A Streaming Video Modification Method for Improving Visibility of the Content in the Video

Katsuhiko Kaji
Graduate School of Engineering,
Nagoya University
kaji@nuee.nagoya-u.ac.jp

Nobuo Kawaguchi
Graduate School of Engineering,
Nagoya University
kawaguti@nagoya-u.jp

Abstract

Recently, demand for video presentation has been on the increase. Various conferences adopt video streaming service for sharing the scene of presentation. In such cases, generally, the video camera captures both the presenter and the presentation slide projected on a screen. However, sometimes, presentation slides in the streaming video are not so visible in the client side. This is due to various reasons such as low resolution, distortion, and angle of view. Users cannot understand the presentation clearly by watching such video. In this paper, we propose a streaming video modification method to improve the visibility of the content in the streaming video. In this method, the client downloads the original data of the content in the streaming video previously then in the watching phase, the image generated from original data is superimposed at the position of the content area in the streaming video.

1. Introduction

Recently, various video streaming services such as YouTube [17] are available. Users are able to watch various types of videos on demand. In particular, the demand of presentation video is high so that many presentation videos are submitted to the video streaming services [13]. Additionally, because of improved broadband environments, live video streaming services such as Ustream [14] and Niko Niko Live [9] are also available. Several conferences and workshops adopt the service to enable the presentations to be availed across the world in real-time.

The videos shared via the above mentioned services tend to contain "core content." In the case of presentation in a conference, a presenter shows some topics using presentation slides (e.g. Microsoft PowerPoint) projected on a screen. When a provider intends to share the scene of presentation, both the presenter and the screen are captured by a video camera and the image is shared via video streaming service. Generally, the presenter believes that the audience can see the presentation slide clearly. Therefore, the audience cannot understand when the visibility of the presentation slide is low. In this situation, we call the presentation slide projected on the screen as "core content."

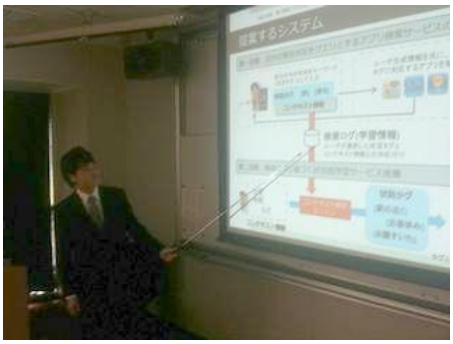

Figure 1: An example of low visibility of the slide in a streaming video.

Unfortunately, when a user watches streaming video, the visibility of core content in the video is sometimes low (Fig. 1). Actually, the audience cannot understand what is described in the slide clearly. The main reasons are as follows.

- Low resolution of the core content.
- Distortion of the core content because the video camera isn't placed in front of the content.
- Video image doesn't contain the whole of the core content.

It is too difficult to solve these problems simultaneously in traditional streaming service. On the server side, core content should be captured clearly. For realization of high visibility, high resolution camera should be prepared. The camera position should be right in front of the content and the angle of view should be adjusted to capture the whole of the content. Even if the core content is captured clearly on the server side, the problem still remains. On the client side, when the user cannot prepare broadband environment well enough to receive large video streaming data, the video is received as low resolution.

There are several web services to share various types of contents such as images [1], video [10, 13, 17], and presentation slides [11]. Therefore if the original data of the core content included in the streaming video is shared in such services, users can download the data before watching the video.

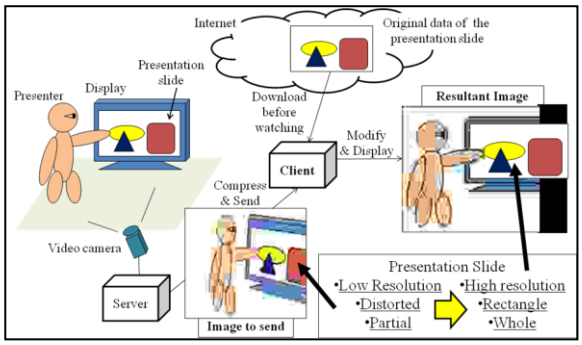

Figure 2: An overview of proposal method.

