

Sketch-Based Annotation and Visualization in Video Authoring

Cui-Xia Ma, Yong-Jin Liu, Hong-An Wang, Dong-Xing Teng, and Guo-Zhong Dai

Abstract—Authoring context-aware, interactive video representation is usually a complex process. A user-friendly multimedia authoring environment is thus solicited to explore and express users' design ideas efficiently and naturally. In this paper we present a sketch-based two-layer representation, called scene structure graph (SSG), to facilitate the video authoring process. One layer in SSG uses sketches as a concise form with which the visualization of scene information is easily understood and the other layer uses a graph to represent and edit the narrative structure in the authoring process. With SSG, the authoring process works in two stages. In the first stage, various sketch forms such as symbols and hand-drawing illustrations are used as basic primitives to annotate the video clips and the hyperlinks encoding spatio-temporal relations are established in SSG. In the second stage, sketches in SSGs are modified and new SSG is composed for any particular authoring purpose. Three user studies are elaborated, showing that the SSG is user-friendly and can achieve a good balance between expressiveness of users' intent and ease of use for authoring of interactive video.

Index Terms—Interaction styles, multimedia computing, sketch-based interface, video authoring.

I. INTRODUCTION

THE digital technological revolution has generated a considerable collection of video data from our daily life. Interactive video authoring now plays an important role in multimedia computing and understanding. Authoring is the collection, selection, preparation, and presentation of information to one or more readers by an author [8]. The collection of original video clips do not support abstraction and interaction other than viewing. For the downstream selection and preparation, annotations on video clips are important to fill in the semantic gap between low-level image features and high-level queries. At the step of information presentation, efficient visualization is important to reduce the authoring burden of users. In this paper, we

propose to use sketch representation for both annotation and visualization of video contents, which serves as an efficient video authoring tool. The final output of the authoring process can be in either MPEG-7 [33] or W3C's SMIL 3.0 [4], and is not the emphasis of this paper.

The purpose of multimedia authoring is that people communicate message with each other using various media forms. Previous work on video authoring (e.g., in [3] and [38]) uses design primitives including texts, captions, keyframes, and videos. Captions, as well as text annotations, can provide valuable semantic information for understanding media [6], [26]. However, different countries may use different written languages and thus using text may find obstacles in a multi-linguistic environment such as those on the internet. Keyframe is another widely used format to summarize the video content [20], [43]. Compared to texts, keyframes are effective in representing visual content of a video sequence and do not have the text recognition problem in a multi-linguistic environment. However, in most video clips, keyframes are static natural images that are well known to be statistically redundant [39]: among all the visual cues in a natural image, human subjects can only see a small fraction. Distinguished from natural images, sketches are concise forms of pictorial information which have rich semantic meanings and summarize well the visual context of videos [7]. In our study, we propose to use sketches in a video authoring environment for both video annotation and visualization.

Video authoring is a design process. It is desired by users to rapidly explore, compare, and communicate diverse design ideas with high-level semantic information in an early design process. Nowadays, common users still prefer working with pen and paper, and use freehand sketches to quickly communicate and record ideas, which help them determine what the early design looks like [12], [28], [30]. In human-computer interaction, the sketch-based interface explores a point in the tradeoff between expressiveness and naturalness [23]. In the application of video annotations, complex message can be conveyed with a single sketch, as an old saying said "A good picture is worth a thousand words". For visualization, users can also sketch the structure of visual layout indicating how to integrate video clips, by retrieving and establishing hyperlinks between video clips and sketch annotations. Eventually, video authoring can be achieved by integrating related video sources based on the visual layout structures.

In this paper, we propose to use sketches to annotate and visualize the content of video resource. First, various sketch forms such as symbols and hand-drawing illustrations are used to annotate the video clips, serving as knowledge creation and extraction in video authoring. Then these sketches are automatically

Manuscript received July 16, 2011; revised November 27, 2011; accepted February 28, 2012. Date of publication March 08, 2012; date of current version July 13, 2012. This work was supported in part by the National Basic Research Program of China (2011CB302205), the Natural Science Foundation of China (61173058, 60970099), and the 863 program of China (2012AA011801). The work of Y.-J. Liu was supported in part by the Program for NCET and TNList Cross-discipline Foundation. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Samson Cheung.

C.-X. Ma, H.-A. Wang, D.-X. Teng, and G.-Z. Dai are with the Institute of Software, Chinese Academy of Sciences, Beijing, China.

Y.-J. Liu is with TNList, Department of Computer Science and Technology, Tsinghua University, Beijing, China.

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.2012.2190389

arranged into a scene structure representation and user can further edit the representation in a sketch-based interface, serving as knowledge reuse in video authoring. The contributions of this work include using various sketch forms for video annotations and utilizing two-layer scene structure graph (SSG) that serves as a concise and easy-to-use form for video authoring. Three user studies are elaborated, showing the advantages of 1) sketch-based annotations, 2) sketch-based video visualization, and 3) sketch-based SSG representations.

The rest of this paper is organized as follows. Section II summarizes the related work. Section III presents the concept of SSG, with which the authoring process works in two stages. In Sections IV and V, the two stages, annotation and authoring stages, are presented in detail. In Section VI, user experience of various sketch representations is discussed. In Section VII, we conclude the presented work and outline directions for future work.

II. RELATED WORK

In a video authoring process, to help user easily extract and organize video content in an abstract interaction, several key techniques are involved, including video annotation, summarization/visualization, and retrieval. Below we briefly summarize some works that relate to ours.

A. Video Annotation

Semantic annotations on videos can provide valuable information for media understanding [11]. Both automatic and manual annotations are explored in video annotation research. Automatic annotation methods usually segment videos into shots and extract low-level features from shots to describe video content. Automatic annotation is done by building model based on low-level features for each keyword in a vocabulary, e.g., the multiple Bernoulli relevance model in [9]. Based on whether a training set is used or not, supervised (e.g., [46]) and unsupervised (e.g., [35]) methods have been proposed for automatic video annotation. Although significant advances have been made in recent years, state-of-the-art automatic video annotations still confront with the obstacles from the large variance and diversity of video data as well as the limited size of training data.

Manual annotations are particularly useful for allowing users to create time-based and personalized annotations of videos. A typical work was presented in [13]. By providing a predictive timing algorithm for temporal alignment of annotations with video content, in [13] several advantages of manual annotations are summarized: 1) allow personalized time-based annotations; 2) allow multiple-video aggregation; 3) allow multiple-users integration; 4) allow timed navigation by using hyperlinks with annotations. In all these previous works [13], [35], [46], captions, keywords, or keyframes are used for video annotations. In this paper, we propose that users draw sketches to annotate the videos. A user study by comparing annotations using keywords, keyframes, and sketches is performed in Section VI-A, showing the advantage of sketch-based annotations.

B. Video Summarization and Visualization

Many video summarization methods have been proposed and a good survey was presented in [43]. Most of these methods extract a small collection of salient images and display them in different ways. Ueda *et al.* [44] used a moving icon associated with each keyframe to represent a shot, whose depth of 3-D structure is reflected by its time length. Yeung *et al.* [48] arranged keyframes into a poster, using temporal order to represent its dynamic content. Taniguchi *et al.* [42] proposed a PanoramaExcerpts system to synthesize keyframes for panning or tilting a shot. Taniguchi's method was limitedly used in videos with slow-motion shots. Video snapshot [32] presented a summarization of keyframes in a pictorial form based on content analysis techniques, including three ingredients, i.e., an attention model, image quality analysis, and video structurization. Hua *et al.* [17] proposed a video booklet system by arranging a set of thumbnails on a predefined set of templates in various forms. These video summarization methods rarely use the contextual information like motion cues or relationship among these cues. However, these information is important and frequently appeared in video clips, as demonstrated by Synopsis [37].

