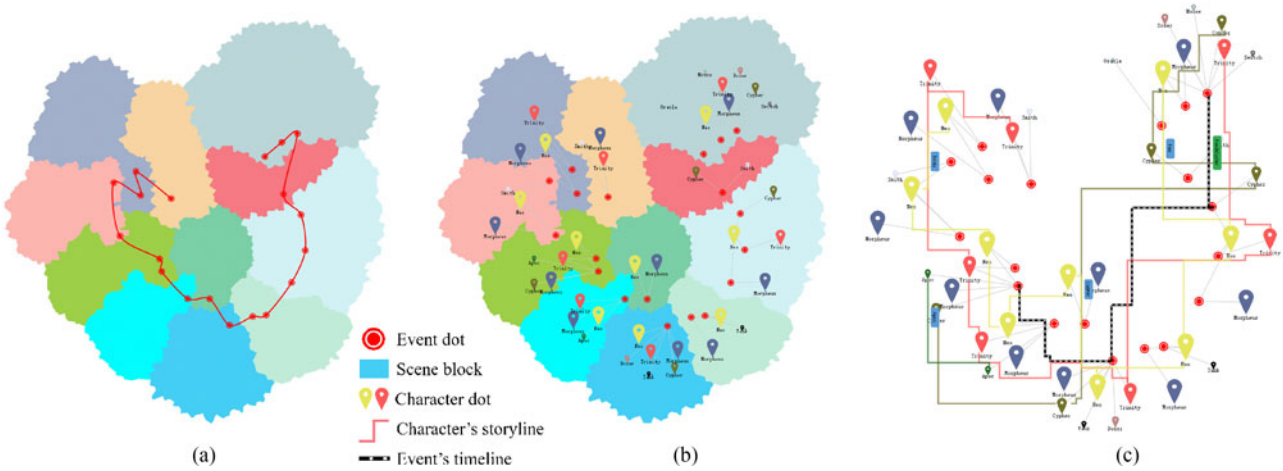

Fig. 1. Geographic map metaphor can help visualize video content with multiscale structure and underlying relationships among them. Characters, events, scenes, and various relationships are mapped into map elements including map locations and roads. The movie *The Matrix* is visualized in this example. (a) Events in the video are represented by red dots and laid out according to event similarity and temporal relations by multidimensional scaling. Different scene blocks are represented by different colors and block sizes reflect time duration of events. The red curve connects all the event dots in chronological order. See Section III for full details. (b) When users zoom in VideoMap, main characters of events and their labels would appear on map. The grey thin lines directly connecting character and event represent that the character is included in the event. (c) Different characters' storylines are in different colors to depict a timeline which connects the corresponding character dots in chronological order (each character's label is shown on the storyline). Events sharing the same characters are connected by event timelines visualized as railways. In this example, events which include "Neo" and "Cypher" are connected by the railway line in chronological order. With convenient and effective interactions including panning, zooming, and sketch gestures between users and the map, various relationships are uncovered through visual analysis and personalized contents are displayed according to users' interests. Images are rendering in high-resolution, allowing for close-up examination.

via multidimensional scaling. Time curves can also be used in video visualization [25]. In the visual analysis of surveillance videos, users are able to find the outlying frames quickly via this type of visualization and avoid watching long, tedious videos. In the visual analysis of movies, users can detect whether similar or the same scenes appear multiple times. Time curves [24] have strong stability and reproducibility, contributing to the mining of time series data.

Interaction is important when analyzing video content. Conventional interaction with videos by dragging the timeline and pressing some functional buttons is burdensome when users are viewing long videos. Recently, many researchers have worked on new interaction techniques that help process video more quickly and efficiently. Shah and Narayanan [26] proposed an object-centric representation for easy and intuitive navigation and manipulation of videos. Object-centric representation allows a user to directly access and process objects as basic video components. Their system allows users to retime, reorder, remove or clone video objects in a 'click and drag' fashion. Nguyen *et al*. [27] proposed a 3D Direct Manipulation Video navigation system, which allows a user to navigate a video by dragging an object along its motion trajectory. The system attempts to break through the limitation of temporal ambiguities in a video with complex motion, such as recurring motion, self-intersecting motion, and pauses. Wang *et al*. [28] presented an interactive system for efficiently extracting foreground objects from a video. This system provides a novel painting-based user interface that allows users to easily indicate the foreground object across space and time. Ramos and Balakrishnan [29] presented a new variation of fish-eye views called twist-lens and incorporated this into a position control slider designed for the effective navigation and viewing of large sequences of video frames. They also explored a new style of widgets that exploits the use of the pen's pressure-sensing capability, increasing the input vocabulary available to the user. Pimentel *et al*. [30] proposed the capture of typical video watching-and-commenting tasks to automatically generate a corresponding annotated interactive video. These works provide various methods to browse videos, but the efficiency of their interaction operations can still be improved. In our work, we use sketch interaction which is more natural and user friendly so that people can use it via native skills without significant training.

In this paper, we propose to use a map metaphor for visualizing video content data with user-friendly interaction. With the map form of visualization, users can browse video as if reading a map. Through operations such as dragging, zooming and path searching, users can browse and understand complex video content more efficiently.

III. VideoMap Visualization and Analysis

A. Multiscale Representation of Video Content

In the paper, we explore a new means of video content description in the form of context-free grammar (CFG), which has been widely used to describe a certain type of the structure of sentences and words in natural language. In our study, we use CFG to summarize video content in a multi-scale representation that exhibit complex video data at different scales. A context-free grammar $G = (N, \Sigma, P, S)$, where G is a quaternion, is qualified to construct its derived structure if and only if (1) N is a set of non-terminal symbols, and Σ represents a set of terminal symbols, $N \cap \Sigma = \varnothing$, and (2) P implies a set of generation rules, e.g., variable X $\rightarrow$ variable Y ('A'-'Z' | 'a'-'z'), where

	Description	Time
S→AB	Hotel→Street	[0:00]
A→*ab*	*Smith* chases *Trinity*	[6:35]
B→*cb*	*Morpheus* helps *Trinity*	[8:55]
B→CD	*Neo*'s apartment→Party room	[9:47]
C→*d*	*Neo* in his apartment	[9:47]
D→*db*	*Neo* meets *Trinity* in the party	[11:16]
D→EF	CORTECHS office→Interrogation room	[17:00]
E→*cda*	*Morpheus* calls *Neo*, *Smith* chases *Neo*	[17:12]
F→*ad*	*Smith* interrogates *Neo*	[17:26]
E→CG	*Neo*'s apartment→Truck bed	[22:36]
C→*cd*	*Morpheus* calls *Neo*	[22:38]
G→*bd*	*Trinity* and *Neo* chat in Truck bed	[25:11]
G→H	LAFAYETTE hotel	[32:25]
H→*dbcef*	*Neo* and *Trinity* meet *Morpheus*, *Cypher* and *Apoc*	[32:25]

Symbol	Main object
A	Hotel
B	Street
C	Neo's apartment
D	Party room
E	CORTECHS office
F	Interrogation room
G	Truck bed
H	LAFAYETTE hotel
a	*Smith*
b	*Trinity*
c	*Morpheus*
d	*Neo*
e	*Cypher*
f	*Apoc*

Fig. 2. Representation of video content based on context-free grammar.

$X \in N$, $Y \in V^*, V = N \cup \Sigma$, * is the Kleene star closure, and (3) S represents a start label in the set of N. Any derivation production represents the transitions among different states. For large amount of contents in a lengthy and complex video, the representation is given in a hierarchical way that provides information with different scales.