In this paper, we propose a streaming video modification method for improving visibility of core content. The proposed method is overlaying the image made from original data of the core content onto the position of the content in the streaming video. Therefore the proposed method is based on the premise that the original data of the core content in the video is able to be acquired via web.

The paper is organized as follows. In section 2, we describe the related work. Then, in section 3, we propose a streaming video modification method for improving visibility of the content in the video. In section 4, we describe an implementation of the method. The system is live streaming of a poster presentation. Conclusion and future work is described in section 5.

2. Related Work

It is said that consistency of physical relationship of the users and objects is important for remote communication using video [2, 3, 5]. If physical relationships are inconsistent, a remote user cannot understand the local user's gesture, sight line, and direction of the body viscerally [2]. Traditional presentation video streaming using Ustream and Niko Niko Live is as follows. A video camera mainly captures the presenter and the slide being projected on a screen. These images are combined on server side, and the presenter's image is superimposed onto the slide image's corner. During the question time, the main image is changed to display the presenter on a large scale. In such video streaming, physical relationship between the presenter and the slide is not consistent.

While projecting desktop image of PC to a screen, it is possible to capture and send the digital image of desktop directly. Actually, several remote collaboration systems send local PC's desktop image to the remote site [5, 6]. By using the technique, desktop image is able to be sent to the remote site with high visibility. However, when the desktop resolution is high, it is known that capture speed of desktop image cannot be fast enough for several types of contents with high frame rate such as video contents.

Omnisio [10] is a web service for presentation video that makes it possible to seek the video by selecting the slide thumbnails listed below the video. When user watches a presentation video from GoogleVideo [4] or YouTube [17], the user also selects the original data of the slide in the presentation video from SlideShare [11]. Though the system uses the original data of slide, it does not improve the visibility of the slide in the video because the slide images are listed as small thumbnail.

3. Proposed Method

Proposed method is to combine streaming video image and the original data for improving the visibility of the core content in the video. Therefore both the server and the client need to download the original data via web previously. An overview of proposed method is shown in Fig. 2. The method requires a constraint that the original data of the core content in the video be shared on the web. As shown in Fig. 2 below, the image to be sent tends to contain the following problems. Resolution of the core content is low, the area of core content is distorted and the image doesn't contain whole of the core content. In the proposed method, the image made from original data of the core content is overlaid onto the position of the core content in the streaming video. At the same time, objects existing in front of the core content are not disappeared by image processing. Additionally, the image is converted to include whole of the core content and the distortion of the core content is corrected as rectangle (Fig.2 right).

3.1. Processing on the Server

On the server side, the system creates several metadata for image processing, and sends the processed images and metadata accordingly.

3.1.1. Content identification. In this process, the server acquires the original data's address in the video. There are several ways to acquire the address such as manual input by the provider and detection of the 2D code pasted near the poster. The server needs the original data of core content for image processing. The system downloads the data before starting the streaming service.

3.1.2. Detection of content position. In order to combine the original data and received image, information about the core content's position is necessary (Fig.3 right). When the core content is quadrangle like a presentation slide and a poster, coordinates of each corner can be taken as content position. For example, coordinates of corners are able to be detected by the content's edge detection, extraction of markers placed at each corner, template matching between captured image and original data. When the whole of the core content is not captured,

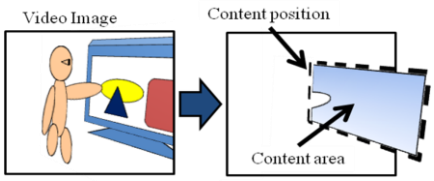

Figure 3: Extraction of content position and area from video image

corners outside the scope must be detected by using partial template matching or information of core content's aspect ratio.

3.1.2. Extraction of content area. Figures of user's finger or objects in front of the core content must not disappear during image processing in the proposed method. Therefore the server extracts the content area that doesn't include the object's area in front of the content (Fig.3 right). This process is necessary each frame.