Storyboard is another popular representation of video content, which is desired by filmmakers to communicate design ideas with others. Some recent works have been proposed to improve traditional storyboard representation for video retrieval and visualization [7], [10]. However, these works focus on the visualization of only one video clip that ignore the overall structure and relationships between similar objects in different video clips. In this paper, we enrich and extend the storyboard form into an SSG that integrates various cues and can be used to visualize the narrative structure in a video authoring process. Closely related to our work, a novel video summarization system was proposed in [6] that also uses a relational graph. However, texts and keyframes were used as primitives in [6], while in our work we consistently use sketches for video visualization as we did in video annotations. A user study comparing keyword-, keyframe-, and sketch-based visualizations is presented in Section VI-B.

C. Content-Based Video Retrieval and Recommendation

One major task of video annotation and video summarization is to provide meaningful accesses to content-based video retrievals [7], [47]. Compared to the text-based retrieval according to textual relevance, content-based retrieval relies on visual content similarity for searching conceptual relevant videos. There is a large body of research on content-based or concept-based video retrieval [24], among which only a few works used free-hand sketch queries [5], [7]. The viewpoint was supported in [7] that people recall events in video using episodic memory and sketches are particularly suitable for episode description.

Most content-based video retrieval methods assume that the user can input a precise information in keywords or pictorial forms. However, people usually start with a fuzzy and inaccurate idea in a video authoring process, and thus contextual video recommendation based on user's historical and current preferences is much desired. Most conventional recommendation systems heavily rely on a sufficient collection of user profiles [2].

By integrating multimodal relevance and user feedback, a pioneer work was presented in [34] in which the presented contextual video recommendation does not need a sufficient collection of user profiles. In our study, based on user's sketching behavior, the video authoring process is run in an interactive way such that when the user sketches the SSG to layout the narrative structure, the recommended videos are more relevant to that particular user.

D. Video Authoring

One key in video authoring is to specify the individual components and their relationships in a video document, based on a collection of video resource. It involves collecting, structuring, and presenting information in digital videos [8]. The goal of an interactive presentation of authoring video is to convey and communicate message with people. In [3], the paradigms for authoring multimedia documents were categorized into four classes: structure-based, timeline-based, graph-based, and script-based. The study in [3] showed that there is no single method better than others to an authoring task and usually a combination is appropriate. Based on CMIFed [45] and SMIL language [4], a structure-based authoring environment *GRiNS* was presented in [3]. By utilizing an SSG representation with a sketch-based interface, the authoring paradigm presented in this paper is a combination of structure-base and graph-based paradigms. In Section VI-C, a user study is presented to compare the expression power of our SSG-based authoring environment and a commercial structure-based authoring environment *Adobe Encore CS4* [1].

To represent the inter-relationship between individual components in authoring video, our SSG-based authoring environment supports using hyperlinks for navigation between concept-related video clips. This is inspired by the successful work of Hypervideo [38] in which the hyperlink structures help supporting the top-down authoring of hypervideo. For multimedia authoring, we also draw attentions from researches in hyper-media and hyper-linkage construction in website design. An authoring system *Anecdote* [14] was developed for a large-scale multimedia representation using texts and images. *Anecdote* supported various authoring styles to construct the scenario framework. Concepts of surrogate media and surrogate scene, which are similar to the SSG in our work, were developed in *Anecdote*. *DENIM* [25] was another typical authoring system of websites. The concept of site maps was developed in *DENIM*, which were high level representations of a site in which pages were grouped and depicted as labels. The functionality of site maps is also similar to the SSG in our work. *Anecdote* [14] and *DENIM* [25] emphasized the system framework of website authoring, while our work focuses on sketch-based annotation and visualization with a SSG representation for video authoring.

E. Sketch-Based Interface

Both *Anecdote* and *DENIM* systems utilize sketch-based interaction. Sketch-based interactive design can be dated back to the early 1960s when Ivan Sutherland published his seminal work on the Sketchpad [41], in which he used a light pen to make

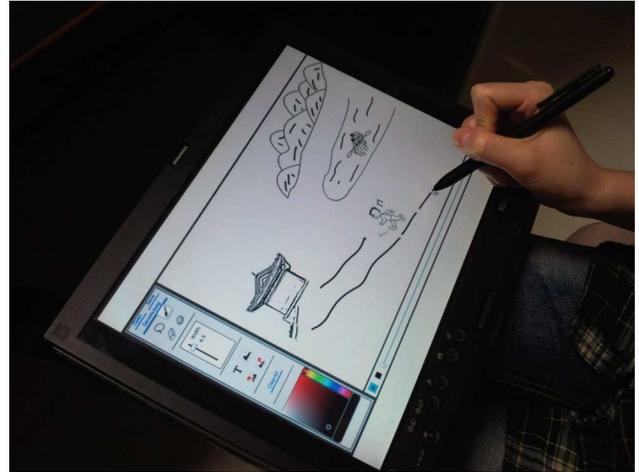


Fig. 1. Screenshot of video authoring using sketches.

drawings and create geometric primitives. Sketch-based interfaces have been successfully applied in many multimedia applications [23], [27], [30]. Closely related to our work, the segmentation, beautification, and grouping of ink through sketch-based interfaces were presented in *Pegasus* [18] and *Flatland* [36]. These systems parsed the strokes and recognize shapes, but they did not care about the collection of drawing cues and sketch contexts during the freeform writing process.

The key idea behind sketch-based interfaces is to mimic traditional paper-and-pencil-like drawing that represents a natural way of thinking and communicating ideas. In this paper, we introduce the sketching techniques into the video authoring process. A work related to ours was presented in [11] with the emphasis on object motion tracking. In this paper an interactive authoring environment is proposed to annotate and visualize video content using sketches. These sketches are then organized into SSGs to develop a narrative structure. A sketch-based interface is used in the proposed authoring process such that users can sketch out their mind like scribbling on physical paper.

III. SSG-BASED VIDEO AUTHORING

In this work, we propose to use sketch-based annotation and visualization for video authoring (refer to Fig. 1). Based on sketch representation, we develop an SSG to represent and edit the narrative structure in an authoring process. First we regard the video summarization as a model-based semantic visualization that maps the screen display to the users' perception. Here the meaning of perception follows the Gestalt law [22] that concerns about grouping elementary perceptual elements into larger structures and understanding the relation between visual stimuli and their perceptions. In a quantitative model, video summarization is a visualization model $V(d, v)$, where d is the database of video clips and v is the set of visualization primitives. For a particular authoring purpose, $V(d, v)$ should optimize the quantity of perception $P(V(d, v), p)$, where p is a set of free parameters in the model.

The proposed video annotation and visualization method uses sketches as visualization primitives v and works as follows. First various forms of sketches are annotated in video clips

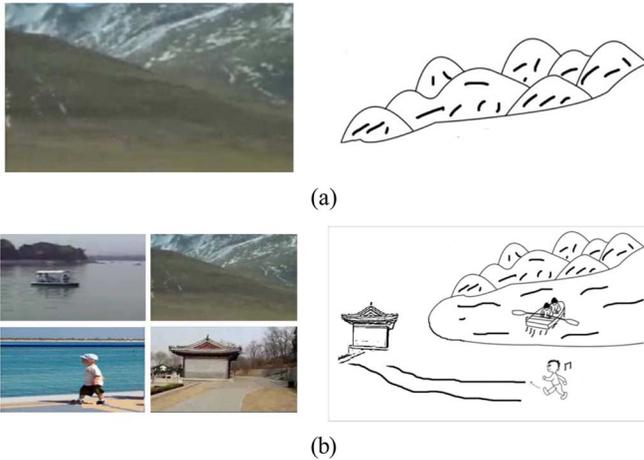


Fig. 2. Sketch-based annotation and visualization for video authoring. (a) Sketch-based annotation. Left: video shot; Right: annotation. (b) Sketch-based visualization. Left: video clips; Right: sketch visualization.

[refer to Fig. 2(a)], by either user sketching or auto-extraction from keyframes. Two forms of sketch-based annotations are presented in Section IV-A. For video visualization, $V(d, v)$ is a canvas that can be quickly perceived by the users and we define $V(d, v)$ by a scene structure graph SSG (to be defined below).

