

# Collaborative Interaction for Videos on Mobile Devices Based on Sketch Gestures

Jin-Kai Zhang<sup>1</sup> (张金凯), Cui-Xia Ma<sup>2,\*</sup> (马翠霞), *Member, IEEE*, Yong-Jin Liu<sup>1</sup> (刘永进), *Member, IEEE*, Qiu-Fang Fu<sup>3</sup> (付秋芳), and Xiao-Lan Fu<sup>3</sup> (傅小兰)

<sup>1</sup>*Department of Computer Science and Technology, Tsinghua University, Beijing 100190, China*

<sup>2</sup>*Institute of Software, Chinese Academy of Sciences, Beijing 100190, China*

<sup>3</sup>*Institute of Psychology, Chinese Academy of Sciences, Beijing 100190, China*

E-mail: zhangjinkai999@gmail.com; cuixia@iscas.ac.cn; liuyongjin@tsinghua.edu.cn; {fuqf, fuyl}@psych.ac.cn

Received May 5, 2013; revised August 9, 2013.

**Abstract** With the rapid progress of the network and mobile techniques, mobile devices such as mobile phones and portable devices, have become one of the most important parts in common life. Efficient techniques for watching, navigating and sharing videos on mobile devices collaboratively are appealing in mobile environment. In this paper, we propose a novel approach supporting efficiently collaborative operations on videos with sketch gestures. Furthermore, effective collaborative annotation and navigation operations are given to interact with videos on mobile devices for facilitating users' communication based on mobile devices' characteristics. Gesture operation and collaborative interaction architecture are given and improved during the interactive process. Finally, a user study is conducted showing that the effectivity and collaborative accessibility of video exploration is improved.

**Keywords** interface, sketch gesture, mobile device, collaborative interaction

## 1 Introduction

Owing to the higher transmission rate, cheaper storage, and improved video-sharing techniques, digital videos are becoming available at an explosive increasing rate. Meanwhile, rapid growth of mobile devices has resulted in an increase in users' expectations to get and share convenient access to relevant digital video content anywhere and anytime, which means that video operation tasks are usually performed at different locations. In this scenario, collaboration between users becomes important. The challenging issues in this mobile collaboration application are that, mobile devices' small screens restrict the amount of information shown to users and constraints on the input mechanisms lead to a slow and inconvenient interface between users and devices<sup>[1]</sup>. The manner in which the video content is presented for access and communication has become an important task for the systems and users. During the process of browsing videos, users prefer to anno-

tate comments efficiently, and then to communicate conveniently with intuitive interaction. In this paper we propose a novel approach of collaboratively annotating video content using a multi-view representation method, which supports efficient video browsing using sketch gestures.

Video annotations are useful for allowing users to create personalized comments and facilitating video navigation and communication. Some approaches have been presented which use the texts and frames as annotations<sup>[2-3]</sup>. Observing that sketch annotations provide more intuitive and direct manipulation for commenting the videos with freeform hand-drawing strokes, in this paper, we propose sketches for annotating video content as well as words.

Traditional video navigation modes based on linear timeline support less interaction. It is not convenient for users to interact and communicate collaboratively with each other using the timeline to get the main idea or locate some specific scenes in a video. However,

---

Short Paper

This work was supported by the National Basic Research 973 Program of China under Grant No. 2011CB302205, the National Natural Science Foundation of China under Grant Nos. 61173058 and 61272228, and the National High Technology Research and Development 863 Program of China under Grant Nos. 2012AA011801 and 2012AA02A608. The work of Liu YJ was supported in part by the Program for New Century Excellent Talents in University of China under Grant No. NCET-11-0273 and funded by Tsinghua National Laboratory for Information Science and Technology (TNList) Cross-Discipline Foundation.

\*Corresponding Author

©2013 Springer Science + Business Media, LLC & Science Press, China

sketch annotations proposed in the paper enhance the index of video content and support efficiently interacting with annotations using gestures. Imaging that you and your friends are watching video clips from different locations, you would like to share some comments. In general, what would you do? For example, you may call her/him or use some instant message tool to keep contact while you both are broken away from the video player interface. Obviously those comments are difficult to be used for later process. However, if you can communicate with each other just on the video player and draw or write down something directly using freehand drawing, it will be easier and more intuitive to send and access information. Furthermore, those comments can be used for video navigation and communication between users.