In our study, a video is described in a hierarchal structure including events, scenes, characters and relations. In this hierarchical structure, video contents are aggregated according to their attributes at different scales, so that users can access necessary information with different scales of detail. A video dataset $G = (N, \Sigma, P, S)$ can describe its content in which N and Σ implies the finite set of scenes and characters, $N \cap \Sigma = \varnothing$. P characterizes the development of the story line in video, including events and relationships, and S depicts the initial state of the video. We represent and visualize the video by beginning at the initial state S and passing through different levels, with event, scene and other informational elements in chronological order. An instance of CFG-based description for video content is described in Fig. 2. For example, the production rule "S → AB" means that an event (A → ab) occurs in scene A causes a successive event (B → cb) that occurs in scene B, where characters "a" and "b" appear in A, characters "c" and "b" appear in B. From the meaning of the symbols, we know that it is a chasing event in which Smith is chasing Trinity from a hotel to a street at the beginning of "The Matrix". We can deduce the resultant grammar according to the corresponding rules. Using any P can access the detailed video content in different scales, such as S → AB, A → ab, B → cb. We derive S → abcb, which helps to learn that "Smith chases Trinity from a hotel to the street, and Morpheus helps Trinity in the street". In the same way, we can obtain more information from this description and the derivation relation following the hierarchal structure.

Note that in the hierarchical structure of a video, the complex content-based development shows the derivation relation based on time (e.g., scene/event A contributes to the next one that has a relevant relationship with A), which can be intuitively expressed with the aid of the description of the production. The special structural expression of video content represents the state transitions and derivation tree for the CFG, and the process of sentence deduction can depict a hierarchy of information (events, scenes, characters and various relations) in VideoMap. Therefore, CFG provides us with an intuitive and visual description for exploring hierarchical details of video content.

B. Video Data Processing

1) Definition of Video Data: Given a set of entities and their relationships in chronological order, VideoMap aims to generate a legible narrative visualization and support real-time exploration and analysis. The input video data applied in VideoMap is a set of entities including scenes, events and characters organized in a multi-scale representation using CFG as presented in Section III-A. Each event, which indicates a time span of a set of interacting characters in a video clip, belongs to a certain scene in which the event takes place. Throughout this paper, we use the movie "The Matrix" as an example to illustrate the basic idea of our method. These data sets are extracted from the movie and other publicly available information manually. In addition, our method can also be easily applied to other datasets with hierarchical structures.

More precisely, the basic form of the data is a set of scenes denoted as $S = \{s_1, s_2, \ldots, s_n\}$, where s_k $(1 \leq k \leq n)$ consists of a corresponding set of temporally successive events. Each event holds four attributes: initial time t_i, duration t_d, involved characters $C = \{c_1, c_2, \ldots, c_m\}$ represented by a chronological list of characters according to their order of occurrence.

After video data pre-processing, different types of video data are mapped into map elements in a graphics library, such as locations and different roads. Events' locations are allocated according to their similarities through multidimensional scaling as explained in Section III-B-4. Afterwards, VideoMap places events on the map and generates blocks surrounding them to represent scenes as explained in Section III-B-5. Event dots are surrounded by the accompanied characters, meaning that those characters are involved in the event. The grey thin roads connect characters with their corresponding events, and the lengths of roads represent their order of occurrence in events. Additionally, some characters may participate in successive events that occur in the same scene, so they are placed in the middle area among these events on the map.

Different types of roads encode different relationships of these datasets. In Fig. 1(a), the red line connects all events in chronological order, and other colourful lines in Fig. 1(c) show the order of appearance of the same character distributed in different events. The events in which two or more characters participate are connected by the railway line according to the order of occurrence.

2) Importance of Characters: Characters are the most important elements in video data, and users pay more attention to them than events and scenes. Consequently, it is necessary to study their relative status or influence and render this information in the map. Inspired by social network analysis, we use

TABLE I
KATZ CENTRALITY VALUES OF MAIN CHARACTERS

Characters	value
Neo	0.55
Morpheus	0.50
Trinity	0.43
Cypher	0.31
Dozer	0.2
Apoc	0.17
Smith	0.15
Tank	0.14
Switch	0.12
Mouse	0.12
Oracle	0.03

the Katz centrality [32] to determine the relative importance of main characters. Characters are regarded as isolated nodes in this network. Two characters are connected with an edge if they both engage in the same event, and the number of events in which both characters participate determines the weight of the corresponding edge. In our experiment, we set the attenuation factor $\alpha = 0.5$. In terms of the convergence condition, the iteration will stop if the total delta of all vertices is below 1e–6. The Katz centrality values of main characters are represented in Table I.

3) Similarity Between Events: In this section, we define a similarity metric of events that is used in the next subsection to develop a layout algorithm for grouping events in a map. To make event grouping intuitive and reasonable, below we define a similarity metric from three different aspects of events: (1) time duration, which is the temporal dimension of an event, (2) high-level semantic feature characterized by main characters' information in the event and (3) low-level visual feature characterized by SIFT features of scenes.

Before computing the similarity, we use unity-based normalization to process the data.

$$t_i = \frac{t_i - t_{\min}}{t_{\max} - t_{\min}}$$

$$\text{score}\ (c_i) = \frac{\text{score}(c_i) - \min\{\text{score}(c)\}}{\max\{\text{score}(c)\} - \min\{\text{score}(c)\}}$$

$$\text{sift}\ (i,j) = \frac{\text{sift}(i,j) - \min\limits_{m \neq n}\{\text{sift}(m,n)\}}{\max\limits_{m \neq n}\{\text{sift}(m,n)\} - \min\limits_{m \neq n}\{\text{sift}(m,n)\}}$$

$$\begin{aligned}\text{Sim}(\text{Event}_i, \text{Event}_j) &= (w_1|t_j - t_i|^2 + w_2|\text{score}(c_j) - \text{score}(c_i)|^2 \\ &\quad + w_3|1 - \text{sift}(i,j)|^2)^{-1/2}.\end{aligned}$$

Note that w_1, w_2 and w_3 are the weights; we set them to 0.55, 0.25 and 0.2, respectively. t_* stands for the duration time of an event, c_* represents the set of main characters involved in the event, and $score(c_*)$ is the sum of characters' Katz centrality value explained in Section III-B.2, $sift(i,j)$ denotes the number of matched feature points between images representing the scene information extracted from $Event_i$ and $Event_j$.

4) Layout Algorithm: After defining the similarity metric of events, we use it to develop an event layout algorithm. In order to preserve events' essential features in our 2D canvas, we take advantage of multidimensional scaling aiming to find hidden patterns in data by rescaling a set of similarity measurements into distances assigned to specific locations in a spatial configuration. We use Sammon's method [34] to solve this problem. The algorithm is summarized as follows. It begins with a pre-configured set of two-dimensional points and then iterates over all event points. With each iteration, it aims to reduce the stress between the actual distance in the high-dimensional space and the distance in the two-dimensional space. The algorithm stops if stress conditions are satisfied. The specific procedures in VideoMap are described as follows:

1) Let $N_{\varepsilon t}$ be the number of events. For each event, we define a vector in the three-dimensional space as $X_{i|i=1,2.....,N_{\varepsilon t}}$, and we define a corresponding vector in the two-dimensional space as $Y_{i|i\,=\,1,2.....,N_{\varepsilon t}}$. For each pair of events, we define the distance in three-dimensional space between them as $D^*_{ij} = dis\,[X_i, X_j] = \text{Sim}(\text{Event}_i, \text{Event}_j)^{-1}$ and the corresponding distance in the 2D plane is $D_{ij} = \text{dis}[Y_i, Y_j]$.
2) We initialize the positions of event points with preconfigured coordinates generated by projecting the 3D data orthogonally on the 2D plane spanned by the original coordinates with the largest variances.
3) We define the energy function E, which reflects the stress between the actual distance in the three-dimensional space and the distance in the 2D plane as

$$E = \frac{1}{\sum_{i<j}\left[D^*_{ij}\right]} \sum_{i<j}^{N_{\varepsilon t}} \frac{\left[D^*_{ij} - D_{ij}\right]^2}{D^*_{ij}}.$$

We use the nonlinear mapping algorithm to search for the minimum of the energy function and achieve the final layout of event points. The algorithm is relatively simple to implement, and its visual output is intuitive to interpret.