One clip may have more than one sketch annotation. Let s be the set of sketches annotated in a clip c . These sketches $s \subseteq v$ are then organized into an elementary $SSG(c, s)$ for each clip $c \subseteq d$, using the layout algorithm presented in Section IV-B. Several elementary $SSG(c, s)$ can be further edited and combined together to form a larger graph $SSG(d, v)$ [refer to Fig. 2(b)]. For efficient communication, the proposed authoring environment (refer to Fig. 1) uses a paper-and-pencil-like sketching interface, with which users can design by sketching, searching, and modifying their idea interactively with immediate and continuous visual feedback, and thus achieve optimized perception of video summarization.

Definition 1: The scene structure graph SSG is a visualization model V that is represented by two layers: a visualization layer and a graph layer.

- The visualization layer uses sketches to present a semantic summarization of the narrative structure in a video authoring process.
- In the graph layer, the nodes are sketched graphical objects. The arcs between nodes indicate the procedural information which also specify the conceptual relationship between nodes.

Fig. 3 shows two examples of SSG representations. The parameters p in $P(SSG(d, v), p)$ are the conceptual relations such as spatio-temporal relations between the sketches in v . In Section IV-B, we present a layout algorithm to optimize these spatio-temporal relations in a simple and efficient way.

The two-layer form of SSG can help users quickly overview the narrative structures and easily interact with video clips. To achieve a good quality of perception $P(SSG(d, v), p)$ in a video authoring process, the authoring environment is composed of two stages:

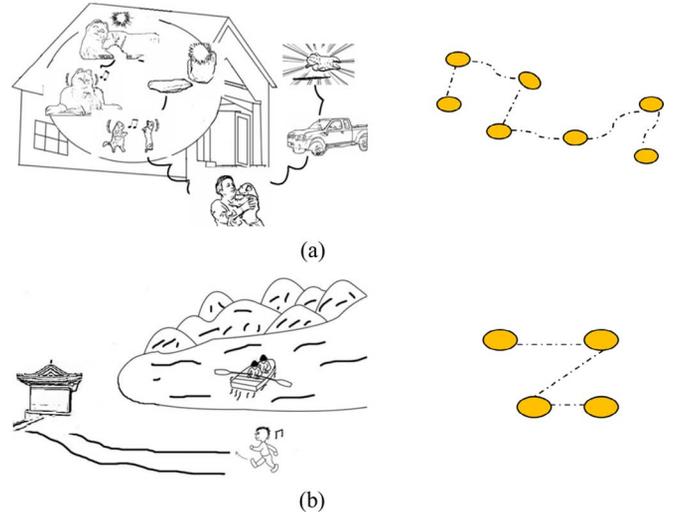


Fig. 3. Two-layer representation of the scene structure graph (SSG). (a) Example one. Left: the visualization layer; Right: the graph layer. (b) Example two. Left: the visualization layer; Right: the graph layer.

- *Annotation stage.* The user browses videos and annotates on shots using sketch forms in v including symbols and hand-drawing illustrations. Then an elemental $SSG(c, s)$ is generated for each clip c by using the sketch set s on c .
- *Authoring stage.* The user designs composite $SSGs$ with high quality $P(SSG(d, v), p)$ to visualize and edit the narrative structure for a particular authoring purpose. During the authoring process, the user can draw sketches or search in the elemental $SSGs$. Parts or whole structures in elemental $SSGs$ can then be reused for the new SSG design. We found that SSG reusability is particularly useful in video authoring.

IV. SKETCH-BASED ANNOTATIONS

Different primitives v can be used for video visualization $SSG(d, v)$, such as handwritten keywords, images or animations, etc. Since automatically summarizing videos with semantic information are computationally expensive, difficult, and tend to be very domain specific [43], in the proposed authoring environment, the tradeoff for this requirement is that we use sketch-based annotations as a kind of primitives to facilitate video structurization and visualization. Sketch-based annotations can enrich and extend the content of video. From the interaction point of view, taking annotations when watching video clips is a means of marking up in order to facilitate the interpretation and the understanding of its content.

A. Sketches for Annotations

Drawing annotations in a video clip has always been a time-consuming work to users, partly due to the sheer volume of video material that must be repeatedly viewed and recalled. In order to reduce the repeated work to an acceptable degree, we provide a user interface enabling users to sketch annotations with two forms:

- Annotation using keyframe-based sketches. The user selects several keyframes in clips. Then the coherent line

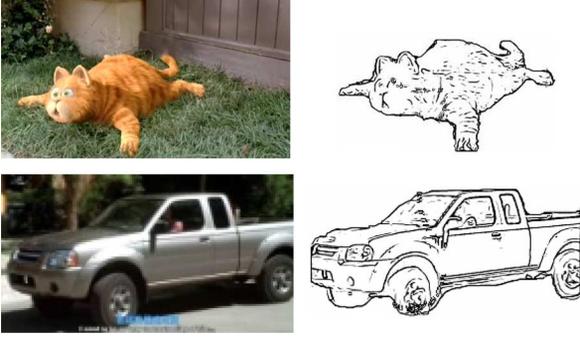


Fig. 4. Keyframe-based sketch generation [21]: the background structure is removed by gesture operations.

drawing algorithm [21] is applied to detect regions of interest and generate smooth and stylistic lines that serves as the sketches. Two examples are shown in Fig. 4. Users can further modify the sketches using gesture operations inherent in sketch-based interface, in which by mimicking traditional paper-and-pencil styles, the gestures identified by freehand sketch strokes include geometries (line, circle, arc, and free curves, etc.) and editing operations (selection, moving, rotation and zoom in/out, deletion, and cancellation, etc.).

- Annotation with sketchbook. If users do not like sketching by themselves, they can search, browse and pick up an appropriate sketch in a sketchbook collected in the system. The annotation in Fig. 2(a) is a sketch in the book under category “mountain”. The sketchbook is growing when more and more sketches are generated using the above two manners.

B. Elemental SSG Generation

Each video clip c could contain a set s of several sketches and we generate an elemental $SSG(c, s)$ for each clip c . For an optimized perception $P(SSG(c, s), p)$, the structure in $SSG(c, s)$ should be reused for later video authoring. In our study, the set p of parameters are spatio-temporal relations that determine the layout of sketches s in $SSG(c, s)$. The layout algorithm details are presented as follows.

First the nodes in SSG represented by sketches are resized based on their contributions to the whole clip. Currently we use the spanned time (duration) of shot containing that sketch as a measure of importance. Let β_i be the importance of the i th node. $contrib_i$ is used to describe the importance rate of the i th node:

$$contrib_i = \max \left(\frac{\beta_i}{\sum_{j=1}^n \beta_j}, 0.05 \right).$$

To avoid near zero contribution of very tiny sketches, a threshold value 0.05 is used. Denote the size of visualization canvas of SSG by $w \times h$, where w, h are width and height of the canvas. To make all nodes fit canvas size, we resize each sketch s_i with a ratio $r_i = scale \times contrib_i$, where $scale = (\alpha \times w \times h) / (\sum_{j=1}^n w_j \times h_j)$ and $\alpha \in (0, 1)$ represents the overall covering rate of sketches, e.g., $\alpha = 1$ means the whole map is completely covered by sketches.