In this paper we propose gesture-based sketch interaction supporting collaborative operations with videos on mobile devices. The main contributions include: 1) a novel approach that uses a sketch interface to help users annotate videos in collaboration; 2) a collaborative interaction architecture for real-time communication using sketches and words; 3) sketch gestures for efficiently navigating videos instead of traditional screen buttons. A user study is elaborated and users value high for the proposed interface and report better experience by using collaborative operations and sketch gestures to implement certain tasks.

## 2 Related Work

Researchers have conducted many studies about video sharing and navigating on mobile devices. The Nokia research center conducted a two-week trial in the student community to find out the trend of capturing and sharing videos with mobile phones<sup>[4]</sup>. The result showed that a user-friendly interface of capturing and sharing videos was solicited to express and organize users' design ideas efficiently. Dezfali *et al.*<sup>[5]</sup> proposed a mobile video sharing system, which could support users in sharing the live video streams with this system in-situ. Hürst *et al.*<sup>[6]</sup> proposed a pen-based navigation of videos on PDAs and cell phones which removes most of the function elements on the screen and supports users in a more natural way to navigate videos on mobile devices. Lee *et al.*<sup>[7]</sup> proposed a gesture-based interaction method for videos on mobile devices. Video annotations can provide valuable information for medial understanding<sup>[8-9]</sup>. In previous work<sup>[10-11]</sup>, captions, keywords or key frames were often used for video annotations. All of these researches focus on individual video sharing and navigating, while little attention has been paid on collaborative interaction for videos on mobile devices.

Sketching is a common and convenient method for depicting visual world, describing viewpoints and retrieving shapes<sup>[12-14]</sup>. Sketch-based interfaces have been successfully applied in many applications, such as architecture design<sup>[15-16]</sup>, modeling<sup>[17]</sup> and editing<sup>[18-20]</sup>. Furthermore, sketch representation for annotating and visualizing video content is regarded as an efficient and effective tool<sup>[21]</sup>. Ma *et al.*<sup>[22]</sup> proposed a system to annotate on the videos with sketches, which could help users present a sketch map of videos with meaningful sketches among different views. Collaboration systems have been applied to many applications in multimedia domains. Davis *et al.*<sup>[23]</sup> proposed a collaborative system named NotePals, which is a note sharing system that gives group members easy access to each other's experiences through their personal notes. All of the group members can write down their own notes and upload those to the shared repository. Beth *et al.*<sup>[24]</sup> introduced a system called Classroom Presenter based on Tablet PC. The Classroom Presenter system allows users to integrate PowerPoint slides with high quality pen-based writing and to separate the instructor's view of the materials from the students' view. This allows more natural and interactive development of class concepts and content. Collaboration operations as mentioned above have been mature and extensively used in office systems<sup>[25-26]</sup>. It would be meaningful to apply collaboration systems to video sharing. In this paper, we propose collaborative interaction with videos on mobile devices, which could help people navigate videos in an efficient way by gestures and share the interesting regions with sketches.

## 3 Collaborative Interaction Based on Sketch Gestures

### 3.1 Collaborative Annotating Based on Sketches

Cooperation appears anywhere in our daily life such as playing music, debate, chat or drawing figures. However, most existing video operation systems on mobile devices do not support collaborative annotating and navigating. In this subsection, we propose a sketch interface, which supports collaboratively annotating videos. Fig.1 shows the sample of collaborative annotating (different users draw sketches annotating the video content).

Most video players mainly consist of the main screen and linear timeline. Considering the limited screen space of mobile devices, compact representation and gesture-based interaction are appearing. In this case, we introduce the multi-view representation and gestures to facilitate navigating videos.

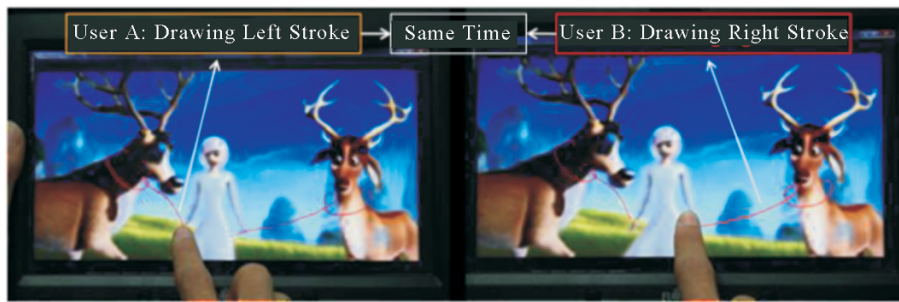


Fig.1. Collaborative annotating.



