

# Virtual Video Frameworks for Generic Video Applications on Internet

**Koji OKAMURA**

Computing and Communications Center,  
Kyushu University  
oka@ec.kyushu-u.ac.jp

**Yasuhiko YOSHIMURA**

Graduate School of Information Science  
and Communication Engineering,  
Kyushu University  
yosimura@dontaku.csce.kyushu-u.ac.jp

**Masatoshi ARIKAWA**

Center for Spatial Information Science  
University of Tokyo  
arikawa@csis.u-tokyo.ac.jp

**Michihiro MURAO**

Center for Spatial Information Science  
University of Tokyo  
mura@csis.u-tokyo.ac.jp

## Abstract

*This paper introduces a new framework, that is, the virtual video (vvideo) capture device [2] on Unix operating system. The vvideo capture device is suitable for realizing augmented reality applications as networked generic applications. It enables the augmented reality applications to become generic applications via the kernel layer so that all ordinary networked software for using videos can incorporate augmented reality applications directly and easily. Also, we discuss the basic mechanism of the announcement method on Internet which enables all sites to reach any augmented reality applications on Internet. Then, we propose a new video information management model for dealing with spatial and temporal characteristics as metadata of video data. Important components of the new model are Fragmented Reality (FR) and Recalled Reality (RR) for storing video data from the real world and for retrieving and processing video data dynamically. Finally, we show some augmented reality prototypes on Internet based on our new frameworks.*

## 1 Introduction

Augmented Reality (AR) is a new important platform for integrating digital information and the real world. AR provides users with more natural and powerful interfaces with considering conditions of the real world and the intention of the users. AR can be imple-

mented by various methods. One of the most general methods for realizing AR is to use videos composed of computer graphics (CG). For example, videos or movies can be composed of pieces of CG objects which can also be used as tags to be linked to related information objects distributed on Internet like a part of the Web. Since the composite videos themselves serve as hypermedia, we call the composite videos *Augmented Video Hypermedia (AVH)* in this paper. In addition that AVH have the capability of integrating various multimedia data such as texts, videos and sounds, AVH can connect the multimedia data with geographic locations. AVH on Internet will offer the opportunities for creating more attractive applications, and many people in the world could benefit from AVH through popular networked applications on Internet.

This paper introduces a new framework, that is, the virtual video (vvideo) capture device [2] on Unix operating system which is suitable for realizing AVH as networked generic applications. The vvideo capture device enables AVH to become generic applications via the kernel layer so that all ordinary networked software for using videos can incorporate AVH directly and easily. Next, we discuss the basic mechanism of the announcement method on Internet which enable all sites to reach any AVH on Internet. Then, we propose a new video information management model for dealing with spatial and temporal characteristics as metadata of video data. Important components of the new model are *Fragmented Reality (FR)* and *Recalled Reality (RR)* for storing video data from the real world and for

retrieving and processing video data dynamically. Also, we show some AVH prototypes on Internet based on our new frameworks.

## 2 Augmented Video Hypermedia

The following is the functions of augmented video hypermedia (AVH):

- Annotation: Positioning Text Characters on Live Videos

The text characters represent the annotations of objects in the live video (Fig. 1). Annotations run after real objects in the video whenever the direction and zooming ratio of a camera are changed. This is the most important function of the application.

- Entry and Deletion of Information in an Augmented Real Space

Spatial databases can be changed through the live video. In order to add a piece of new data to the database, we create a new annotation in the live video. Deletion of annotations means that the data are deleted from the database.

- Control of the Remote Camera

A user can control the direction and zooming ratio of a remote camera, and direct it toward his intended objects by selecting an entry from a visual menu.

- Clickable Augmented Real Space

An annotation is clickable for associating with its corresponding WWW page. This function turns the real world into a kind of hypermedia.

- Levels of Detail (LoD)

In the case of zooming in on some objects using a video camera, we can obtain more details of the object as additional CG data composed of a live video. On the other hand, CG objects are not displayed with a lower zooming ratio. LoD controls the quality of graphical representations of CG objects.

- Protecting Privacy for Remote Live Videos

Ensuring privacy of remote live videos is an important issue on Internet. The technique of placing text characters in a video can be applied to protecting privacy of remote live videos by means of replacing a partial image on the video. For example, a building in a live video is replaced by a low resolution image or its past image.