To make the map nodes properly aligned in visualization canvas, we define the conceptual relations using a penalty function:

$$p = w_{\text{time}} \times p_{\text{time}} + w_{\text{rel}} \times p_{\text{rel}} + w_{\text{ovl}} \times p_{\text{ovl}} + w_{\text{cross}} \times p_{\text{cross}} \quad (1)$$

where $w_{\text{time}}, w_{\text{rel}}, w_{\text{ovl}}, w_{\text{cross}} \geq 0$ are weights that balance the contributions of $p_{\text{time}}, p_{\text{rel}}, p_{\text{ovl}}, p_{\text{cross}}$ (their meanings will be defined below) to the penalty function p and $w_{\text{time}} + w_{\text{rel}} + w_{\text{ovl}} + w_{\text{cross}} = 1$. In our current implementation, we use parameters $w_{\text{time}} = 0.35, w_{\text{rel}} = 0.25, w_{\text{ovl}} = 0.2, w_{\text{cross}} = 0.2$. In (1), p_{ovl} represents the overlay area among sketches and p_{cross} represents the number of cross-intersections of the story line (will be defined below). p_{time} in (1) represents the temporal constraints. We assume that in a properly aligned visualization canvas, the sketch with the earlier timestamp should lie to top-left of the later one as much as possible. The penalty for disordered time sequence is defined as

$$p_{\text{time}} = \sum_{i=1}^n \delta x_i + \sum_{i=1}^n \delta y_i$$

$$\delta x_i = \begin{cases} 0 & \text{if } x_i > x_{i-1} \\ 1 & \text{otherwise} \end{cases}, \delta y_i = \begin{cases} 0 & \text{if } y_i > y_{i-1} \\ 1 & \text{otherwise} \end{cases}$$

where (x_i, y_i) is the barycenter coordinate of sketch s_i . p_{rel} in (1) is a relation parameter representing spatial constraints. We use both the ratio between center distance and similarity between two nodes to represent their relation penalty:

$$p_{\text{rel}} = \sum_{i=1, i \neq j}^n \sum_{j=1}^n \frac{\text{dist}(\text{node}_i, \text{node}_j)}{\text{similarity}(\text{node}_i, \text{node}_j)}$$

where $\text{dist}(\text{node}_i, \text{node}_j)$ is the Euclidean distance between centers of nodes i and j and

$$\text{similarity}(\text{node}_i, \text{node}_j) = \text{shape_match}(\text{node}_i, \text{node}_j) + \lambda \|fmat_i \cdot fmat_j^T\|_1$$

$\text{shape_match}(\text{node}_i, \text{node}_j)$ is measured by the words-of-interest method [29], λ is a balance weight, and $fmat$ is a feature vector that contains various high-level semantic features:

$$fmat = (\text{indoor/outdoor}, \text{face/noface}, \text{day/night}, \text{cityscape/landscape})$$

where we define

$$\text{term}_i \cdot \text{term}_j = \begin{cases} 1 & \text{if } \text{term}_i = \text{term}_j \\ 0 & \text{otherwise} \end{cases}$$

and $\text{term}_i, \text{term}_j \in \{\text{indoor}, \text{outdoor}, \text{face}, \text{noface}, \text{day}, \text{night}, \text{cityscape}, \text{landscape}\}$.

The variables in the penalty function p are center positions (x_i, y_i) of all the sketches in s . Usually the number of sketches in each video clip is less than 10 and the dimension n of p is not large. Then the storage is not a serious constraint in numerical optimization. To minimize the function p , we use the direction-set method whose storage is of order n^2 .

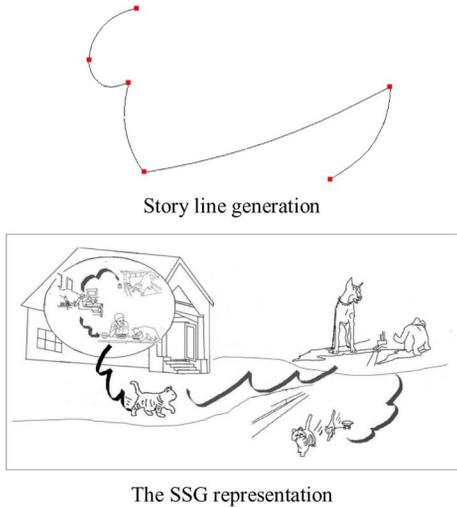


Fig. 5. Elemental SSG generation.

Story line generation. Given the locations $DP_i = (x_i, y_i), i = 1, \dots, n$, a cubic B-spline curve $C(t)$ is computed to pass through the map nodes. The control points $CP_j, j = 0, 1, \dots, n+1$ is found by solving the linear system:

$$\begin{bmatrix} 6 & 0 & 0 & & & 0 \\ 1 & 4 & 1 & 0 & & 0 \\ 0 & 1 & 4 & 1 & 0 & 0 \\ & & & \ddots & & \\ & & & & 1 & 4 & 1 \\ & & & & & 0 & 6 \end{bmatrix} \begin{bmatrix} CP_0 \\ CP_1 \\ \vdots \\ \vdots \\ CP_n \\ CP_{n+1} \end{bmatrix} = 6 \begin{bmatrix} DP_1 \\ DP_2 \\ \vdots \\ \vdots \\ DP_{n-1} \\ DP_n \end{bmatrix}.$$

For the start and end points of the curve, two additional constraints are given:

$$CP_0 = 2CP_1 - CP_2, \quad CP_{n+1} = 2CP_n - CP_{n-1}.$$

One example of elemental SSG generation is shown in Fig. 5. After generating an initial elemental SSG, users can further modify it using sketches with pen strokes. The elemental SSG is similar to schematic storyboard proposed in [10], but with a different purpose. Schematic storyboard is based on an extended frame layout (one kind of panorama) and is suitable for applications including video summarization, assembly instructions, and camera motion illustrations, etc. The elemental SSG is designed with a visualization layer and a graph layer. This two-layer representation is suitable for video authoring as demonstrated in the user experience, presented in Section VI.

V. VIDEO AUTHORING WITH SSGS

Based on sketch annotations, in a video authoring process, the user can sketch his/her idea using freeform strokes and the authoring environment infers the user's intent and executes the appropriate operations, such as searching for similar sketches, recommending related video clips, and manipulating (cut, paste and group) elemental SSGs into a new SSG, etc. During the interactive authoring process, a new, composite SSG is formed, which represents the narrative structure among different video clips.

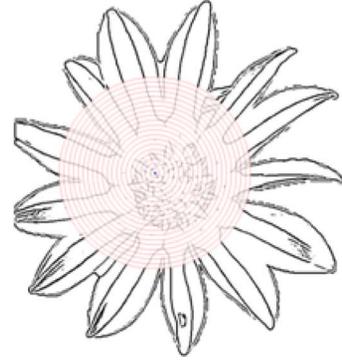


Fig. 6. Histogram of 20 radial bins centered at a sample point.

A. Sketch Similarity Based on User Profile

We use our previous work [29] to extract a set of feature vectors from an annotated sketch k , which is briefly summarized below. Given a sketch k , we first find a bounding rectangle $B(k)$ of k . Then 100 points are randomly sampled in $B(k)$. Let r be one fifth of the diagonal length of $B(k)$. For each sample point s , we locate a circle of length r centered at s . The circle is partitioned into 20 radial bins to form a histogram (Fig. 6). Let $\{p_i\}$ be the pixels representing the sketch image. Given a histogram at sample point s , a feature vector f_s is defined as

$$f_s(j) = \{\#p \in \text{bin}(j)\}, \quad j = 1, 2, \dots, 20$$

where $\#p$ is the number of pixels that fall into the different radial bins. Finally all the feature vectors are normalized with magnitude 1. The similarity $\text{Sim}(f_1, f_2)$ between two feature vectors is measured by $(f_1 \cdot f_2) / (\|f_1\| \|f_2\|)$.

Each annotated sketch contributes 100 feature vectors. Based on the bag-of-words (BoW) model in [40], we apply the K-means clustering on the feature vectors of all the annotated sketches to build a visual vocabulary, in which each visual word is a representative feature vector in a cluster. In [40], all the visual words in the vocabulary are of equal importance. In our study, we extract words-of-interest (WoI) from BoW according to user sketching history during the authoring process, based on a feature transfer technique proposed below.

Note that sketches in annotation and authoring stages may be drawn by different users. We use a Markov chain model to select WoI based on user sketching history at the authoring stage. First the visual vocabulary is considered as a finite state space of a Markov chain model. In principle, visual words with higher probability to occur in the user sketching history are selected as WoI. Let the user sketching history be represented as a weighted vector of visual words $W_i = \{n_i w_i\}, i = 1, 2, \dots, n_v$, where w_i is a visual word in the vocabulary, n_v is the size of the vocabulary, and n_i is the frequency of the visual word w_i appeared in the vocabulary. The spatial proximity of two visual words w_i, w_j is defined by

$$S_{ij} = \frac{1}{n_i n_j} \sum_{m=1}^{n_i} \sum_{n=1}^{n_j} \text{Sim}(f_m, f_n)$$

			
	0.155	0.111	0.081
	0.232	0.195	0.214
	0.258	0.201	0.169
	0.354	0.312	NA
	0.451	NA	NA

Fig. 7. Annotation sketches recommendation by dissimilarity ranking: the recommendation is in an incremental fashion by user iteratively refining his/her sketch. The recommended annotation sketches are shown in black numbers; the rejected sketches in the incremental refinement are shown in red numbers.