Fig.2. Multi-view representation sample during the collaboration. (a) User 1: video view annotating sketch. (b) User 2: annotation view.

Multi-view representations (Fig.2) for annotations provide different scales for users' understanding and organizing video content<sup>[27-28]</sup>. Furthermore, the comments embedded in the videos may be annoying to users who concentrate on video content. The presented approach supports hiding annotations when the video is displaying and annotations can also be managed individually using an annotation list in a hierarchy.

Fig.3 shows the sketch interface supporting video operation. Users could draw some simple strokes on the screen to navigate videos such as speed-up, slow-down, pause and other actions. The white line on the bottom is the traditional timeline in an arc form, which represents the length of the video. Users could click on the white line for any time he/she wants to take a look. The timeline will disappear when the video is playing and no any other action is taken for a while. Users could use a finger to touch the bottom of the screen to re-activate the timeline.

### 3.2 Collaborative Interaction Architecture

In order to support the collaboration using the sketch interface, we propose a collaborative interaction

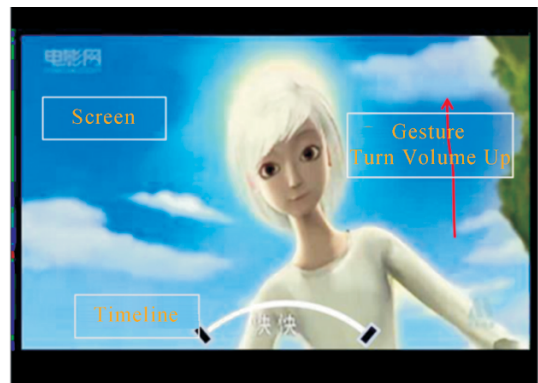


Fig.3. Sample of the sketch interface.

model (CIM), which is based on the C/S model (Fig.4). The CIM consists of the server and clients. From the clients, users draw and write down some sketches as the input, and the client applications convert the input information into commands after being recognized. The server obtains the commands and broadcasts them to proper clients. At the same time, the commands are stored into the database. Finally the clients display the information on the screen after the command is parsed.

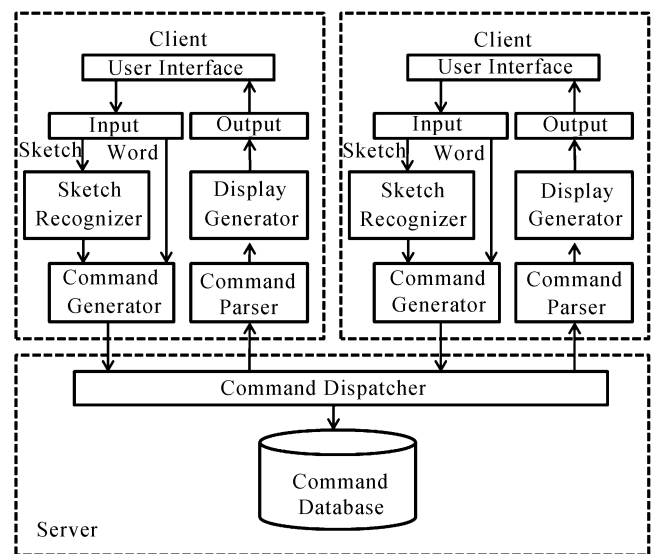


Fig.4. Collaborative interaction architecture.

In our approach, the user-input could be sketches and words. The user could press “Enter” on the keyboard and input the words for sharing in collaboration. Or he/she could draw some strokes and finally draw a “√” to deliver the strokes or draw a “×” to delete them. Some gestures are given in Table1.