5) Generation of Scene Blocks: To emphasize locations where events take place, VideoMap adopts a simple method based on [35] to generate closed contours that surround corresponding events and characters. In VideoMap, scene blocks are presented by subsets of Voronoi cells constructed in the following way.

First, a large number of random points with uniform distribution are generated on the canvas. We take these points as generators to build a Voronoi diagram as shown in Fig. 3(a). Next, for each event and its involved characters, we select a subset of Voronoi cells whose union forms the scene block of this event. A Voronoi cell is selected if the shortest distance between its generator and the event or involved characters is within a pre-set threshold d. In order to visualize the duration of events in an intuitive way and make the area of the scene block encode the total duration of its corresponding event, each event and its involved characters are assigned with a threshold d proportional to its duration t_d. In VideoMap, we set $d = \alpha t_{\rm d}$, $\alpha = 2.2$. Consequently, the larger the threshold d (i.e., the larger $t_{\rm d}$), the larger the area of the scene block where the event is located.

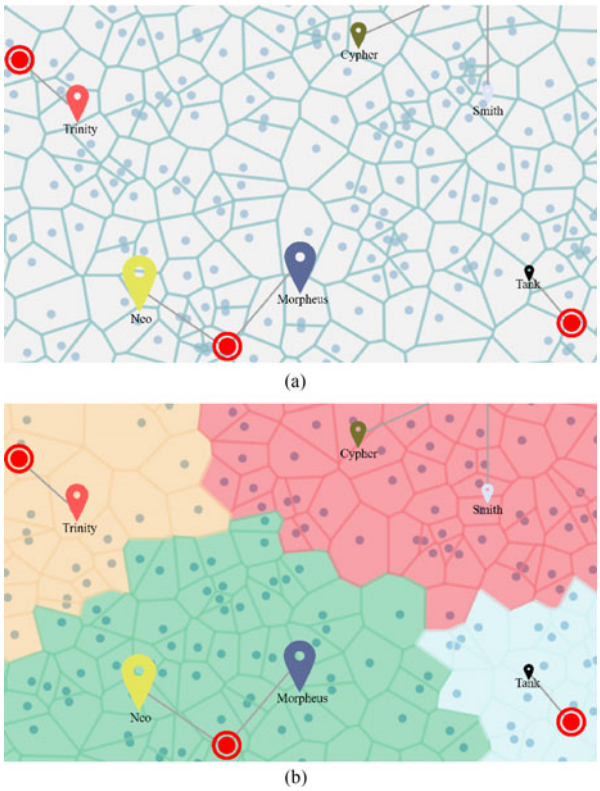

Fig. 3. Generation of scene blocks. (a) The canvas is tessellated by a Voronoi diagram of a large number of random points. (b). Voronoi cells belonging to the same scene are painted in the same color.

According to the scene information defined in Section III-B-1, each Voronoi cell is allocated to a scene block and all Voronoi cells belonging to the same scene block are painted in the same colour as shown in Fig. 3(b). This method can produce the boundaries of scene blocks that roughly follow the shape of the point set. In addition, the randomness of the points also contributes to some randomness of the outer boundaries, thus making them more map-like and natural [35].

6) A Short Summary of Map Layout: In Section III-B.1 to III-B.5, we present the algorithmic details to visualize video content (scenes, events, characters and relationships) in a map, leading to a map layout with the following characteristics: 1) nearby events in the map correspond to more similar events, 2) nearby scene blocks in the map correspond to temporally nearby scenes sharing similar characters, 3) the size of a scene block is related to the duration of the event and the number of characters in the scene, 4) the importance of a character is defined by the Katz centrality and larger character markers correspond to more important characters, and 5) all characters are sorted by their importance; only those can be displayed on the VideoMap (determined by the size of VideoMap on the screen) are treated as main characters; the remainder are secondary characters that are visualized as a single clustering object in VideoMap.

C. Visual Analysis

To help users better understand and analyze multiple relationships of a lengthy and complex video, VideoMap provides users

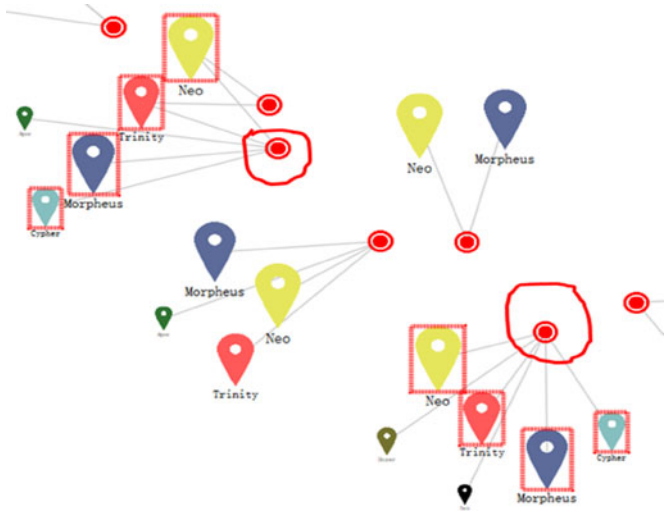

Fig. 4. Characters shared in two events are shown. Characters in red rectangular appear in both events selected by circling.

with several intuitive ways to explore video data. Relationships are encoded by different types of lines in the form of roads and railways each of which connects one or several entities in chronological order. Here, roads and railways are pictorial draws that simply illustrate two different character/event relationships. These relationship lines would not be visible at first and are served as return results from users' query requests via clicking or drawing some specific sketch gestures on the map.

On the one hand, this visual analysis process enables users to satisfy their diverse personal requirements. On the other hand, there will not be excessive redundant information that may result in visual clutter on the map. The user experience is identical to searching paths in real digital maps.

Definition of relationships: Here we define two types of relations:

1) Relationships between different characters. For example, we choose character A in event M and character B in event N (assuming that event M occurs before event N). We iterate the events happened between events M and N and call these events "middle events". If both characters A and B show up in one or more events belonging to "middle events", we recognize that they have some type of relationship and display them in the form of paths which will be explained in the following part.
2) Relationships between different events. If we select two events M and N using a circle selection gesture, the characters shared by them will be marked (Fig. 4).