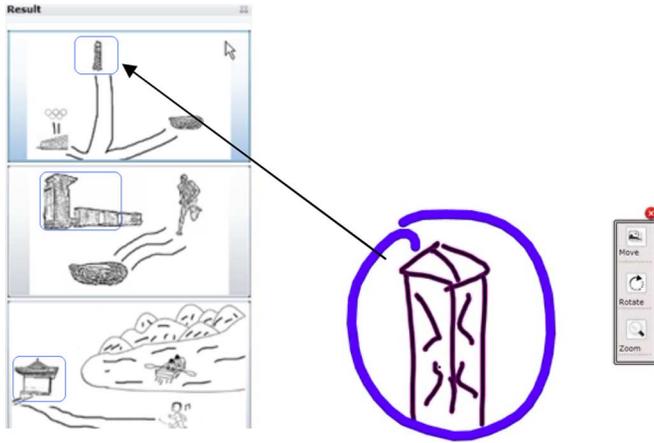


Fig. 8. Interface of sketch recommendation.

where f_m, f_n are the m th and n th instances of visual words w_i, w_j , respectively. The higher similarity two visual words have, the more possibilities a feature transfer from one visual word to the other. We define the visual word transfer probability matrix as

$$\mathbf{P} = [p_{ij}] = \left[\frac{S_{ij}}{\sum_{j=1}^{n_v} S_{ij}} \right]$$

where n_v is the total number of visual words.

The conditional probability that visual word w_i occurs in the user sketching history is defined to be $(n_i)/(n_f)$, where n_f is the number of features appeared in the user sketching history. Then the initial state distribution of the Markov chain model can be formulated as

$$\pi(0) = \left\{ \frac{n_i}{n_f}, i = 1, 2, \dots, n_v \right\}.$$

By using visual words, a sketch is similar to a textual document. It was proved in [16] that a Markov chain used for representing such a document is ergodic. So the limit state distribution π^* exists and we run a sufficient large number of steps

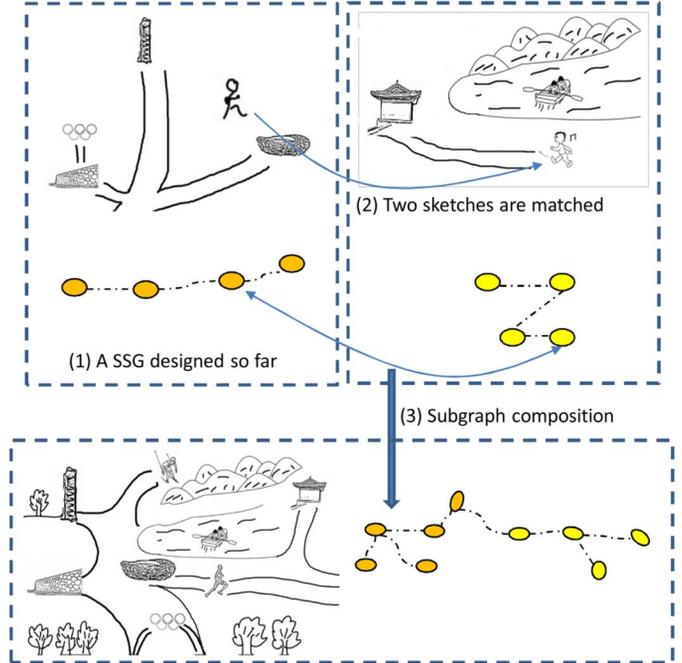


Fig. 9. Sketch-matching-based SSG composition.



Fig. 10. Wacom 17-inch Tablet with 1024 × 768 pixels resolution is used with a HP Compaq computer (Intel Core 2 CPU 2.13 GHz) running Windows XP.

(100 in our experiments) to obtain π^* . Based on π^* , the visual words are sorted based on the probability of occurrence and the top 30% visual words in the vocabulary are selected as WoI and the remainders are non-WoI. Now any sketch can be represented by a weighted vector of WoI $W^I = \{n_i w_i^I, i = 1, 2, \dots, n^I\}$ and a vector of non-WoI $W^{nI} = \{n_j w_j^{nI}, j = 1, 2, \dots, n^{nI}\}$. Let $F^I = \{n_i, i = 1, 2, \dots, n^I\}$ and $F^{nI} = \{n_j, j = 1, 2, \dots, n^{nI}\}$ be the frequency vectors of WoI and non-WoI, respectively. In the proposed authoring environment, the dissimilarity of two sketches k_p, k_q is defined by the distance metric

$$D(k_p, k_q) = \alpha \|F_p^{nI} - F_q^{nI}\|_2 + (1 - \alpha) \|F_p^I - F_q^I\|_2 \quad (2)$$

where F_p^{nI}, F_q^{nI} are the frequency vectors of non-WoI of sketches k_p, k_q respectively, and F_p^I, F_q^I are the frequency

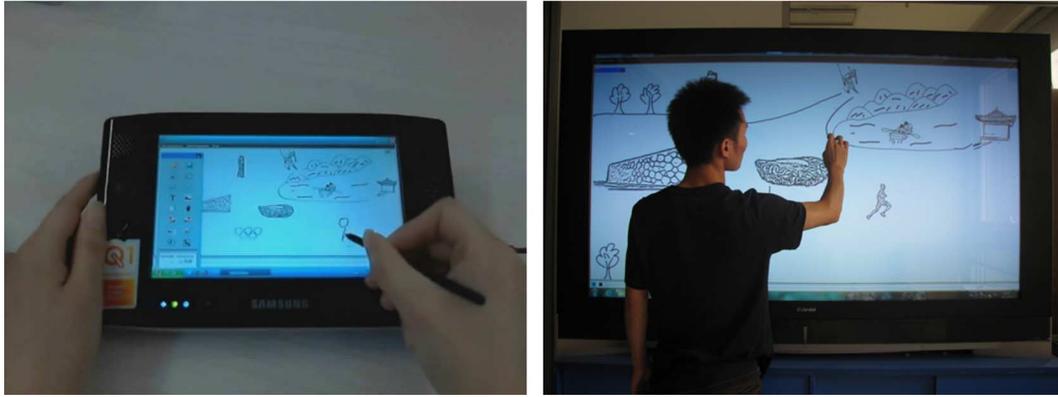


Fig. 11. Interactive devices with different displaying scales. Left: the 7-inch tablet (Samsung UMPC). Right: 71 inch interactive whiteboard supporting touch operations.



Fig. 12. Video annotations using keywords (left), keyframes (middle), and sketches (right).

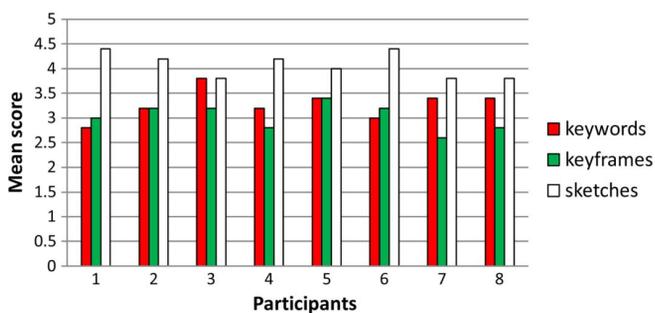


Fig. 13. Mean scores of video annotations using keywords, keyframes, and sketches, respectively, based on subject evaluation.

vector of WoI of k_p, k_q , respectively. We use $\alpha = 0.2$ in all our experiments.

B. Video Recommendation and SSG Composition

In the proposed authoring environment, once video contents are annotated and visualized by sketches, the user begins to construct a new SSG for realizing a rough authoring idea. At a particular node of SSG, the user can sketch a meaningful drawing that is representative for a particular authoring purpose. The authoring environment will search the database by matching

the user-input sketches with annotated sketches that are hyper-linked to the clips (Figs. 7 and 8). From the top matched sketches (e.g., within 10% highest ranks), the authoring environment automatically recommends the most related candidate video clips to the user. The candidate list is displayed in a panel in the interface (Fig. 8), from which the user can view, drag and lay down desired ones into the node of SSG.