The client should generate the command before accessing the server and parse the command after receiving the data from the server. We use XML as the format of the command. Fig.5 shows some examples of the command files.

```
<?xml version = "1.0" encoding = "utf-8" ? >
<Command>
  <CreateInfo>
    <UserID>123</UserID>
    <UserNickName>ZSMJ</UserNickName>
    <CreateTime>2012-12-12 20:31</CreatTime>
    <CommandType>WORD</CommandType>
  </CreateInfo>
  <VideoInfo>
    <VideoID>1013</VideoID>
    <VideoName>TITANIC</VideoName>
    <VideoTime>00:37:32</VideoTime>
    <Duration>10</Duration>
  </VideoInfo>
  <WordList>
    <WordString>Hellow World!</WordString>
  </WordList>
</Command>
```

(a)

```
<?xml version = "1.0" encoding = "utf-8" ? >
<Command>
  <CreateInfo>
    ...
    <CommandType>SKETCH</CommandType>
  </CreateInfo>
  <VideoInfo>
    ...
  </VideoInfo>
  <PointList>
    <Bound>
      <StartPoint>10.12, 12.76</StartPoint>
      <EndPoint>13.17, 8.52</EndPoint>
    </Bound>
    <Mid>
      <Point>11.01, 11.54</Point>
      <Point>11.94, 10.32</Point>
    </Mid>
  </PointList>
</Command>
```

(b)

Fig.5. Command in XML format. (a) Command for word. (b) Command for sketch.

The command XML file consists of three nodes: create information, video information, and word list/point

list. The command type label has two different values, “WORD” and “SKETCH”. “WORD” means that the user-input is a word string and “SKETCH” means that the user-input is the strokes on the screen. Label “VideoTime” shows the time of the video when the user finishes the command. Label “Duration” shows the time of the command information displaying on the screen. The last node word list and point list show us the main content of the information.

After the client receives the command with XML format, it would parse the XML document into specific data structure to store the information from the command. Then, it presents all the information on the screen until 10 seconds later according to the “Duration” label as shown in Fig.5.

If the command type is “WORD”, the word would appear on the screen immediately. If the command type is “SKETCH”, there would be a small information window to show the frame with the strokes. The user could click the window and skip to the specific frame with the strokes on it.

### 3.3 Gesture-Based Navigation System

Considering the functional elements on the interface of the video player occupying more than 20% of the screen and the limited space on mobile devices, we introduce a gesture-based video navigation system for users. Users could draw freehand strokes as gesture commands to interact with videos instead of clicking buttons of these functional elements.

#### 3.3.1 Gesture Modeling and Recognition

Gestures are collections of points and strokes in our system. Gestures could be divided into two types: single touch gestures and multi-touch gestures. The single touch gesture is usually represented with one stroke and easy to be recognized but it is hard to represent donzes of functions meeting users’ requirement of interacting with videos. Here, a stroke is any continuous movement of a finger (from finger down to finger up). The multi-touch gesture consists of more than one stroke and constraint among them. The combination of more than one single touch gesture with time or spatial constraints could be a specific multi-touch gesture.

A gesture model is given to minimize the data size, which compresses hundreds of points into the characters<sup>[29]</sup>. Here, it is based on the radian vector table. The radian vector table consists of eight directions: left, left-down, down, right-down, right, right-up, up, left-up, and the eight directions are represented by eight characters, “0”~“7”. So we could record the directions instead of points after a user draws a stroke. For example, after a user draws a circle in a clockwise

direction, we could get the path on the screen as shown in Fig.6. The red arrows on the circle represent the directions of the strokes. The black rectangles show the characters of the arrows. In other words, we could use character string “01234567” to express the circle.

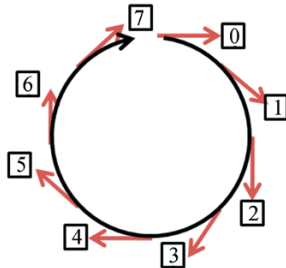


Fig.6. Circle in a clockwise direction.

The model of multi-touch gesture consists of several character strings and constraints. The constraints mainly include separation, gather and cross. For example, after the user completes two strokes, the system has to recognize the two single gestures first and analyze the constraints between them.

3.3.2 Interaction with Gestures

Table 1 shows several typical gestures for navigating videos. Fig.7 shows the user uses a multi-touch gesture “Line Gather” to get the playlist.