Path finding: Relationship analysis serves as a powerful supplementary for VideoMap that uncovers hidden relationships among the original video data. Multiple relationships are mainly visualized as different types of paths each of which illustrates a possible interaction between two entities. More precisely, if users intend to explore the relationship between two characters distributed in different events, we iterate the "middle events" taking place between the two events to find events in which both characters are involved. These events provide important spatiotemporal information on when and where the two

characters meet each other; thus, we find bridges connecting them as in the analysis of people's trajectories in surveillance videos. The process of finding paths on VideoMap is outlined below:

1) Select two character dots c_m and c_n in chronological sequence on VideoMap, and their corresponding events are denoted as e_i and e_j.
2) Define the adjacent matrix E with the size of $(j-i+1)\times(j-i+1)$. The rows and columns of E are denoting the events $\{e_{k|i\le k\le j}\}$. The values of elements in E are initialized to 0 and are updated as follows:
   Traversing these events $\{e_{k|i\le k\le j}\}$ in order,
   if c_m is involved in both e_a and e_b $(a<b)$ and doesn't show up in e_c $(a<c<b)$,
   then $E(a,b) = 1$.
   If c_n is involved in both e_a and e_b $(a<b)$ and doesn't show up in e_c $(a<c<b)$,
   then $E(a,b) = 2$.
   Consequently, the matrix E is an upper triangular matrix. In matrix E, if the sum of the elements of a row (or a column) is equal to 3, that means the two characters meet each other in the event corresponding to this row (or column)
3) Given the graph *G* with vertices $\{e_{k|i\le k\le j}\}$, and edges as adjacent matrix E. There is an edge between e_a and e_b if $E(a,b)\neq 0$. From source e_i to destination e_j, we use the depth first search to find all paths.

Note that a character may appear multiple times in a video. If a character appears in many scenes, the tag of this character would show up in all such scenes. In VideoMap, the tags of the same character in different scenes contain different semantic information, because this character would interact with other different ones in different scenes. For example, if the tag of Neo shows up in both scenes A and B, assuming that Neo and Morpheus participate in the same event in scene A but not in scene B, paths between Neo and Morpheus will be different according to different selected Neo tags.

Path drawing: After finding all eligible paths, we use fold lines to visualize these paths. First, we use square grids with equal size to divide the VideoMap. We then use the Simplified Memory Bounded A$*$ algorithm [33] to minimize the crossover points of paths (Fig. 5). During this process, grids are occupied by character dots and event dots, and the outermost boundaries constructed by scene blocks are marked as unreachable; these grids are placed in a set and our algorithm will not iterate them. Thus, the paths will not cover the important dots and will not appear outside the VideoMap. In our implementation, each grid *n* has at most four directions ('up', 'down', 'left', 'right') to propagate the path. At each grid, we store a value $f(n)$ that denotes the cost of reaching the objective grid by taking a path through this grid

$$f(n) = g(n) + h(n)$$

where $g(n)$ denotes the number of steps from the start grid to grid *n*, and $h(n)$ is a heuristic function that stands for the Manhattan distance from grid *n* to the objective grid.

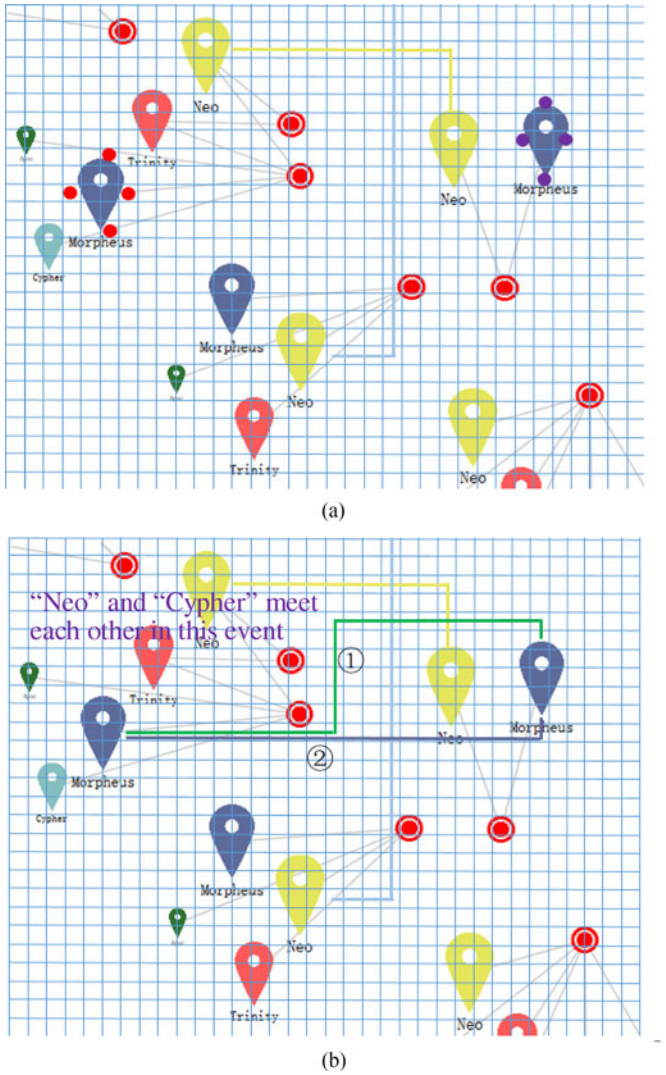

Fig. 5. Find paths with minimal crossover points of paths. (a). Map is divided by small square grids. The four red dots and four purple dots represents start points and end points, respectively. (b). In this case, line ① has two crossovers with existing lines, whereas line ② has one crossover with others. Thus line ① would be discarded in the algorithm.

If users have several simultaneous queries, increasing number of lines will increase the likelihood of more crossovers. In our implementation, we use the following method to reduce crossover between lines. Each character has four grids that represent start grids or end grids. Under these conditions, there will be 16 (4×4) paths between them. We iterate each path and record its crossovers with other existing lines and choose the one with the fewest crossover interactions. More examples of pathfinding supported on VideoMap are shown in Figs. 6 and 7. When a user picks up two character dots by a circle selection gesture (Fig. 6), the task is identical to choosing start and end points on a real map. VideoMap returns several accessible paths to show the different possible relationships between the selected characters (Fig. 7).

There are some other functions that can be used to support the exploration of video content, such as by choosing some specific event or character dots; VideoMap will display only

Fig. 6. Characters selected by drawing a line between them using sketches.

related elements corresponding to what a user chose. See the accompanying demo video for details. A user can play the specific content of a video by clicking the corresponding event point. Meanwhile, VideoMap can also give a brief summary of this event to help the user understand the details. When a user selects a character point, the relationship with other characters will be displayed. All these functions in relationship analysis will benefit users by helping them understand better video content. Sketch-based annotation is also supported in VideoMap, which assists users to sketch down their idea conveniently and facilitate later operations.

D. Interaction

VideoMap enables a rich set of user interactions for real-time data exploration and analysis in the multi-scale representation formulated by CFG with a sketch interface. The sketch-based interface explores a point in the trade-off between expressiveness and naturalness during the interaction with a map. The multi-scale representation allows users to view multiple directories of video content with different forms in a zoomable environment. Users can select, zoom in/out or drag different structures in VideoMap at any time from coarse to fine scales. For example, the first coarse scale shows scenes in different colors and the events located in certain scenes. Using the zoom-in operation, characters in events will be shown in a finer level. Furthermore, different relationships can be visualized by following the interaction operations that reflect user's requirements.