Content area can be extracted by following three techniques. One is optical technique. In this technique, polarizing filter is pasted on the face of core content and the lens of video camera then only the objects in front of core content are captured and content area turns black [5, 12]. Therefore black region in the content position can be taken as content area.

Next technique is image processing based on background subtraction. Before starting streaming service, server saves image as background image with no objects in front of the content. After starting streaming service, content area is extracted by subtraction of background image from captured image. Then, only the objects in front of core content appear in the subtracted image, and content area turns black. Then, the black region can be taken as content area. Several robust methods of background subtraction are proposed for illumination/motion changes [7, 8].

Third technique is to use computer screen image directly from the computer and subtract the image on the screen from camera image [16].

There are two ways to inform a client of a content area. One is using vector information such as SVG (Scalable Vector Graphics) [15], and the server sends it as metadata. The other is sending an image in which the content area is filled as black. After a client receives the image, the client extracts the black region in the content position and takes the region as content area. The latter method needs an additional process for content area extraction on the client side. The latter method can decrease the amount of streaming data because the content area is filled as black. It can be said that the latter method is effective for situations where the ratio of the core content's area in video image is large.

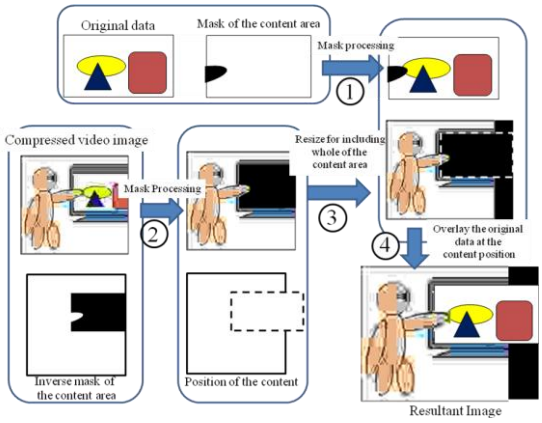

Figure 4: Combination process of video image and original data.

3.1.3. Detection of content state. Several types of core content such as presentation slide and video content vary with time and user operation. Therefore the server needs to detect current state of the content. Examples of states are the current slide to display and the current position of the seek bar of video content. There are mainly two techniques to detect the state. One is using the operation history of the PC, and the other is image processing based on template matching.

3.1.4. Correction of distortion. When a video camera is not correctly placed in front of the core content, the figure of the content in the image tends to be distorted. To correct the distortion, the server process affine transformation by using aspect ratio and position of the core content. During processing, the information of the content's position and area is transformed correspondingly.

3.1.5. Streaming. Even through the processing is on the server side, the information of content URI, position, area, and state is acquired. The server sends the information to the clients accordingly. In the situation that content position/state varies with time, the information should be detected dynamically and sent to the clients.

3.2. Processing on the Client

For preparation, the client first receives the metadata from the server, and downloads the original data of the core content via web. On the other hand, we can also use frame buffer streaming protocol such as RDP (Remote Desktop Protocol) to get core content from server side, when the presenter allows installing special software for capturing and streaming the desktop image. It is difficult to apply the method when the core content cannot be downloaded previously. For example, when the presenter demonstrates his own system interactively at the server site, it can be assume that the core content is the behavior

of the system shown in the screen. At the time, if the presenter doesn't allow installing desktop capture and streaming software, the client cannot deal with the core content previously.

After downloading the original data, the client starts receiving images and metadata. Procedure of image processing using those data is shown in Fig. 4. First, by using mask image from content area information, the original image data is generated (Fig. 4-1). The region where some objects exist in front of the core content is filled as black in the image. Next, by using a received image and an inverse mask image of content area, an image in which the content area is filled as black is generated (Fig. 4-2). Then by using position information of the core content, the image generated in the second process is resized to contain whole of the content region (Fig. 4-3). Lastly, the image generated in first process is overlaid onto the core content's position in the image generated in third process (Fig.4-4). These processes should be done according to the image size of original data, received image, and resultant image.