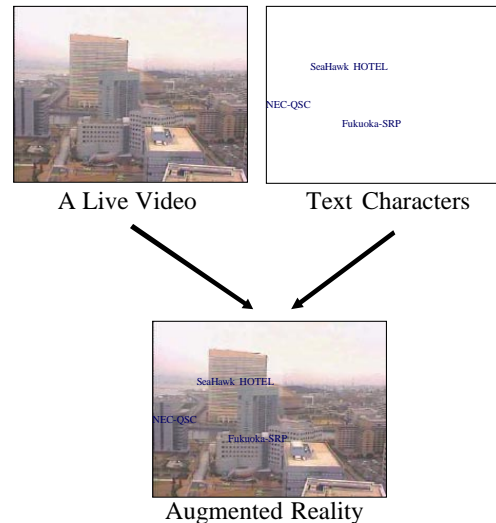


Figure 1. Augmented reality as composition of a live video and text characters.

Fig. 2 shows the basic configuration of spatial hypermedia. The real-world area represented by the video frame can be calculated by the data regarding the condition of the camera, e.g., its direction and zooming ratio. Information about objects in the video is retrieved based on the camera condition data. Finally, a user can obtain augmented reality, which consists of a real video and CG text characters.

We have implemented a prototype system of a spatial hypermedia application *Name-at*[1]. It is derived from the operation that we register the *names* at the objects in the video as if we put some memos on the desk.

## 3 Virtual Video Frameworks to Realize AVH on Internet

We introduce two new frameworks to realize the Augmented Video Hypermedia on Internet. The first one is *virtual video capture device*, shortly called *vvideo*, for virtualizing remote video capture devices as local files. And the next one is a video information management system which handles spatial and temporal characteristics of video data.

### 3.1 Virtual Video Capture Device

For an application program to handle some movies captured at a remote host, the application program

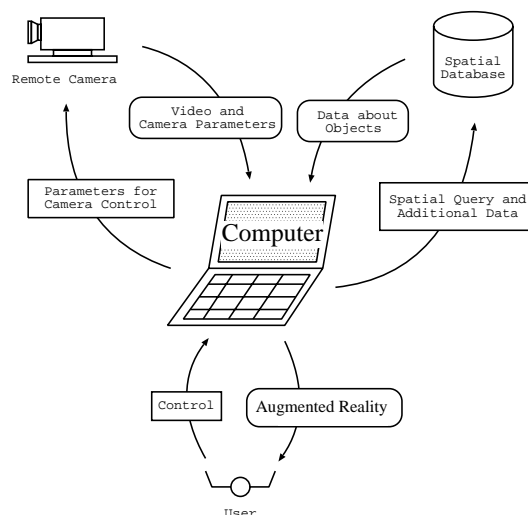


Figure 2. Basic configuration of spatial hypermedia.

should have functions for both encoding and decoding video data as well as transferring the video data over networks. If an application program is designed for treating only local multimedia data and it must be extended to deal with remote multimedia data on Internet, the extension is generally not easy because other functions must be created and added to the program and many parts of the program must be modified. We propose *virtual video (vvideo) capture device*, shortly called *vvideo*, to provide an easy and powerful framework to solve the above case. In other words, vvideo can offer transparent video capture devices on some network environment. Using vvideo, all of remote video capture devices can be used in the same way as local video capture devices. Vvideo makes easy to extend local application programs to remote ones.

Recently, controllable cameras have become popular. The controllable camera provides functions for personal computers to change camera parameters such as pan, tilt and zoom ratios through cables connected with the computers. Although both controlling camera parameters and capturing video data are executed separately at the device driver level, an application program for using a controllable camera must synchronize both camera parameter data and video data. It is difficult to create program codes for realizing the synchronization. Vvideo can solve the synchronization problem because vvideo can provide both camera parameter data and video data from one device driver. Thus, it integrates the control of the camera and video

data from the camera, in addition that it integrates remote video capture devices and local video capture devices. The following is detailed explanation for the two capabilities.

### Transparent video capture device over Internet

Vvideo offers the function of transparent video capture device over Internet, which makes it possible to use remote video capture devices and local video capture devices in the same way. Using vvideo, video application programs access special device files such as `/dev/video0` for local video capturing in order to capture videos from remote hosts. Its concepts is illustrated in Fig. 3. The system calls such as `open()`, `close()`, `ioctl()` and `mmap()` are used in application programs for handling video data captured at remote hosts.