Sketch matching using metric (2) recommends related video clips for a particular node of the SSG under design. The elemental SSG of that video clip can then be reused for the designed SSG. Standard graph techniques [15] are adopted to enhance the reusability of elemental SSGs:

- Classical radial layout algorithms [15] are used to dynamically adjust and visualize the graph layer in SSGs.
- Diverse graph operations are supported: select subgraphs, modify nodes' positions for a better arrangement, add or delete edges to modify the spatio-temporal relations, etc.
- Several subgraphs can be combined to make a composite SSG graph.

The composite SSG graph should be connected and we dynamically monitor this property using the graph scanning algorithm that runs in linear time in terms of the number of graph nodes.

One example of SSG composition is shown in Fig. 9. First, user designs by sketching a SSG and the authoring environment maintains the two layers of that SSG (top-left in Fig. 9). At a particular node, user matches that node’s sketch to the database and the authoring environment recommends some most similar sketches. Given the video clip containing the matched sketch (top-right in Fig. 9), the user selects subgraphs and composites two SSG subgraphs into a new SSG (bottom in Fig. 9). The user experience presented in Section VI shows that the sketch-based video annotation and visualization can reduce users’ cognitive load during the authoring process.

VI. USER EXPERIENCE

The presented sketch-based authoring environment aims to provide an efficient and intuitive tool, through an integration of the sketch-based annotations and SSG representation of narrative structures in a video authoring process. The authoring environment has been tested in devices with diverse displaying scales, including a Toshiba Tablet PC in Fig. 1, a Wacom Tablet with a HP Compaq computer in Fig. 10, an ultra-mobile personal computer (UMPC) in the left of Fig. 11 (for mobile computing) and an interactive whiteboard in the right of Fig. 11 (for a large-scale representation). A demo video showing the authoring process with different interactive devices is submitted along with this paper.

To test the usability and gain feedback about the functionality of the presented sketch-based authoring environment, three user studies have been conducted. For a consistent evaluation, the Wacom 17-inch Tablet was used in all three studies. A UMPC and an interactive whiteboard were also used in the third study. The first study evaluated different video annotation methods, including typed keywords, keyframes, and sketches. The second study evaluated different video content visualization methods, using keywords, keyframes, and sketches, respectively. The third study evaluated the video authoring process by comparing the commercial system *Adobe Encore* and our sketch-based environment.

A. Video Annotations With Keywords, Keyframes, and Sketches

Participants. Sixteen participants from a Chinese university were invited, including 7 females and 9 males. Their ages range from 23 to 37. They were divided into two groups of equal size.

Methods. Five video clips were provided to them, whose lengths ranged from 2 to 8 min. One group was asked to annotate these video clips using typed keywords, keyframes, and sketches, respectively (Fig. 12). After annotations, the other group was asked to evaluate how well each type of annotations characterizes the clips, by rating with “excellent”, “good”, “fair”, “poor”, and “bad”. We use scores from 5 to 1, which is a variant of the ITU-R five-point quality scale [19]. At the end of this experiment, an informal interview was made to participants about how they felt about the flexibility and usability of different annotation methods.

Results. We collected the subjects’ evaluation and averaged the scores over five clips. The mean score results are presented in Fig. 13, which shows that sketch-based annotations have the highest scores. A repeated measure ANOVA was conducted and

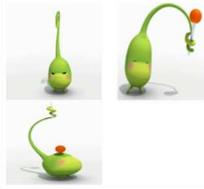
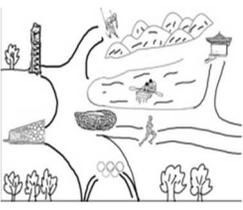
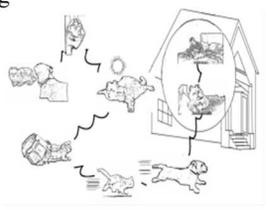
Keyframes of video clips	Keyword description	SSG description
	A subject, Music, Dance, Shake	
	Basketball, Shoot, Run	
	Park, Lake, Boat, Bird nest, Water cube, Tower, Pavilion, Mountain, Sprint	
	Subjects, Fight, Falling down, Dancing cat, Dog, Outside	
	Subjects, Teapot cups, Thinking, Dreaming	
	Subjects, Cat’s falling down, Running, Cat, Dog, Outside, Sleep	

Fig. 14. Video visualization using keywords, keyframes, and SSG-based representations.

the results showed that the main effect of different annotation methods was significant, $F(2, 14) = 26.052, p < 0.01$. The results of the pairwise comparisons with Bonferroni correction showed that

- There was significant difference between sketch ($M = 4.075, SD = 0.260$) and keyword ($M = 3.275, SD = 0.301$) annotations, $p = 0.013$.



Fig. 15. Video authoring using Adobe Encore (left) and SSG-based representation (right).

- There was also significant difference between sketch ($M = 4.075$, $SD = 0.260$) and keyframe ($M = 3.025$, $SD = 0.271$) annotations, $p < 0.001$.

About the informal interview, 75% participants (12 of 16) thought that keywords are intuitive for video annotation, while keyframe- and sketch-based annotations convey more visual information.

B. Video Visualization With Keywords, Keyframes, and SSGs

Participants. The same set of 16 participants were invited in this experiment. They were familiar with different annotations after the first experiment.

Methods. Visualization of six video clips (Fig. 14), using keywords, keyframes, and SSGs, respectively, was presented to the participants. After presentation, they were asked to rank how much the three visualization methods match the video contents. For each video clip, the five-point scores evaluated by participants were averaged into a mean score.

Results. For six video clips, the mean score vectors of keyword-, keyframe-, and SSG-based visualizations are (2.8, 2.5, 3.0, 2.8, 2.3, 2.4), (3.0, 3.5, 3.3, 3.4, 3.3, 3.4), and (3.4, 4.3, 4.0, 4.1, 3.7, 3.6), respectively. In this experiment, SSG-based visualization has the highest score. A repeated measure ANOVA was conducted and the results showed that the main effect of different visualization methods was significant, $F(2, 10) = 37.825$, $p < 0.01$. The results of the pairwise comparisons with Bonferroni correction showed that

- There was significant difference between SSG-based ($M = 3.850$, $SD = 0.339$) and keyword-based ($M = 2.633$, $SD = 0.273$) visualizations, $p = 0.002$.
- There was also significant difference between SSG-based ($M = 3.850$, $SD = 0.339$) and keyframe-based ($M = 3.317$, $SD = 0.172$) visualizations, $p = 0.008$.

At the end of experiment, an informal interview with participants revealed that more than a half of participants thought that sketch-based visualization using SSG represents more contextual information than the other two representations.

C. Video Authoring by Adobe Encore and SSG Representation

We hypothesize that based on the understanding of two-layer representation in SSG, users can easily author videos in a cogni-

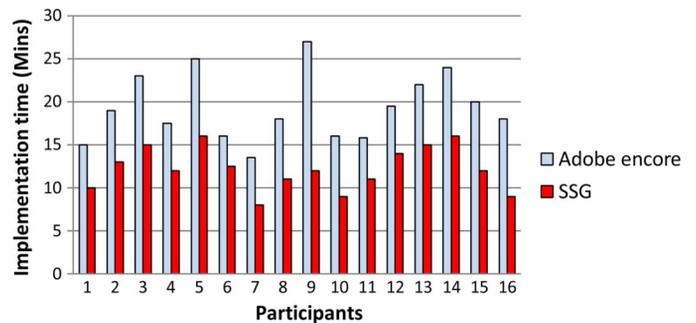


Fig. 16. Implementation time of video authoring using Adobe Encore and SSG-based representation.

tion optimization manner. The following experiment was conducted to evaluate SSG-based authoring process by comparing with the Adobe Encore authoring software [1].

Participants. The same set of 16 participants were invited in this experiment. They were familiar with sketch-based annotation and visualization.