VideoMap allows users to draw editable sketches freely on the map to facilitate the exploration and visual analysis of video content. The sketching interaction for operating VideoMap consists of sketch gestures and annotations. As shown in Fig. 8(a), viewers can customize the display of VideoMap by tapping on a specific event, and a railway that connects the events sharing the same characters will appear on the map. The same method can also be applied to characters: in this case, the selected

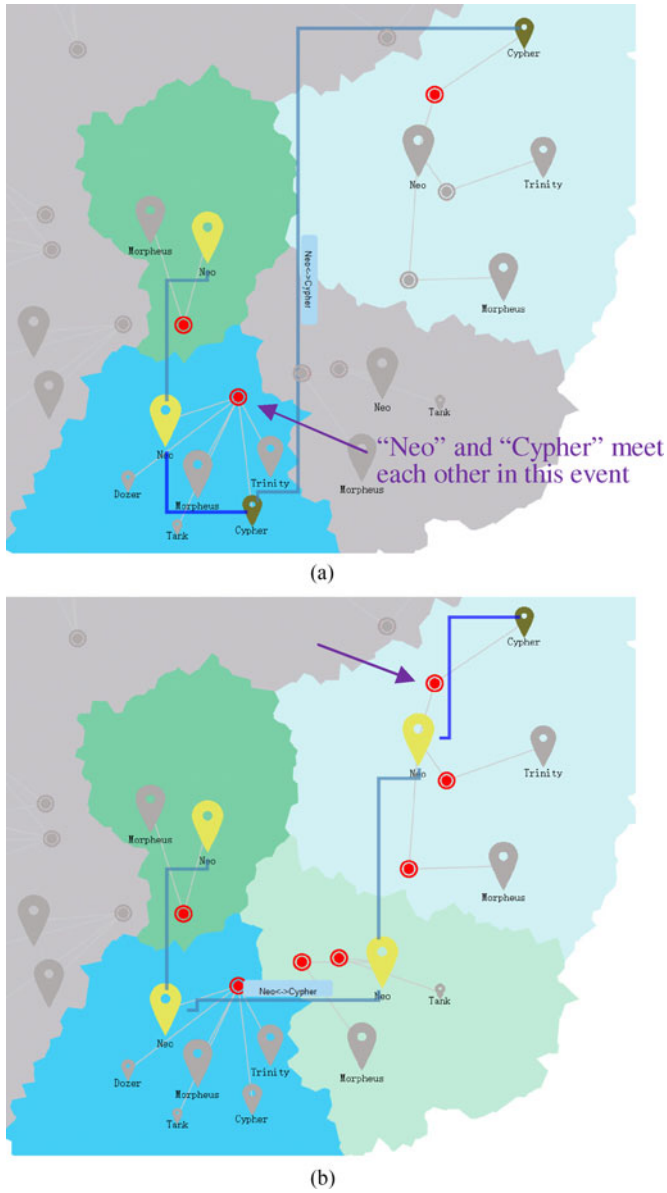

Fig. 7. Two accessible paths of different possible relationships. VideoMap uses DFS to find events in which the two specified characters meet each other. Each path represents a type of relationship between them and emphasizes the location where they meet each other. (a) and (b) show two different paths between "Neo" and "Cypher" within the specified time period. (a) An event in which "Neo" and "Cypher" meet is indicated in the figure and accordingly a path is generated. (b) Another event in which "Neo" and "Cypher" meet is indicated in the figure and accordingly a path is generated.

character's storyline shows up in response, as illustrated in Fig. 8(b). Viewers can zoom in and out on the VideoMap to show more details by finger sliding. VideoMap recognizes the sketch gesture and automatically completes different operations on the map, such as zooming, panning, or other operations for relationship analysis.

IV. Implementation

VideoMap implemented in d3.js consists of four modules: data pre-process, layout generation, projection from video to map and sketch interaction. The data pre-process module

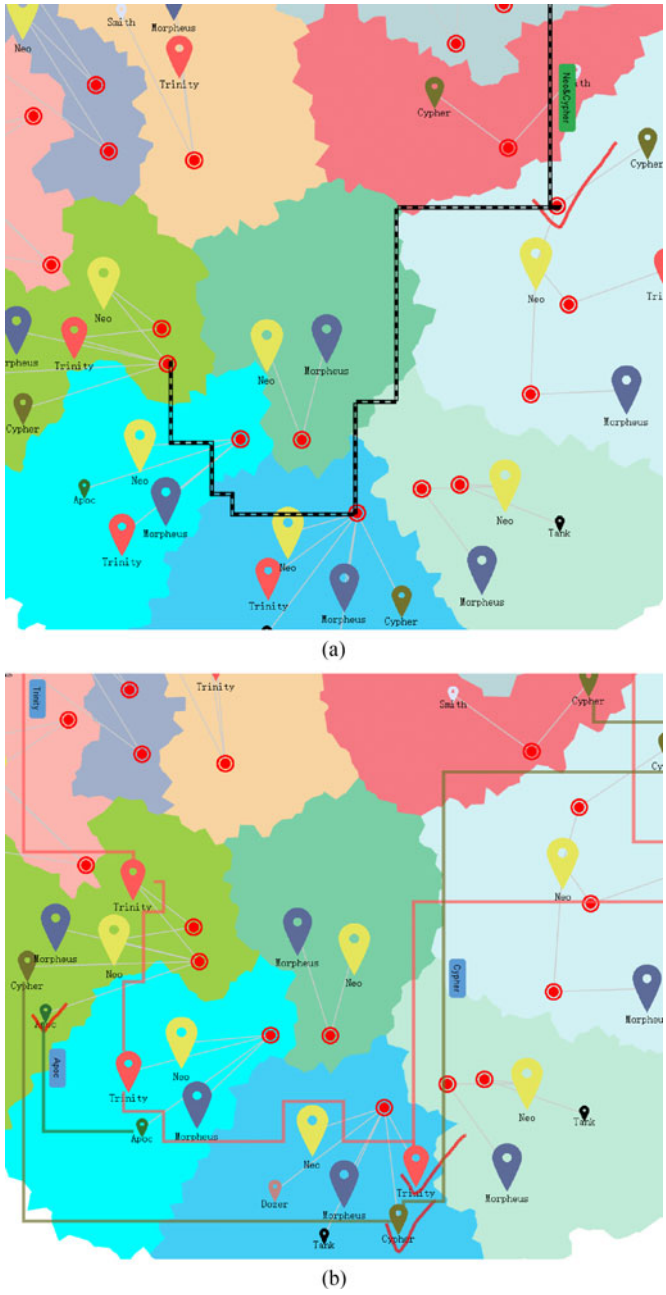

Fig. 8. VideoMap provides flexible user interactions for exploring specific events or characters. (a) Specific event tracking. Events that include the same characters with specific events are connected by the railway line in chronological order. (b) Three characters' tracking instances. Users can analyze the three characters' relationship, such as whether they ever met or whether they appeared in the same event or scene.

begins with a pre-configured hierarchical structure of video data. Users can apply different video processing systems to construct a conforming dataset. After being input with the expected form of dataset, VideoMap analyzes the importance of different characters and generates the layout of events with several attributes through multidimensional scaling. Afterwards, the corresponding characters and surrounding scene blocks are generated.

The interaction module offers several interactive functions, such as circling two or more character dots to find their connection and customized display of specific information. During the sketch interaction between viewers and maps, additional underlying relationships show up in the form of various roads, which avoid covering key elements by producing crossovers between roads and maintain the map's clarity. In addition, we provide interaction operations such that viewers can input their visual feedback such as correcting the definitions of events or changing the relationships among characters. In this way, VideoMap can exhibit personalized content based on viewer's interests.