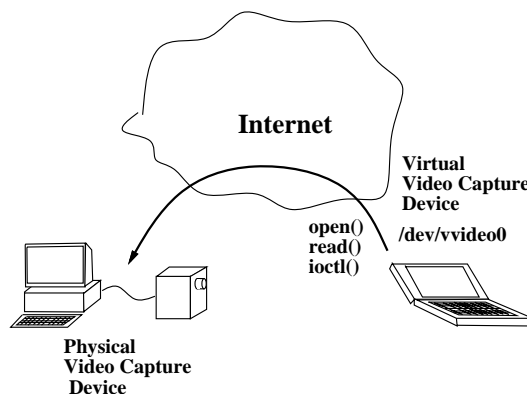


Figure 3. Transparent video capture device over Internet.

### Integration of control and video stream

Vvideo integrates both control devices for camera parameters and video capture devices. An application program can control a camera through the identifier of its video capture device. The control is executed by `ioctl()` system call, and parameters for the control are passed through the `ioctl()`. Synchronizing both control data and video data stream can be easily realized in application programs through vvideo.

Vvideo consists of three modules: client device driver, client daemon and server daemon. Fig. 4 shows relations among the three modules for vvideo.

### Client Device Driver

*Client device driver* is realized as a collection of system calls called by application programs. Two

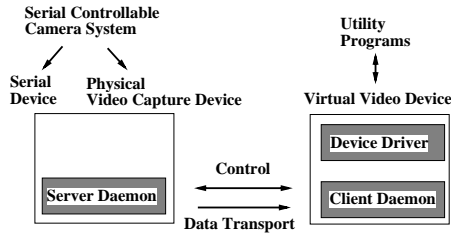


Figure 4. Modules for vvideo.

main system calls in the client device driver are `mmap()` for video capture and `ioctl()` for video capture device and control of camera.

The client device driver has a virtual frame buffer in memory for holding video images. Every minor device such as `/dev/video0` and `/dev/video1` has its own virtual frame buffer cache.

#### Client Daemon

*Client daemon* receives video data from some server daemon and updates the frame buffer for captured video data in the client device. And if the status of the controllable camera is changed by an application, the client daemon requests the server daemon to update the status of the remote camera.

The communication between the server daemon and the client daemon is executed when application programs request for video capturing. If there is no video capture requests from application programs, no video data is sent from the server to the client. This transport control is managed by the client daemon.

#### Server Daemon

*Server daemon* controls directly a camera of the remote host and sends video data according to the requests from the client daemon.

### 3.2 Spatio-temporal video information management

In this section, we propose Fragmented Reality(FR) and Recalled Reality(RR) model to accept multiple user's requests for a camera's direction (pan, tilt and zoom) at the same time. Fig. 5 overviews the relationship between the real world and FR/RR.

First, FR is a collection of fragmented images sampled by cameras. FR represents a kind of image database which consists of images and the relational

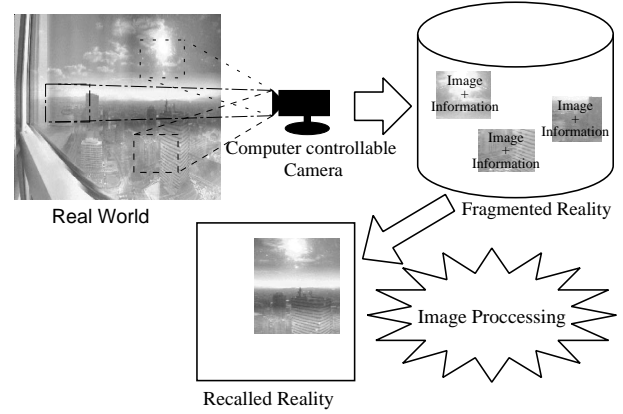


Figure 5. The real world and FR/RR.

information between the real world and the images. In this paper, we call the image *FR image* which has relational information concerning some real world objects.

On the other hand, RR is the world which is generated by FR. RR is realized by image processing including geometric transformations, combination, color blending, and so on.

The difference between RR and the real world depends on how to construct FR and the methods of image processing. It is important for FR to select which relational information and which method of image processing to be used.

## 4 Real applications of VVIDEO frameworks

This section presents some real applications based on vvideo frameworks.