Table 1. Typical Gestures for Navigating Videos

Gesture Types	Gesture Names	Description	Function
Single touch	Line Left	←	Fast reverse
	Line Right	→	Fast play
	Line UP	↑	Turn volume down
	Line Down	↓	Turn volume up
	Click	•	Play/pause
	Double Click	•(2)	Full screen/Cancel full screen
	Tick	✓	Select the strokes
	Cross	×	Delete the strokes
	Circle	○	Close
	Multi touch	Line Gather	→←
Line Separate		←→	Return to original video



Fig.7. Sample of Gesture operation. (a) Gesture “Line Gather”. (b) Playlist.

In order to improve the gesture recognition, in our system, we restrict that a gesture should be completed without interruption. If not, it may be divided into several gestures. We set the interval between two different gestures to be at least 1 second. For example if the user wants to get the playlist, he/she should draw a left line and a right line for gesture “Line Gather” at the same time. If he/she draws the right line a second later after he/she finishes the left line, the system would take it as two different single touch gestures. If the user begins drawing the right line in a second after the first stroke, the system would recognize it as that the user wants to send a message to others. The rule of communicating will be described in next section.

4 Implementation

We design and implement a collaborative system to share the creation and communication of sketches’ and words’ annotations in video content. Considering the mobility that is quite different from traditional video operations on desktops based on WIMP paradigm, our presented system provides the sketch interface to manipulate the videos supporting collaborative interaction.

Fig.8 presents the proposed system architecture for implementing the interface, which mainly includes a gesture-based navigation module and a collaborative annotating module. The gesture-based navigation module concerns gesture modeling and recognition. The collaborative annotating module mainly concerns communication among users. We use the C/S structure to model the process of communication among users, and use XML document as the format of command.

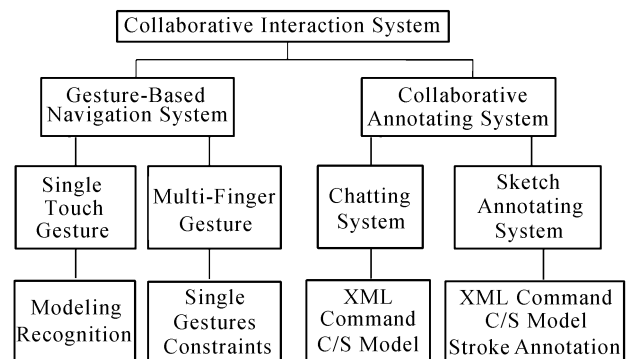


Fig.8. Structure chart.

The system is implemented in C# and is tested with a tablet PC (shown in Fig.9). It provides users an efficient manner to navigate videos with gestures, chat online and share the interesting regions by sketch annotations while watching videos. It also improves user experience without redundant buttons.

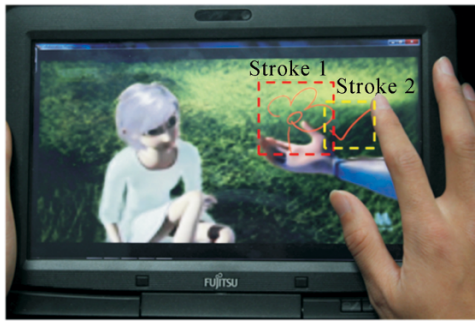


Fig.9. Sketch strokes.

After executing an operation, the system would get the track points left by the user. The track points are the source data of our system. The system transforms the points into a gesture model or command file. In our system we use XMPP for message delivery between clients and the server. XMPP is a kind of instant messaging based on XML. So the command could be nested in XMPP as a message. The clients could get the command while receiving messages from the server.

The sample given in Fig.10(a) shows that Rose and Jack are chatting while they are watching a movie. Users could type some words in the text on the bottom of the player. Fig.10(b) shows Rose draws a stroke on a frame and shares it with Jack. This process would be explained as below. First, the sketches would be sent to Jack and displayed on the lower right corner while Rose finishes several strokes on her screen. Therefore Jack can click the thumbnail to skip to the special frame that Rose wants to share.