Method. The participants had been trained for Adobe Encore CS4 by watching the tutorial demo video. To use the SSG-based authoring environment, Wacom 17-inch Tablet (Fig. 10) was used as the platform. The test database includes 30 video clips downloaded from the Internet, in which 6 are about the China national stadium. Sketch-based annotations had been input in the database. The authoring task is to create an interactive tour guide that introduces athletic sports in the Olympic Park at Beijing. Given this particular task, participants were asked to find clips related to the task and structurize them in any form, using Adobe Encore and SSG representation, respectively (Fig. 15). After completing the authoring task, the participants also watched the demo videos of SSG-based authoring using devices of UMPC (left in Fig. 11) and interactive whiteboard (right in Fig. 11). Then a questionnaire was presented to the participants to record their opinions about the authoring process. Fig. 17 shows the questionnaire in which most items are self-explanatory. For item ⑥, consistency/inconsistency checks whether the content layout and interaction behaviors in different operating interfaces are consistent or not. For item ⑧, satisfaction means that the user experience is good.

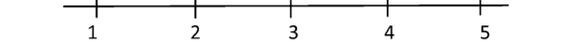
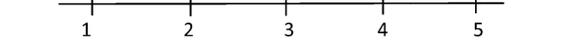
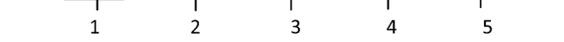
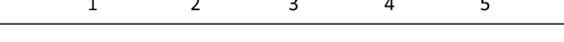
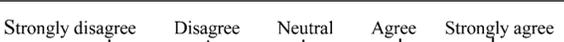
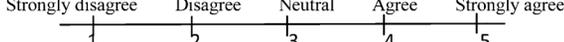
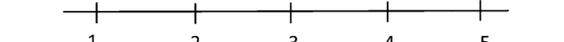
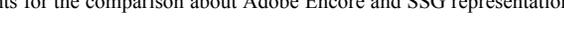
Please circle the number that represents how you feel about the software you have used during the process of implementing the task.	
① It is efficient to obtain the required video clips by capturing the main content.	Strongly disagree Disagree Neutral Agree Strongly agree 
② It is very ease to have an overall structure about the scenario for authoring videos.	Strongly disagree Disagree Neutral Agree Strongly agree 
③ It is easy to change the order of video clips.	Strongly disagree Disagree Neutral Agree Strongly agree 
④ It is simple to use.	Strongly disagree Disagree Neutral Agree Strongly agree 
⑤ It is fun to use.	Strongly disagree Disagree Neutral Agree Strongly agree 
⑥ I do not notice any inconsistencies when I use it.	Strongly disagree Disagree Neutral Agree Strongly agree 
⑦ The software is user-friendly.	Strongly disagree Disagree Neutral Agree Strongly agree 
⑧ I was satisfied with the process.	Strongly disagree Disagree Neutral Agree Strongly agree 

Fig. 17. Questionnaire to the participants for the comparison about Adobe Encore and SSG representation.

Results. We record the total time of participants used to complete the authoring task. The results are presented in Fig. 16. A repeated measure ANOVA was conducted and showed that the main effect of the authoring methods was significant, $F(1, 15) = 117.149$, $p < 0.001$, i.e., authoring using SSG representation ($M = 12.22$, $SD = 2.51$) achieves significantly better time efficiency than Adobe Encore ($M = 19.33$, $SD = 3.901$). At the end of experiment, the participants completed the questionnaire in Fig. 17. The results are summarized below:

- 94% of participants (15 of 16) gave positive feedback about sketch-based interface and SSG-based authoring process.
- 88% of participants (14 of 16) ranked SSG as a useful and convenient method for understanding the overall structure during authoring process.
- 69% of participants (11 of 16) thought that sketch-based operations are interesting and fun.
- 81% of participants (13 of 16) gave positive feedback about the portable device (UMPC shown in the left of Fig. 11) using sketch-based interface.
- 56% of participants (9 of 16) gave positive feedback about sketch-based authoring with the large interactive whiteboard shown in the right of Fig. 11.

VII. CONCLUSION

Sketching is prevalent at the design process, and common users intend to adopt freehand sketching as the main method

of communicating their ideas. In this paper, we present an interactive video authoring environment which uses sketches to facilitate the annotation and visualization of video contents. From the viewpoint of knowledge engineering, annotation by sketches can be regarded as knowledge extraction and representation, and video content visualization and reorganization using SSG can be regarded as knowledge creation and reuse. In the presented authoring environment, SSG with two-layer representation and simple sketching tools are provided. Three user studies have been conducted, showing that with the aid of SSG and sketching tools, users can easily annotate and author videos in a way which helps improve user experience in an early-stage design process.

Limitations of the presented method. Currently the proposed interactive authoring environment only supports simple sketching styles. It is difficult to have a precise understanding about complicated sketches. Although the two-layer integrated representation of SSG helps alleviate some of these problems, in the authoring process, users still prefer to provide sketches of different complexities based on complexities of authoring tasks. Future research will extend this work to cover sketch understanding with domain knowledge and support adaptive sketching based on a user attention model akin to the one in [31].

ACKNOWLEDGMENT

The authors would like to thank the reviewers for their valuable comments that help improve this paper.