V. Evaluation

VideoMap aims to serve as an efficient and intuitive tool for exploring video content. We conducted the study to evaluate the VideoMap and demonstrated how the system can facilitate exploring video content in an intuitive way and help users to quickly understand and find events of interest. First, we organized a symposium for the purpose of obtaining suggestions and comments on VideoMap from participants who are engaged in video related industry. We then evaluated the time that participants needed to become familiar with VideoMap and evaluated the multi-scale functions in VideoMap with complex video data. Afterwards, we presented the user study to compare the proposed VideoMap with Storyline [31], a state-of-the-art video visualization and interaction method. For a consistent evaluation, participants were required to use the specified methods to browse videos, and all experiments were performed on a Fujitsu Limited LIFEBOOK T Series (Intel Core i3 U 380 1.33 GHz) running Window 7.

A. Symposium Evaluation

In the symposium, five participants who are engaged in video-related industry were invited to become familiar with VideoMap and share their opinions on how well our design meets their standards on exploring the content of video. Two of them are TV directors who teach at the university level, and three of them are graphics designers for advertising production who majored in digital media. All of them had viewed the movie previously, and three among them had viewed it more than once. A TV director expressed that the proposed method is novel, useful and helpful from the standpoint of a TV professional. We summarize below a few of their representative viewpoints:

1) Acquiring the video content of interest with VideoMap is efficient and intuitive. The process of understanding and exploring video content is interesting and similar to a treasure hunt.
2) The manifestation of a character's relationship between different events and scenes in videos should be improved because the existing form of visual information is seemingly somewhat complicated.
3) The definition of events performed in video could be optimized to a better extent when considering that different standpoints of video information classification might make a different contribution to efficient access of a video.

After the symposium, some valuable suggestions about the design representation from participants were integrated into VideoMap. In particular, the color matching of scene blocks (for further optimizing event visualization), icons representing characters and events (for better classification), and lines representing relationships (for better visualization of relationships) were redesigned and improved.

It is interesting to note that a participant found some new information on a relationship between Cypher and Tank by using VideoMap. Cypher and Tank are not main characters in the movie. Therefore, most fans would not pay much attention to them. With VideoMap, when this participant drew a line between them, their relationship appeared on the map in an intuitive way.

B. Time Evaluation on Familiarization With VideoMap

Considering that VideoMap shows a novel multi-scale representation of complex video content, we provided an experiment to measure the time spent on familiarization with the map. We invited 8 participants who had not previously viewed the movie. Their ages ranged from 21 to 25, and all of them were at the postgraduate level. There was no significant difference in the skills of computer operation and video browsing among participants according to their self-reports and our observations. They were divided into four groups (G1, G2, G3 and G4), each of which contained two participants.

Experimental tasks. We designed four levels of training schedules. For G1, there was no training time before starting the evaluation session; For G2, G3 and G4, each group received one of the training schedules (15-min, 30-min and 45-min of training time). The participants practiced using VideoMap based on the developer's explanation and viewing of the training demo. The movie "Star Wars: Revenge of the Sith" was used in the training.

After the training, in the experiment, the movie "The Matrix" was used. Each group was asked to select one correct answer from three candidates for two questions: "Who helped Neo come to the true world from the Matrix?" and "Who was killed by the antivirus of the Matrix?" We recorded the time they took to complete the task.

Experiment results: The average time to complete the above questions for each group was as follows: no training, M = 67.5 min; 15 min of training, M = 36.5 min; 30 min of training, M = 27 min; 45 min of training, M = 24.5 min. The completion time decreased with increasing training time. The differences between two successive training schedules decreased, and the learning curves flattened when participants received more training.

C. Evaluation of Multiscale Operations for Scaling With Complex Data in VideoMap

When the video data become increasingly complex, the map will expand in size and cannot fit in a single screen. This would require more zooming and panning operations to search, which would increase the browsing time. The multi-scale operations in VideoMap can help improve the browsing efficiency with complex data, as in the multi-scale Google Maps that can navigate from an overview of the earth to a single street in Beijing easily and quickly.

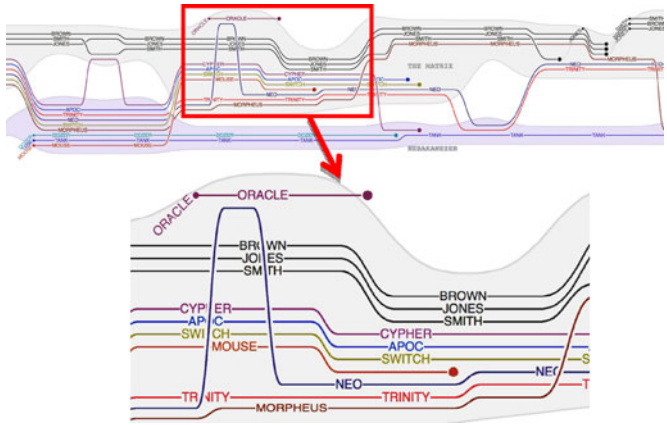

Fig. 9. Storyline [31] used in the experiment.

To evaluate the performance of multi-scale operations in VideoMap, two versions of VideoMap were constructed: 1) VideoMap-1: all information is represented simultaneously in a VideoMap without semantic zooming in/out operations, and 2) VideoMap-2: information is represented in a multi-scale structure supported by zooming in/out operations. Other interaction operations are the same in both versions of VideoMap.

Experimental procedure. We invited 20 participants (12 females and 8 males, all at the postgraduate level) who had not previously viewed the movie. Participants were divided into two groups of equal size (G1, G2). G1 was trained to use VideoMap-1 for approximately 15 min, whereas G2 received training instructions on how to use VideoMap-2 for approximately 15 min too. G1 and G2 were required to complete the two questions by providing the correct answer from three candidates: "Who took Neo to meet Morpheus?" When does Neo realize his potential of infinite power?" The time to complete was recorded.

Experimental results: We compared the two groups using nonparametric test for independent samples. The VideoMap-1 group (M = 47.90, SD = 22.11) has statistically larger mean than that of the VideoMap-2 group (M = 29.70, SD = 6.45) at significance level 0.05 (Mann-Whitney U = 73, p = 0.04425). Results show that when the map expands in size and does not fit in the screen with complex video data, the multi-scale operations can improve the browsing efficiency.

D. Comparison Between VideoMap and Storyline [31]

We further evaluated VideoMap by comparing it with the state-of-the-art visualization method *Storyline* [31] (Fig. 9). Sixteen participants took part in our study. In order to reduce the effects of performance caused by individual capabilities or backgrounds, we recruited university students with similar educational backgrounds and computer skills whose ages range from 20 to 35, including 8 females and 8 males. They were divided randomly into two groups of equal size to use VideoMap (G1) and Storyline (G2) (Fig. 9). All participants had not previously viewed the movie.

TABLE II
QUESTIONNAIRE 1: PLEASE SELECT ANSWERS REPRESENTING YOUR OPINION ABOUT THE SYSTEM YOU HAVE USED

(1) It was an efficient and intuitive system for exploring video content.
(2) It was easy to have an overall structure of the hierarchical contents in the video.
(3) I thought this visualization method is novel and easy to master.
(4) I thought the interactive operation is convenient and useful.
(5) I was satisfied with the process.

TABLE III
QUESTIONNAIRE 2: PLEASE SELECT THE ANSWER THAT REPRESENTS YOUR OPINION ABOUT VIDEOMAP

(1) VideoMap provides the pathfinding to facilitate analysis and understanding of video content.
(2) The sketch-based interaction on VideoMap is efficient and convenient.
(3) Zooming in/out of VideoMap was a useful and convenient for exploring video content and finding goals of interest.