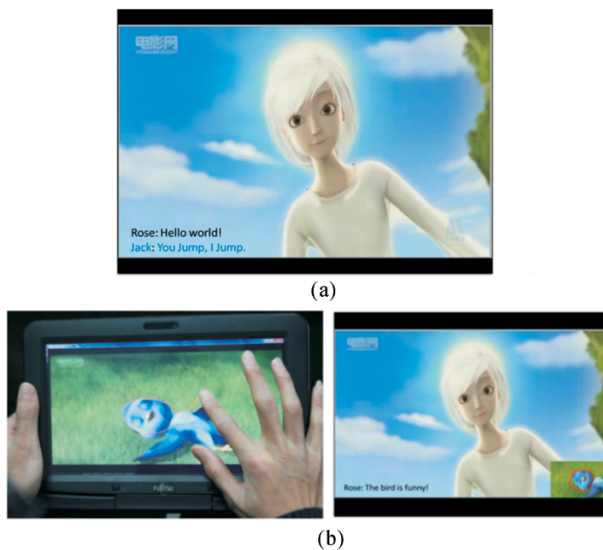


Fig.10. Collaborative UI. (a) Communicating by words. (b) Communicating by sketches.

Furthermore, user can annotate the regions they are interested in using strokes on the screen and send these

strokes to others for communication. In Fig.9, there are two types of strokes on the screen. Stroke 1 in red rectangle shows the region of interest and stroke 2 in yellow rectangle means that the strokes are completed. The interval of the two strokes must be within a second. If they are separated by more than 1 second, the two strokes would be recognized as two different gestures.

### 5 User Study

Firstly, a preliminary requirement analysis is done based on interviewing with some users before we implement the proposed interface. We invite five participants who all are master students and familiar with mobile devices. We introduce the scenario of collaborative interacting with videos on mobile devices and ask them about the operation requirements. Most of them point out that 1) the interface should be intuitive easy and simple to handle, 2) sharing the ideas about a video is common and interesting, which needs an effective and efficient tool.

Secondly, to test the usability and gain feedback about the functionality of the presented annotating and navigating modules using the sketch interface, we compare our system with the popular Mukio Player, which supports word annotation.

Six participants from a university are invited, including 3 females and 3 males, aging from 20 to 24. Two video clips are provided to them, whose lengths range from 20 to 130 seconds. Participants are asked to annotate these video clips at appointed scenes using typed keywords and sketches respectively.

Fig.11 shows the cost of time. Repeated measures ANOVA are conducted and the results show that the main effect of different methods is significant,  $F(1, 5) =$

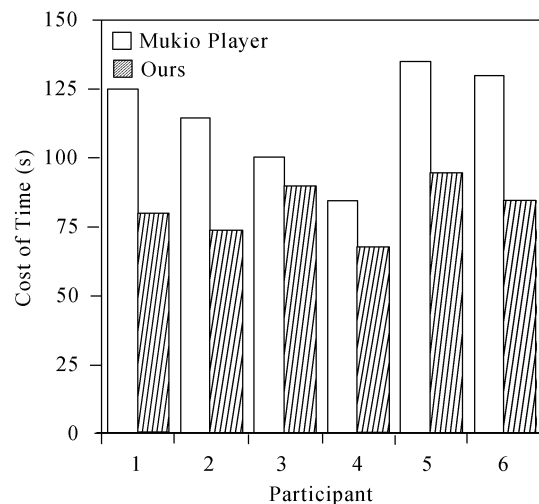


Fig.11. Implementation time of annotating using the two methods.



22.9,  $p < 0.01$ . There is a significant difference in the spent time of completing tasks between using our method ( $M = 85$ ,  $SD = 11.8$ ) and the traditional timeline slider ( $M = 117$ ,  $SD = 20$ ).

After finishing the annotations, participants are asked to evaluate which type of annotations characterizes the clips well and which type of interactions improves user experience well, by rating with “excellent”, “good”, “fair”, “poor” and “bad”. We use scores from 5 to 1. We collect the participants’ evaluation and average the scores over two clips. Paired-sample  $t$ -test is performed to measure the feedback. For sketch and word annotation,  $t(5) = -3.70$ ,  $p < 0.01$ , which shows that users prefer to sketch annotations. For user experience,  $t(5) = -5.21$ ,  $p < 0.01$ , which shows that users prefer to the sketch interface.

Participants prefer to the gesture interaction of our method instead of traditional mouse clicking operations. Most of participants value high for annotating video using freehand drawings in the collaborative environment. Participants also give some feedback for improving the interface, such as to improve the gesture’s recognition rate, add more strokes styles and save command files for reviewing.