### 4.1 Applying AVH to ordinary video applications

This section gives examples applying Augmented Video Hypermedia (AVH) to ordinary video applications on Internet. Vic (Video Conference Tools) and RealSystem are chosen as representatives of popular Internet video applications. Vic is a video conference tool which transports video data using Real-Time Transport Protocol. RealSystem is a video on demand (VOD) system. These tools use a local video capture device. AVH is realized by vvideo as the local video capture device and these tools can use AVH directly. Fig.6 shows the announcement of AVH to Internet using vvideo. Fig.7 presents the distribution of AVH to Internet as a VOD system using vvideo.

Fig.8 is an example of AVH applied for a panoramic image. The program which generates the panoramic image captured from a remote camera uses only local video capture device. Since the remote video capture device can be used transparently by vvideo, panoramic images can be generated from cameras located at anywhere on Internet.



Figure 6. Name-at using Video Conference Tool.



Figure 7. Name-at using RealSystem.

## 4.2 Web-Camera Systems based on FR/RR Model

In this section, we discuss the design of Web-Camera systems which is applicable to multiple users' different requests for camera's direction.

### 4.2.1 Designing a Web-Camera System

Fig. 4.2.1 shows the model of a Web-camera system. In this model, one of the most remarkable characteristics is that a sever constructs FR by making a database of images and statistics, and that each clients realize RR by retrieving the database of FR. This system is compatible with multiple users' different requests for camera direction.

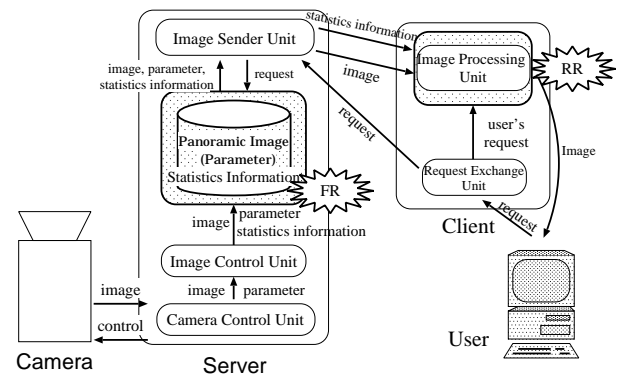


Figure 9. Model of Web-camera system.

*Server's* primary functions are to control cameras, to realize proper FR for the service and to provide images to clients. The server's functions consist of some units described below.

**camera control unit** It controls cameras by their parameters through vvideo device and captures images.

**FR constructing unit** It accumulates images of camera control unit, constructs FR to add parameter information to images, and calculate statistics information.

**image sender unit** In parallel with a user's request, it sends images and statistics information to the user from the constructed FR .

*Client's* primary functions are to exchange a user's abstract request, to proper request for FR, to get requested images from FR and to realize RR. The client's functions consist of some units described below.



Figure 8. Name-at using a panoramic image.

**request transformation unit** It receives a user's abstract requests and transforms them to FR parameters. It also gets proper FR to reconstruct RR.

**image processing unit** It makes RR images using FR images and the assignment of request transformation unit. It shows a live video to a user as a result.

#### 4.2.2 Service of Web-Camera System

If we provide a live video service on Internet which uses FR/RR model, it is important to consider how to construct FR. In this paper, we use a panoramic image to construct FR. Thus, users will be able to select parameters of pan, tilt, zoom, time and play speed (for past video only) depending on their intention. Applications such as WebView Livescope are used by many people to see sight view video, and this FR/RR model helps the applications to extend the possibility of service. But we have to consider trade-off of scalability and quality of service.

## 5 Conclusion

In this paper, we explained and showed the implementation of the vvideo, which is the new functions of the operating system for integrating multimedia

data distributed over Internet and 3D geographical information. And we explained the mechanism for the announcement method of the Augmented Video Hypermedia all over Internet. In this mechanism, we introduced the new model, Fragmented Reality (FR) and Recalled Reality (RR).

For further work, we have a plan to incorporate FR and RR into daemons, which cooperate with the vvideo for generating Augmented Video Hypermedia. Our proposed mechanism and the way of implementation will help to create new applications, which can make full use of the media existing on Internet and 3D geographical information.

## References

- [1] Michihiro MURAO, Masatoshi ARIKAWA, Koji OKAMURA: "Networked Augmented Spatial Hypermedia Systems on Internet", Proc. of IFIP TC2 WG2.6 15th Working Conference on Visual Database Systems, pp. 239 - 253. (2000).
- [2] Koji Okamura, Michihiro Murao, Masatoshi Arikawa: "Design of Network Transparentable Virtual Video Capture device", ISPJ-DPS, (1999) (in Japanese).