In the training process using the movie "Star Wars: Revenge of the Sith", the participants were trained to become familiar with VideoMap or Storyline based on the explanation and watching a training demo for approximately 30 min. In more details, for G1, they were instructed on the meaning of map elements in VideoMap and how to use interaction operations, such as multi-scale zooming in/out, sketch gestures, path finding and so on. A demo video was then provided to participants for a comprehensive understanding. For G2, they were instructed on the meaning of elements in Storyline and how those are organized to represent the video content.

After the training, an unstructured interview was conducted with participants regarding how they felt about the flexibility and usability of VideoMap and Storyline from some aspects, including ① how it works for exploring video content; ② the method of representing the different kinds of video data; ③ the visual feedback of its layout and interface; ④ different interaction operations; ⑤ the experience on the operation process. Participants exchanged their opinions about these aspects during the communication. We summarized some viewpoints:

1) The participants thought that VideoMap is novel, fresh, and interesting. For example, from the view point of interface, they could interact with video content as if reading a map by panning and zooming in/out. From the viewpoint of layout, the different roads representing different relationships are new, intuitive and interesting.
2) Multi-scale representation can provide video content from the overview to the part of focus. We can reach an overall perception following the scene blocks and further access the details about relationships among specific characters.

After the training and interview, the participants were required to complete the questionnaire in Table II based on the communication in the interview. The participants in G2 were then introduced to VideoMap for approximately 30 min. Finally, the sixteen participants were required to complete the questionnaire in Table III.

We applied the Likert 7-level scale to analyze and summarize their results. We recorded every answer of each participant using scores of 1 to 7 (strongly disagree to strongly agree). For the results of questionnaire 1 (Table II), the average answer of each group for different methods is summarized in Fig. 10. Furthermore, nonparametric (two independent samples) tests were performed to compare two groups of cases on each question at an alpha level of 0.05. Participants agreed that VideoMap ($M = 6.38$, $SD = 0.52$) was easier to achieve an overall structure of the hierarchical contents in video than Storyline ($M = 4.63$, $SD = 0.74$) (Mann-Whitney $U = 2.5$, $Z = -3.23$, $p = 0.001$). They thought that the interactive operation of VideoMap ($M = 6$, $SD = 0.93$) was more convenient and useful than Storyline ($M = 4.38$, $SD = 0.52$) (Mann-Whitney $U = 4.5$, $Z = -3.03$, $p < 0.01$), and the process of VideoMap ($M = 5.75$, $SD = 0.71$) was more satisfactory than Storyline ($M = 4.88$, $SD = 0.64$) (Mann-Whitney $U = 12.5$, $Z = -2.23$, $p < 0.05$).

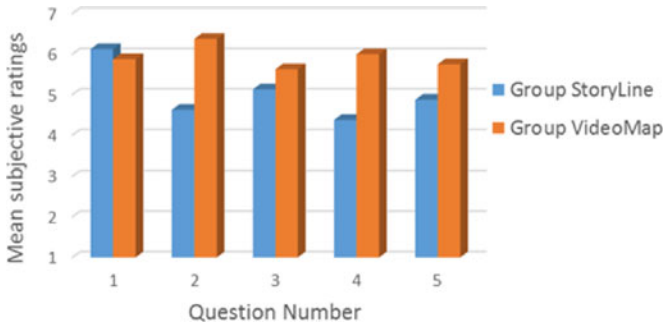

Fig. 10. Comparison of the subjective ratings on questions in Table II between VideoMap and Storyline.

From the results of questionnaire 2 (Table III), we conclude the following:

1) 94% of participants (15 of 16) gave positive feedback about VideoMap.
2) 88% of participants (14 of 16) thought that the pathfinding of VideoMap was interesting and useful for facilitating analysis and understanding of video content.
3) 81% of participants (13 of 16) gave positive feedback about the sketch interaction in VideoMap.
4) 81% of participants (13 of 16) thought that the multi-scale views using VideoMap zooming in/out was a useful and convenient method for exploring video content and finding interesting goals.

The above results show that VideoMap is an intuitive, efficient and friendly approach to help users explore the content of video via an interesting experience.

VI. CONCLUSION

In this paper, we present VideoMap, a novel and narrative technique based on the map metaphor for visualizing video data with hierarchical structures. VideoMap exhibits both the overview and level-of-detail features from a very large scale to a small scale of finely detailed representation. It can not only help users trace the evolution of story but also serve as a tool for revealing hidden associations and patterns behind the original video data. Moreover, VideoMap incorporates free interactions widely used in digital maps (e.g., Google Maps) such

as multi-scale zooming and panning to assist users in reasoning while providing visual data and reducing the burden of viewing the entire content of any given video. A sketch interface for VideoMap is provided to facilitate the visual analysis. User studies were performed and the results showed that VideoMap offers a promising tool for facilitating users in efficiently exploring video content with an intuitive and natural interaction.

A limitation of VideoMap is that it does not work well on non-chronological storylines because the relationships between different characters in VideoMap rely on this temporal information. Future work includes optimizing the layout algorithm with a fast Euclidean distance transformation [37] and developing alternative methods to generate VideoMap for non-chronological storylines and with less computational cost. Furthermore, more advanced analysis methods of exploring video content are possible in VideoMap through data description and sketch interactions.

ACKNOWLEDGMENT

The authors would like to thank the editor and reviewers for their valuable comments, which helped improve this paper.

REFERENCES

[1] G. N. Ye, Y. T. Li, H. L. Xu, D. Liu, and S. F. Chang, "EventNet: A large scale structured concept library for complex event detection in video," in *Proc. 23rd ACM Int. Conf. Multimedia*, Oct. 26–30, 2015, pp. 471–480.

[2] J. Yang, B. Price, X. Shen, Z. Lin, and J. Yuan, "Fast appearance modeling for automatic primary video object segmentation," *IEEE Trans. Image Process.*, vol. 25, no. 2, pp. 503–515, Feb. 2016.

[3] D. Mashima, S. G. Kobourov, and Y. Hu, "Visualizing dynamic data with maps," *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 9, pp. 1424–1437, Sep. 2012.

[4] E. R. Gansner, Y. Hu, and S. North, "Visualizing streaming text data with dynamic graphs and maps," in *Graph Drawing*. Berlin, Germany: Springer, 2013, pp. 439–450.

[5] C. Gray, J. Kim, P. Asente, and J. Collomosse, "Comprehensible video thumbnails," *Comput. Graph. Forum*, vol. 34, no. 2, pp. 167–177, 2015.

[6] Y. Taniguchi, A. Akutsu, and Y. Tonomura, "Panoramaexcerpts: Extracting and packing panoramas for video browsing," in *Proc. 5th ACM Int. Conf. Multimedia*, 1997, pp. 427–436.

[7] D. B. Goldman, B. Curless, D. Salesin, and S. M. Seitz, "Schematic storyboarding for video visualization and editing," *ACM Trans. Graph.*, vol. 25, no. 3, pp. 862–871, 2006.

[8] X. S. Hua, S. Li, and C. Z. Zhu, "Video booklet," *IEEE Comput. Soc.*, 2010.

[9] C. Nguyen, Y. Niu, and F. Liu, "Video summagator: An interface for video summarization and navigation," in *Proc. SIGCHI Conf. Human Factors Comput. Syst.*, 2012, pp. 647–650.