4. Poster Presentation Video Streaming System

In this section, we describe a live streaming system for poster presentation based on the proposed method. Poster presentation is one of the most effective situations for the method, because figures and texts in a poster are relatively smaller than the presentation slide for a general conference. Therefore, users cannot understand what is described in the poster by watching streaming video. Additionally, the presenter tends to be close to the poster and use various gestures like pointing at the poster.

4.1. Implementation

This system is implemented using VC++ as the programming language, Qt as network and GUI platform, and OpenCV for image processing. Fig. 5 shows snapshots of a image captured for a poster presentation, the image to be sent, and the resultant image. You can find that the visibility of the poster content in the resultant image is higher than the captured image.

The Proposed method mentioned in section 3 is implemented as follows. Video camera captures only one poster, and the whole of the poster is in the angle of view. The address of the poster's original data is input by a provider at the sender side manually. For easy detection of poster position, red markers are located on the poster's four corners. Poster position is detected by extraction of red color in captured image. Detection of content area is realized by image processing. Before starting streaming service, the system saves 50 frames of captured image with no objects in front of the poster. Background subtraction is calculated by using the average and

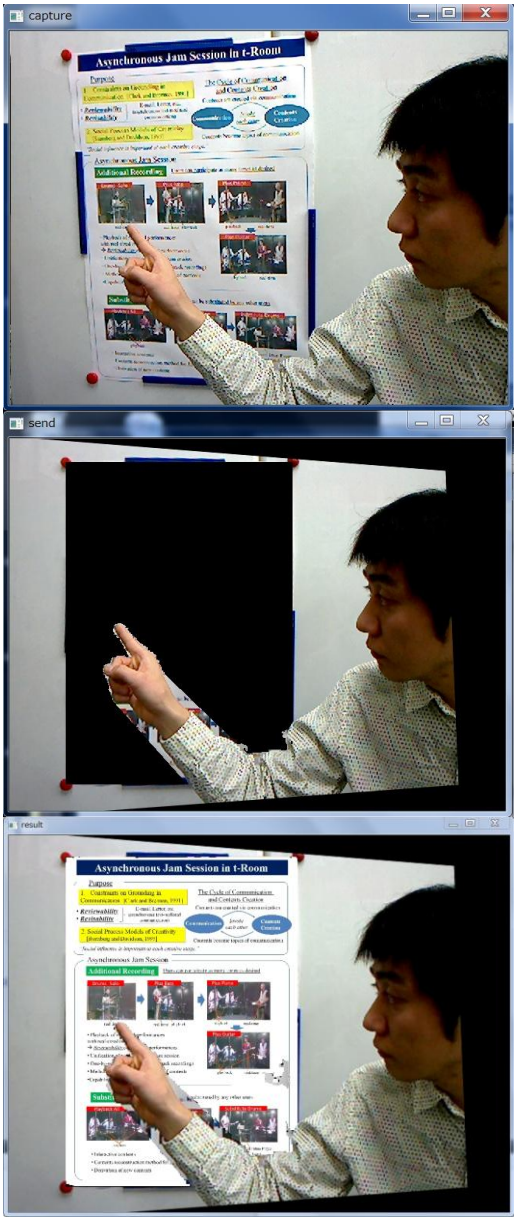

Figure 5: An example of capture image, send image, and resultant image.

variance of the saved images. We don't adopt polarization film technique to avoid disappointing the people in the presentation room due to poor visibility.

In the system, the ratio of poster's area in captured image tends to be high. Therefore, to reduce the amount of streaming data, we adopt the technique that fills the poster area as black. The client detects the poster area by extracting black region (section 3.2) (Fig. 5 center).

Distortion correction is processed by using original data's aspect ratio and the length between the markers of upper-left and lower-left in captured image. Affine transformation is processed for the poster's area to be square and correct aspect ratio (Fig. 5 center). In poster

Figure 6: Client interfaces. Navigation window (left), and main window (right).

presentation, content information doesn't change dynamically. Therefore, the system sends metadata once at the timing of starting connection to client. After sending the metadata, the server only send the image (Fig.5 center) as Motion JPEG. Compression ratio of JPEG and frame rate can be changed dynamically by the user interface of server side.