REFERENCES

- [1] "Authoring DVDs with Adobe Encore CS4," in *Adobe Premiere Pro CS4 Classroom in a Book*. Berkeley, CA: Adobe Press, 2008, ch. 21, Adobe Creative Team.
- [2] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 6, pp. 734–749, 2005.
- [3] D. Bulterman and L. Hardman, "Structured multimedia authoring," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 1, no. 1, pp. 89–109, 2005.
- [4] D. Bulterman and J. Jansen, "Synchronized multimedia integration language (SMIL 3.0)," W3C Recommendation, 2008. [Online]. Available: <http://www.w3.org/TR/smil>.
- [5] S. F. Chang, W. Chen, H. J. Men, H. Sundaram, and D. Zhong, "A fully automated content-based video search engine supporting spatio-temporal queries," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 5, pp. 602–615, 1998.
- [6] B. W. Chen, J. C. Wang, and J. F. Wang, "A novel video summarization based on mining the story-structure and semantic relations among concept entities," *IEEE Trans. Multimedia*, vol. 11, no. 2, pp. 295–312, 2009.
- [7] J. P. Collomosse, G. McNeill, and Y. Qian, "Storyboard sketches for content based video retrieval," in *Proc. Int. Conf. Computer Vision (ICCV'09)*, 2009, pp. 245–252.
- [8] A. Csinger, "User Models for Intent-Based Authoring," Ph.D. dissertation, Univ. British Columbia, Vancouver, BC, Canada, 1996.
- [9] S. L. Feng, R. Manmatha, and V. Lavrenko, "Multiple Bernoulli relevance models for image and video annotation," in *Proc. Computer Vision and Pattern Recognition (CVPR'04)*, 2004, pp. 1002–1009.
- [10] D. B. Goldman, B. Curless, D. Salesin, and S. M. Seitz, "Schematic storyboarding for video visualization and editing," *ACM Trans. Graph. (Proc. SIGGRAPH'06)*, vol. 25, no. 3, pp. 862–871, 2006.
- [11] D. B. Goldman, C. Gonterman, B. Curless, D. Salesin, and S. M. Seitz, "Video object annotation, navigation, and composition," in *Proc. 21st Annu. ACM Symp. User Interface Software and Technology*, 2008, pp. 3–12.
- [12] V. Goel, *Sketches of Thought*. Cambridge, MA: MIT Press, 1995.
- [13] R. L. Guimaraes, P. Cesar, and D. Bulterman, "Creating and sharing personalized time-based annotations of videos on the web," in *Proc. DocEng'10*, 2010, pp. 27–36.
- [14] K. Harada, E. Tanaka, R. Ogawa, and Y. Hara, "Anecdote: A multimedia storyboarding system with seamless authoring support," in *Proc. ACM Multimedia '96*, 1996, pp. 341–351.
- [15] I. Herman, G. Melancon, and M. S. Marshall, "Graph visualization and navigation in information visualization: A survey," *IEEE Trans. Vis. Comput. Graphics*, vol. 6, no. 1, pp. 24–43, 2000.
- [16] E. Hoenkamp and D. W. Song, "The document as an ergodic Markov chain," in *Proc. ACM SIGIR'04*, 2004, pp. 496–497.
- [17] X. Hua, S. Li, and H. Zhang, "Video booklet," in *Proc. IEEE ICME*, 2005.
- [18] T. Igarashi, S. Matsuoka, S. Kawachiya, and H. Tanaka, "Pegasus: A drawing system for rapid geometric design," in *Proc. ACM CHI'98*, Los Angeles, CA, 1998, pp. 24–29, ACM Press.
- [19] *Methodology for the Subjective Assessment of the Quality of Television Images*, ITU-R Recommendation BT.500-11, International Telecommunication Union: Geneva, 2002.
- [20] I. Jiebo, C. Papin, and K. Costello, "Towards extracting semantically meaningful keyframes from personal video clips: From humans to computers," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 2, pp. 289–301, 2009.
- [21] H. Kang, S. Lee, and C. Chui, "Coherent line drawing," in *Proc. ACM Symp. Non-Photorealistic Animation and Rendering*, San Diego, CA, 2007, pp. 43–50.
- [22] G. Kanizsa, *Organization in Vision: Essays in Gestalt Perception*. New York: Praeger, 1979.
- [23] J. J. Laviola, "Sketch-based interfaces: Techniques and applications," SIGGRAPH 2007 Course 3, 2007.
- [24] M. S. Lew, N. Sebe, C. Djeraba, and R. Jain, "Content-based multimedia information retrieval: State of the art and challenges," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 2, no. 1, pp. 1–19, 2006.
- [25] J. Lin, M. W. Newman, J. I. Hong, and J. A. Landay, "DENIM: Finding a tighter fit between tools and practice for web site design," in *Proc. ACM CHI'00*, 2000, pp. 510–517.
- [26] Y. J. Liu, K. L. Lai, G. Dai, and M. M. F. Yuen, "A semantic feature model in concurrent engineering," *IEEE Trans. Autom. Sci. Eng.*, vol. 7, no. 3, pp. 659–665, 2010.
- [27] Y. J. Liu, C. X. Ma, and D. L. Zhang, "Easytoy: A plush toy design system using editable sketch curves," *IEEE Comput. Graphics Appl.*, vol. 31, no. 2, pp. 49–57, 2011.
- [28] Y. J. Liu, Z. Q. Chen, and K. Tang, "Construction of ISO-contours, bisectors, and Voronoi diagrams on triangulated surfaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1502–1517, 2011.
- [29] X. Luo, W. J. Guo, Y. J. Liu, C. X. Ma, and D. W. Song, "A words-of-interest model of sketch representation for image retrieval," in *Proc. 2011 Asian Conf. Design and Digital Engineering*, 2011.
- [30] C. X. Ma, Y. J. Liu, H. Y. Yang, D. X. Teng, H. A. Wang, and G. Z. Dai, "KnitSketch: A sketch pad for conceptual design of 2D garment patterns," *IEEE Trans. Autom. Sci. Eng.*, vol. 8, no. 2, pp. 431–437, 2011.
- [31] Y. F. Ma, L. Lu, H. Zhang, and M. Li, "A user attention model for video summarization," in *Proc. ACM Multimedia '02*, 2002, pp. 533–542.
- [32] Y. F. Ma and H. Zhang, "Video snapshot: A bird view of video sequence," in *Proc. Int. Multimedia Modeling Conf.*, 2005, pp. 94–101.
- [33] B. S. Manjunath, P. Salembier, and T. Sikora, Eds., *Introduction to MPEG-7: Multimedia Content Description Interface*. New York: Wiley, 2002.
- [34] T. Mei, B. Yang, X. S. Hua, and S. Li, "Contextual video recommendation by multimodal relevance and user feedback," *ACM Trans. Inf. Syst.*, vol. 29, no. 2, 2011, Article No. 10.
- [35] E. Moxley, T. Mei, and B. S. Manjunath, "Video annotation through search and graph reinforcement mining," *IEEE Trans. Multimedia*, vol. 12, no. 3, pp. 184–193, 2010.
- [36] E. D. Mynatt, T. Igarashi, W. K. Edwards, and A. Lamarca, "Flatland: New dimensions in office whiteboards," in *Proc. CHI'99 Human Factors in Computing Systems*, New York, 1997, pp. 45–54, ACM Press.
- [37] A. Rav-Acha, Y. Pritch, and S. Peleg, "Making a long video short: Dynamic video synopsis," in *IEEE Proc. CVPR*, 2006, pp. 435–441.
- [38] F. Shipman, A. Girgensohn, and L. Wilcox, "Authoring, viewing, and generating hypervideo: An overview of hyper-hitchcock," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 5, no. 2, 2008, article no. 15.
- [39] E. Simoncelli and B. Olshausen, "Natural image statistics and neural representation," *Annu. Rev. Neurosci.*, vol. 24, pp. 1193–1216, 2001.
- [40] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. Int. Conf. Computer Vision (ICCV'03)*, 2003, pp. 1470–1477.
- [41] I. E. Sutherland, "Sketchpad—A man-machine graphical communication system," in *Proc. Spring Joint Computer Conf.*, 1963, pp. 329–346.
- [42] Y. Taniguchi, A. Akutsu, and Y. Tonomura, "PanoramaExcerpts: Extracting and packing panoramas for video browsing," in *Proc. ACM Int. Conf. Multimedia*, 1997, pp. 427–436.
- [43] B. T. Truong and S. Venkatesh, "Video abstraction: A systematic review and classification," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 3, no. 1, 2007.
- [44] H. Ueda, T. Miyatake, S. Summino, and A. Nagasaka, "Automatic structure visualization for video editing," in *Proc. Human Factors in Computing Systems*, 1993, pp. 137–141.
- [45] G. van Rossum, J. Jansen, K. Mullender, and D. Bulterman, "CMIFed: A presentation environment for portable hypermedia documents," in *Proc. ACM Multimedia '93*, 1993, pp. 183–188.
- [46] M. Wang, X. S. Hua, J. H. Tang, and R. Hong, "Beyond distance measurement: Constructing neighborhood similarity for video annotation," *IEEE Trans. Multimedia*, vol. 11, no. 3, pp. 465–476, 2009.
- [47] C. Xu, J. Wang, H. Lu, and Y. Zhang, "A novel framework for semantic annotation and personalized retrieval of sports video," *IEEE Trans. Multimedia*, vol. 10, no. 3, pp. 421–436, 2008.
- [48] M. Yueng and B. Yeo, "Video visualization for compact presentation and fast browsing of pictorial content," *IEEE Trans. Circuit Syst. Video Technol.*, vol. 7, no. 5, pp. 771–785, 1997.



Cui-Xia Ma received the Ph.D. degree from the Institute of Software, Chinese Academy of Sciences, Beijing, China, in 2003.

She is an Associate Professor with the Institute of Software, Chinese Academy of Sciences. Her research interests include human-computer interaction and multimedia computing.



Yong-Jin Liu received the Ph.D. degree from the Hong Kong University of Science and Technology, Hong Kong, in 2003.

He is an Associate Professor with the Department of Computer Science and Technology, Tsinghua University, Beijing, China. His research interests include computer graphics, computer vision, and computer-aided design.



Dong-Xing Teng received the Ph.D. degree from Tsinghua University, Beijing, China, in 2001.

He is an Associate Professor with the Institute of Software, Chinese Academy of Sciences, Beijing, China. His research interests include information visualization and human-computer interaction.



Hong-An Wang received the Ph.D. degree from the Institute of Software, Chinese Academy of Sciences, Beijing, China, in 1999.

He is a Professor of the Institute of Software, Chinese Academy of Science. His research interests include real-time intelligence and user interface.



Guo-Zhong Dai received the B.S. degree from the University of Science and Technology of China, Hefei, China, in 1968.

He is a Professor with the Institute of Software, Chinese Academy of Sciences, Beijing, China. His research interests include human-computer interaction and computer graphics.