## 6 Conclusions and Future Work

Video sharing service is becoming more and more popular with the development of Internet. In this paper we proposed an approach supporting collaborative annotating and navigating system for video sharing. It adopts a collaborative interaction model, which forms commands between clients and the server as XML files to implement collaborative operations. Gestures are given for navigating videos and a radian vector table is used for gesture modeling and recognition. A gesture-based navigation system has been implemented based on the approach and model. Compared with the previous work, the proposed system improves the communication efficiency in annotating and navigating videos.

However, there are some limitations in our system. For example, we cannot support too many gestures in our current system because of the low recognition rate. In our future work, we would focus on gesture description and efficient recognition algorithms to improve the sketching interface. Besides, user cognition in the collaborative environment would be considered in our interface.

## References

- [1] Wang S D, Higgins M. Limitations of mobile phone learning. In *Proc. IEEE International Workshop on Wireless and Mobile Technologies in Education*, Nov. 2005, pp.179-181.
- [2] Guimarães R L, Cesar P, Bulterman D. Creating and sharing personalized time-based annotations of videos on the web. In *Proc. the 10th ACM Symposium on Document Engineering*, 2010, pp.27-36.
- [3] Moxley E, Mei T, Manjunath B S. Video annotation through search and graph reinforcement Mining. *IEEE Trans. Multimedia*, 2010, 12(3): 184-193.
- [4] Nurminen J K, Karonen O, Farkas L, Partala T. Sharing the experience with mobile video: A student community trial. In *Proc. the 6th IEEE Conference on Consumer Communications and Networking Conference*, Jan. 2009, pp.1197-1201.
- [5] Dezfuli N, Huber J, Olberding S, Mühlhäuser M. CoStream: In-situ co-construction of shared experiences through mobile video sharing during live events. In *Extended Abstracts of Conference on Human Factors in Computing Systems*, May 2012, pp.2477-2482.
- [6] Hürst W, Götz G, Welte M. Interactive video browsing on mobile devices. In *Proc. the 15th International Conference on Multimedia*, Sept. 2007, pp.247-256.
- [7] Lee S A, Jayant N. Gestures for mixed-initiative news video browsing on mobile devices. In *Proc. the 17th ACM International Conference on Multimedia*, Oct. 2009, pp.1011-1012.
- [8] Goldman D B, Gonterman C, Curless B, Salesin D, Seitz S M. Video object annotation, navigation, and composition. In *Proc. the 21st ACM Symposium on User Interface Software and Technology*, October 2008, pp.3-12.
- [9] Hauptmann A G. Lessons for the future from a decade of informedia video analysis research. In *Proc. the 4th ACM Int. Conf. Image and Video Retrieval*, July 2005, pp.1-10.
- [10] Wang M, Hua X S, Tang J H, Hong R. Beyond distance measurement: Constructing neighborhood similarity for video annotation. *IEEE Trans. Multimedia*, 2009, 11(3): 465-476.
- [11] Yan R, Naphade M R. Semi-supervised cross feature learning for semantic concept detection in videos. In *Proc. Computer Vision and Pattern Recognition*, June 2005, pp.657-663.
- [12] Eitz M, Hays J, Alexa M. How do humans sketch objects? *ACM Transactions on Graphics*, 2012, 31(4): Article No. 44.
- [13] Eitz M, Richter R, Boubekeur T, Hildebrand K, Alexa M. Sketch-based shape retrieval. *ACM Transactions on Graphics*, 2012, 31(4): Article No. 31.
- [14] Chao M, Lin C, Assa J, Lee T. Human motion retrieval from hand-drawn sketch. *IEEE Transactions on Visualization and Computer Graphics*, 2012, 18(5): 729-740.
- [15] Sheng Y, Yapo T C, Young C, Cutler B. A spatially augmented reality sketching interface for architectural day lighting design. *IEEE Transactions on Visualization and Computer Graphics*, 2011, 17(1): 38-50.
- [16] Lin J C, Igarashi T, Mitani J, Liao M H, He Y. A sketching interface for sitting pose design in the virtual environment. *IEEE Transactions on Visualization and Computer Graphics*, 2012, 18(11): 1979-1991.
- [17] Zhu X Q, Jin X G, Liu S J, Zhao H L. Analytical solutions for sketch-based convolution surface modeling on the GPU. *The Visual Computer*, 2012, 28(11): 1115-1125.
- [18] Xu K, Li Y, Ju T, Hu S M, Liu T Q. Efficient affinity-based edit propagation using K-D Tree. *ACM Transactions on Graphics*, 2009, 28(5): Article No. 118.
- [19] Xu K, Wang J P, Tong X, Hu S M, Guo B N. Edit propagation on bidirectional texture functions. *Computer Graphics Forum*, 2009, 28(7): 1871-1877.
- [20] Xu K, Chen K, Fu H B, Sun W L, Hu S M. Sketch2Scene: Sketch-based co-retrieval and co-placement of 3D models. *ACM Transactions on Graphics*, 2013, 32(4): Article No. 123.
- [21] Liu Y J, Tang K, Joneja A. Sketch-based free-form shape modelling with a fast and stable numerical engine. *Computers & Graphics*, 2005, 29(5): 771-786
- [22] Ma C X, Liu Y J, Wang H A, Teng D X, Dai G Z. Sketch-based annotation and visualization in video authoring. *IEEE*