Currently, the system doesn't handle the situation that whole poster image is not completely captured. By using red markers to be captured and partial template matching between the image and the original data, the system can handle such situation. General poster presentation uses one static poster for one presentation, so that the system doesn't need content state detection mentioned in section 3.1.3 for poster presentation.

To adapt the situation of poster presentation, we also implemented zooming function and finger tracking function. Because texts and figures in a poster tend to be small, the user at the client side can read the detail content only by using enough large display. Zooming function solve the problem. User can change zoom level by slide bar above the navigation window shown at Fig. 6 left. Additionally, position to zoom can be changed by clicking the navigation window. The zoomed image is shown in main window (Fig. 6 right).

In the case of poster presentation, presenter frequently points at the poster. Finger tracking function allows users to look at the region pointed by the presenter easily. In the system, we use flesh color extraction as finger detection. Position to zoom is automatically changed by using center coordinate of extracted flesh color region in the poster area. Finger position is shown as a green rectangle (Fig. 6 left). Finger detection is processed on the client side. Position to zoom changed when the finger stays definite area for a given length of time, so that position of zoom doesn't move when the finger moves slightly, and when the finger disappears in front of the poster. Automatic finger tracking function is switchable, so that the user can also focus elective position manually.

We are also considering several design variations. The proposal method can be applied for not only streaming video but also static video. Besides, finger image can be overlaid as semi-transparent for visibility of the part of the core content under the finger image.

4.2. Experiment

For evaluation of performance of the system, we tested about image processing time and the amount of streaming data. Evaluation environment was as follows. Presentation poster was created using Microsoft PowerPoint, and printed out as A1 size. Original data of the poster is saved as PNG format and the size was same as the printed poster (A1: 2170x3072). We used Logicool QCam Ultra Vision as video camera and set the capture size as 640x480. The camera was placed approximately 1.5m in front of the printed poster. The whole of the poster and upper body of the presenter was included in the capture image. In the setting, proportion of poster area is about 34%. JPEG quality was set as 80%, and frame rate was 20fps.

A server and a client program was operated in one note PC (OS: Windows7 64bit, CPU: Core i7 2.8GHz, Memory: 8GB), and we used 30 frames of streaming for test data. The first test was about processing time. The average processing time on the server side was 41ms and that of client side was 22ms. Consequently, it should be assumed that the processing time of the proposed method is 63ms which is the summation of both the client and server side processing. We assume that 63ms is too small and that such a delay cannot be a problem to most general live streaming situations. The next test is about the amount of data. In the setting, the average of JPEG data size of captured image was 51Kbytes. On the other hand, the average of JPEG data size of the poster area that was filled as black was about 27Kbytes. Consequently, the proposed system succeeded to reduce the data amount to the half level. Of course, reduction ratio of data amount must be changed according to the area ratio of core content.

To evaluate the effectiveness of poster presentation recognition, we conducted subject experiment. A presenter showed certain research theme by using the printed poster on the server side. We divided subjects into two groups namely the proposed system group and the traditional system group. Subjects in the proposed system group could watch the poster presentation live streaming using full functionality of the proposed system. On the other hand, the server and client do not process the images in the traditional system group. Subjects in traditional system group could use only the zooming function. The resolution of note PC was as 1280x800, and main window size for display poster presentation video was as 960x720.

The subjects then changed groups such that each participant experienced both groups by using different poster. The order of group was counterbalanced. After the

experiment, subjects responded to a questionnaire with regards to.