[10] R. Shah and P. J. Narayanan, "Interactive video manipulation using object trajectories and scene backgrounds," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 9, pp. 1565–1576, Sep. 2013.

[11] C. X. Ma, Y. J. Liu, H. A. Wang, D. X. Teng, and G. Z. Dai, "Sketch-based annotation and visualization in video authoring," *IEEE Trans. Multimedia*, vol. 14, no. 4, pp. 1153–1165, Aug. 2012.

[12] J. Sang and C. Xu, "Character-based movie summarization," in *Proc. 18th ACM Int. Conf. Multimedia*, 2010, pp. 855–858.

[13] B. Benfold and I. Reid, "Stable multi-target tracking in real-time surveillance video," in *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, Jun. 2011, pp. 3457–3464.

[14] W. M. Hu, "Semantic-based surveillance video retrieval," *IEEE Trans. Image Process.* vol. 16, no. 4, pp. 1168–1181, Apr. 2007.

[15] C. Piciarelli, G. L. Foresti, and L. Snidaro, "Trajectory clustering and its applications for video surveillance," in *Proc. IEEE Conf. Adv. Video Signal Based Surveillance*, Sep. 2005, pp. 40–45.

[16] L. Klein, H. Schlunzen, and S. K. Von, "An advanced motion detection algorithm with video quality analysis for video surveillance systems," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 1, pp. 1–14, Jan. 2011.

[17] M. M. Yeung and B. L. Yeo, "Video visualization for compact presentation and fast browsing of pictorial content," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 5, pp. 771–785, Oct. 1997.

[18] S. Uchihashi, "Video manga: Generating semantically meaningful video summaries," in *Proc. 7th ACM Int. Conf. Multimedia*, Oct./Nov. 1999, pp. 383–392.

[19] G. Y. Zhu *et al.*, "Trajectory based event tactics analysis in broadcast sports video," in *Proc. 15th Int. Conf. Multimedia*, Sep. 24–29, 2007, pp. 58–67.

[20] M. L. Parry *et al.*, "Hierarchical event selection for video storyboards with a case study on snooker video visualization," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 12, pp. 1747–1756, Dec. 2011.

[21] L. X. Xie *et al.*, "Structure analysis of soccer video with hidden Markov models," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2002, vol. 4, pp. IV-4096–IV-4099.

[22] M. Tapaswi, M. Bäuml, and R. Stiefelhagen, "StoryGraphs: Visualizing character interactions as a timeline," in *Proc. IEEE Conf. Comput Vis. Pattern Recog.*, Jun. 2014, pp. 827–834.

[23] G. M. Jing, Y. T. Hu, Y. W. Guo, Y. Z. Yu, and W. P. Wang, "Content-aware video2comics with manga-style layout," *IEEE Trans. Multimedia*, vol. 17, no. 12, pp. 2122–2133, Dec. 2015.

[24] B. Bach *et al.*, "Time curves: Folding time to visualize patterns of temporal evolution in data," *IEEE Trans. Vis. Comput. Graph.*, vol. 22, no. 1, pp. 559–568, Jan. 2016.

[25] Y. J. Liu *et al.*, "An interactive spiraltape video summarization," *IEEE Trans. Multimedia*, vol. 18, no. 7, pp. 1–14, Jul. 2016.

[26] R. Shah and P. J. Narayanan, "Trajectory based video object manipulation," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2011, pp. 1–4.

[27] C. Nguyen, Y. Niu, and F. Liu, "Direct manipulation video navigation in 3D," in *Proc. SIGCHI Conf. Human Factors Comput. Syst.*, pp. 1169–1172, 2013.

[28] J. Wang *et al.*, "Interactive video cutout," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 585–594, 2005.

[29] G. Ramos and R. Balakrishnan, "Fluid interaction techniques for the control and annotation of digital video," in *Proc. 16th Annu. ACM Symp. User Interface Softw. Technol.*, 2005, pp. 105–114.

[30] M. G. Pimentel, R. Goularte, R. G. Cattelan, F. S. Santos, and C. Teixeira, "Enhancing multimodal annotations with pen-based information," in *Proc. 9th IEEE Int. Symp. Multimedia Workshops*, 2007, pp. 207–213.

[31] Y. Tanahashi and K. L. Ma, "Design considerations for optimizing storyline visualizations," *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 12, pp. 2679–2688, Dec. 2012.

[32] L. Katz, "A new status index derived from sociometric analysis," *Psychometrika*, vol. 18, no. 1, pp. 39–43, 1953.

[33] R. E. Korf, "Linear-space best-first search," *Artif. Intell.* vol. 62. no. 1, pp. 41–78, 1993.

[34] J. W. Sammon, "A nonlinear mapping for data structure analysis," *IEEE Trans. Comput.*, vol. C-18, no. 5, pp. 401–409, May 1969.

[35] E. R. Gansner, Y. Hu, and S. Kobourov , "GMap: Visualizing graphs and clusters as maps," in *Proc. IEEE Pacific Vis. Symp.*, Mar. 2010, pp. 201–208.

[36] M. Yu, Y. J. Liu, S. J. Wang, Q. Fu, and X. Fu, "A PMJ-inspired cognitive framework for natural scene categorization in line drawings," *Neurocomputing*, vol. 173, pp. 2041–2048, 2016.

[37] Y. S Leung, X. Wang, Y. He, Y. J. Liu, and C. C. L. Wang, "A unified framework for isotropic meshing based on narrow-banded Euclidean distance transformation," *Comput. Vis. Media*, vol. 1, no. 3, pp. 239–251, 2015.

Cui-Xia Ma received the B.S. and M.S. degrees from Shandong University, Jinan, China, in 1997 and 2000, respectively, and the Ph.D. degree from the Institute of Software, Chinese Academy of Sciences, Beijing, China, in 2003.

She was a Research Associate in the Department of Computer Science, Naval Postgraduate School in Monterey, Monterey, CA, USA, from 2005 to 2006. She is currently a Professor with the State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences. Her research interests include human computer interaction and multimedia computing.

Yong-Jin Liu (M'13–SM'16) received the B.Eng. degree from Tianjin University, Tianjin, China, in 1998, and the M.Phil. and Ph.D. degrees from the Hong Kong University of Science and Technology, Hong Kong, China, in 2000 and 2004, respectively.

He is currently an Associate Professor with the Tsinghua National Laboratory for Information Science and Technology, Department of Computer Science and Technology, Tsinghua University, Beijing, China. His research interests include computational geometry, multimedia, computer graphics, and computer-aided design.

Guozhen Zhao received the B.S. degree in industrial engineering from Tianjin University, Tianjin, China, in 2007, and the M.S. and Ph.D. degrees in industrial and systems engineering from the State University of New York, Buffalo, NY, USA, in 2009 and 2011, respectively.

Since 2012, he has been an Assistant Professor with the Institute of Psychology, Chinese Academy of Sciences, Beijing, China. His current research interests include mathematical modeling of human cognition and performance, transportation safety, human computer interaction, and neuroergonomics and their applications in intelligent system design.

Hong-An Wang (A'01–M'01) received the Ph.D. degree from the Institute of Software, Chinese Academy of Sciences, Beijing, China, in 1999.

He is a Professor with the Institute of Software, Chinese Academy of Sciences. He is currently the Director of Intelligence Engineering Laboratory. His research interests include human–computer interaction, real-time intelligence, and real-time active database.