*Transactions on Multimedia*, 2012, 14(4): 1153-1165.

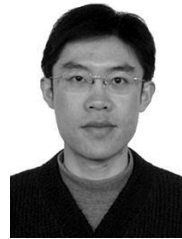
- [23] Davis R, Landay J, Chen V, Huang J, Lee R, Li F, Lin J, Morry C B, Schleimer B, Price M, Schilit B. NotePals: Lightweight note sharing by the group, for the group. In *Proc. SIGCHI Conference on Human Factors in Computing Systems*, May 1999, pp.338-345.
- [24] Beth S, Anderson R, Wolfman S. Activating computer architecture with classroom presenter. In *Proc. Workshop on Computer Architecture Education*, June 2003, Article No. 11.
- [25] Xia S, Sun D, Sun C Z, Chen D, Shen H F. Leveraging single-user applications for multi-user collaboration: The CoWord approach. In *Proc. ACM Conference on Computer Supported Cooperative Work*, Nov. 2004, pp.162-171.
- [26] Liu Y J, Luo X, Joneja A, Ma C X, Fu X L, Song D W. User-adaptive sketch-based 3D CAD model retrieval. *IEEE Transactions on Automation Science and Engineering*, 2013, 10(3): 783-795.
- [27] Wang H A, Ma C X. Interactive multi-scale structures for summarizing video content. *Science China Information Sciences*, 2013, 56(5): 1-12.
- [28] Furnas G, Bederson B. Space-scale diagrams: Understanding multiscale interfaces. In *Proc. ACM Conference on Human Factors in Computing System*, May 1995, pp.234-241.
- [29] Liu Y J, Chen Z Q, Tang K. Construction of iso-contours, bisectors and Voronoi diagrams on triangulated surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011, 33(8): 1502-1517.



**Jin-Kai Zhang** received his B.S. degree from the Department of Computer Science and Technology, Tsinghua University, Beijing, in 2012. He is currently a Master's student in the Department of Computer Science and Technology at Tsinghua University. His research interests include computer graphics and human-computer interaction.



**Cui-Xia Ma** received her Ph.D. degree in computer science from Institute of Software), Chinese Academy of Sciences (CAS), Beijing, in 2003. Currently, she is an associate professor with the Institute of Software, CAS. Her research interests include human computer interaction and multimedia computing. Dr. Ma is a member of IEEE.



**Yong-Jin Liu** received the Ph.D. degree in mechanical engineering 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, China. He is a member of IEEE, a member of IEEE Computer Society and IEEE Communications Society. His research interests include pattern analysis, human-computer interaction, computer graphics and computer-aided design.



**Qiu-Fang Fu** received her Ph.D. degree in cognitive psychology from Institute of Psychology, Chinese Academy of Sciences, Beijing, in 2006. Currently, she is a senior researcher in cognitive psychology. Her research interests include implicit learning, unconscious knowledge, category learning, and subliminal perception. At present, she is a member of Chinese Psychological Society and a fellow of the Association of Scientific Study of Consciousness.



**Xiao-Lan Fu** received her Ph.D. degree in cognitive psychology from Institute of Psychology, Chinese Academy of Sciences, Beijing, in 1990. Currently, she is a senior researcher in cognitive psychology. Her research interests include visual and computational cognition: attention and perception, learning and memory, and affective computing. At present, she is the director of Institute of Psychology, Chinese Academy of Sciences, and the vice director of the State Key Laboratory of Brain and Cognitive Science of China, Beijing.