1. Degree of poster presentation recognition
2. Readability of the content printed in the poster
3. Frustration for watching the contents
4. Effort of poster presentation recognition
5. Usefulness of finger tracking function (only proposal system)

Figure 7 shows the result. From question 1 and 2, we should say that the proposed system is effective for presentation recognition and visibility of the poster. In particular, we confirmed the significant difference between the groups in readability of the poster by using t-test ($p < 0.01$). However, we couldn't point out the significant difference in degree of recognition. We suppose the reason is that the research themes of poster presentation were too simple. From questionnaire 3 and 4, frustration of watching and effort of recognition can be suppressed to low level by using the proposed system. We found out the significant difference between the groups in both of these two questions ($p < 0.05$). From the result of question 5, most subjects answered that the finger tracking was useful (avg: 4.50, sd: 0.53). Consequently, it can be said that the proposed system improves the visibility of poster content, so that the user feels low frustration and needs low effort to recognize the presentation.

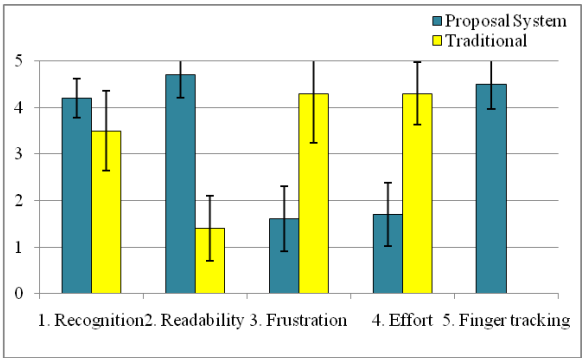

Figure 7: The result of questionnaire.

5. Conclusion

In this paper, we proposed a streaming video modification method for improving visibility of core content in the video. The proposed method overlays the image made from the original data of the core content, which is previously downloaded via web, at the position of the content in the streaming video. We also implemented a poster presentation live streaming system based on the proposed method, and we found the effectiveness of the system through experiments.

Future work is as follows. 1. Installing the poster presentation streaming system to real conference for verifying the effectiveness of the proposed method in the real-world. 2. Implementation of video streaming system for other types of core contents such as presentation slide.

6. References

[1] Flickr: http://www.flickr.com/.

[2] Fussell, S. R., Setlock, L., Yang, J., et al.: Gestures Over Video Streams to Support Remote Collaboration on Physical Tasks, *Journal of Human-Computer Interaction*, Vol.19, pp.273-309 (2004).

[3] Gaver, W., Sellen, A., Heath, C., et al.: One is not Enough: Multiple Views in a Media Spaces, *INTERCHI'93*, pp.335–341 (1993).

[4] GoogleVideos: http://video.google.co.jp/

[5] Hirata, K., Harada, Y., Takada, T., et al.: Video Communication System Supporting Spatial Cues of Mobile Users, *Proc. of CollabTech 2008*, pp.122–127 (2008).

[6] Ishii, H., Naomi, M.: Toward An Open Shared Workspace: Computer and Video Fusion Approach of TeamWorkStation, *Communications of the ACM*, Vol. 34, Issue 12 (1991).

[7] Koller, D., Weber, J., Huang, T., et al.: Towards Robust Automatic Traffic Scene Analysis in Real-Time, *Proc. of International Conference on Pattern Recognition*, pp. 126-131 (1994).

[8] Lo, B.P.L., and Velastin, S.A.: Automatic congestion Detection System For Underground Platforms, *Proc. of 2001 International Symposium on Intelligent Multimedia, Video and Speech Processing*, pp.158–161 (2001).

[9] Niko Niko Live: http://live.nicovideo.jp/

[10] Omnisio: http://www.omnisio.com/

[11] SlideShare: http://www.slideshare.net/

[12] Tang, J. C. and Minneman, S. L.: VideoDraw: A Video Interface for Collaborative Drawing, *Proc. of CHI*, pp.313–320 (1990).

[13] TED: Ideas worth spreading: http://www.ted.com/

[14] Ustream: http://www.ustream.tv/

[15] W3C: Scalable Vector Graphics (SVG), http://www.w3.org/Graphics/SVG/

[16] Xin, Q., Chen, M., Takatsuka, M., Hand Segmentation on a Dynamic Screen using Image Subtraction, *Proc. of IEEE VGTC Pacific Visualization Symposium (PacificVis)*, 2008

[17] YouTube: http://www.youtube.com