

Real-time and Object-based Video Streaming Techniques with Application to Communication System

Prof. Nitin. R. Talhar, Prof. Mrs. K. S. Thakare,

AISSMS College of Engineering, Pune, Pune University, INDIA
SIT, Sinhgad College of Engineering, Pune, Pune University, INDIA

Abstract. This is a component based system which allows the multiple users (clients) to communicate through video, audio and text on C2C and B2C websites. There is no component available in Market/Inter network which provides video, audio & text communication between clients without installing any 3rd party application on the client computer. This implementation will provide this component which can be used for the communication. Our goal is to enhance video-conferencing features. It proposes a real-time and object based video streaming techniques which can be deployed in telepresence environments. The streaming characteristics of real-time and object based video sequences are investigated under various system and networking conditions. Our video pipeline uses streams samples as basic primitives and is fully integrated with a communication framework handling acknowledgment information for reliable network transmissions and application control data. The video reconstruction process dynamically adapts to processing and networking bottlenecks. Our results show that a reliable transmission of our frame-based streaming leads to the best performance in terms of bandwidth. This implementation definitely will be the future video conferencing for effective communication. Also it reduces the cost and hence benefits to the society.

Keywords: C2C: Client to Client; B2C: Business to Client; VSMS: Video Streaming Media Server RIA: Rich Internet Applications; FCS: Flash Communication Server; RTP/RTCP: Real Time Control Protocol; SOAP: Simple Object Access Protocol; XSS: Cross Site Scripting; Codecs: Coder decoder; MPEG: Moving Picture Expert Group; API: Application Programming Interface; SSL: Secure Socket Layer; XML: Extensible Markup Language; FMS: Flash Media Server; URL: Uniform Resource Locator; HTTPS: Hyper Text Transfer Protocol over Secure Socket Layer; FTP: File Transfer Protocol

1. Introduction

Today and future is the era of information technology. Now a day's technology has grown up to the level best. It enhances the way we are communication with others. At present various communication systems are available, e.g. telecommunication through mobile/phone or using various devices; group/individual chatting in a network based environment and none the less video conferencing using various communication devices.

All the above means of communication requires complex setup process and also it involve cost. Hence these ways of communication are very cost effective.

Our main goal is to focus on implementation of real-time and object-based video streaming techniques with application to video conferencing. Now a day's market doesn't have a chip video conferencing system, i.e. it requires a setup processes and video conferencing devices. Hence idea comes in mind why not to create such a system which doesn't require a setup at user side (client side). It should have minimum interfacing devices such as camera and headset/microphone connected to the system. All conferencing handling issues will be take care by video streaming server. Hence the birth of the system.

1.1. Scope and Nature of the System

- Multiple users to communicate through video, audio and text.

- The Component-Class can be wrapped in different Websites CMS, existing and dominating the Internet.
- The Websites, not developed using CMS's can also make use of this component by adding a new webpage wrapped with it.
- Different Versions will be available for different requirement to capture the target audience!

Setup of video conferencing devices requires prior knowledge and skills and also it involves cost. To minimize the all these issues and to make it easier for a normal human being, we are implementing such a system to do video conferencing. Hence the motivation of this work originates. For the Solution of this Problem it requires:

- Video Streaming Media Server: Multimedia data such as video and audio will be process by Video Streaming Media Server (VSMS).
- Application Server: It will be for to send and receive the multimedia data video/audio/text streams from client to client.
- Web Server: Client must have internet connection hence client is connected to web server for global video conferencing facility.

2. System Architecture

Below Fig.1 depicts the main modules- Video Streaming Media Server (VSMS/FMS), Application Server and Web Server interconnected through internetwork.

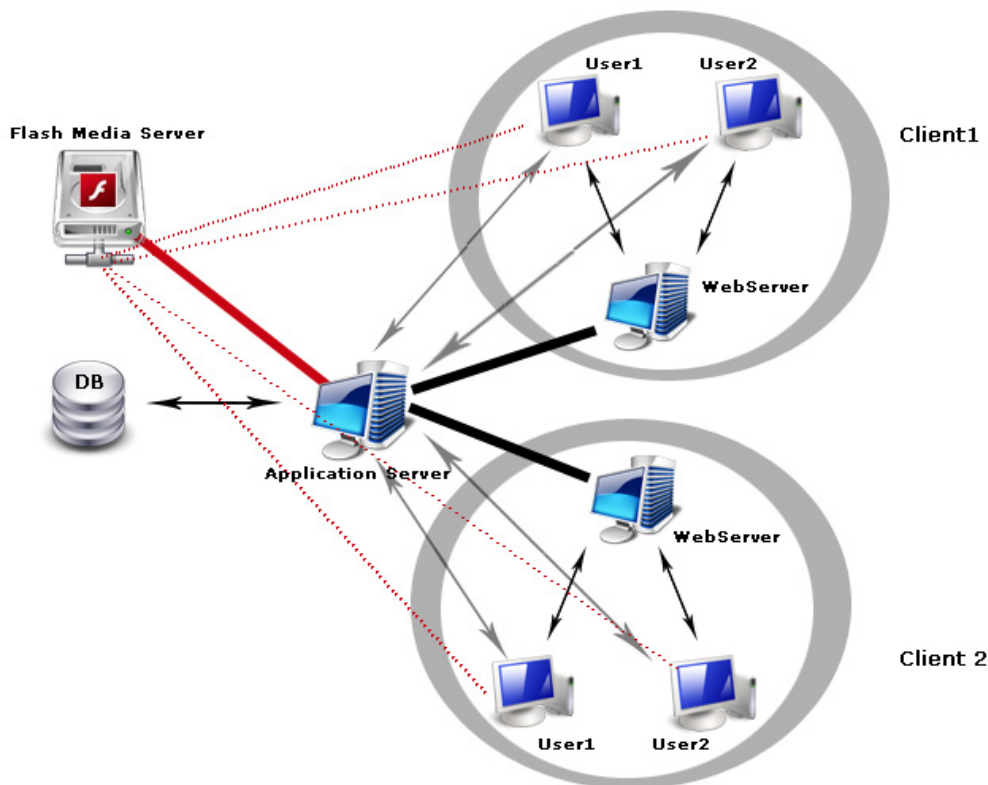


Fig. 1: Architecture of Communication System

The goal is to enhance collaborative VR applications with video-conferencing features. We propose a Real-time and Object-based video streaming technique which can be deployed in telepresence environments. The streaming characteristics of real-time video sequences are investigated under various system and networking conditions.

2.1. Real-Time Streaming of Video

Today, the robust generation and transmission of real-time 3D video is still a challenging problem. Our research on real-time 3D video systems is motivated by our interest in novel immersive projection and acquisition environments for telepresence. In particular, our 3D video system is composed of 16 camera

nodes, which acquire images and perform 2D image processing. The resulting information is streamed to a reconstruction node, which computes the actual 3D representation of the observed object. Camera and reconstruction nodes are at the same physical location and are connected in a local area network. The 3D video data is then streamed to a rendering node, which, in a real-world telepresence application, runs at a remote location.

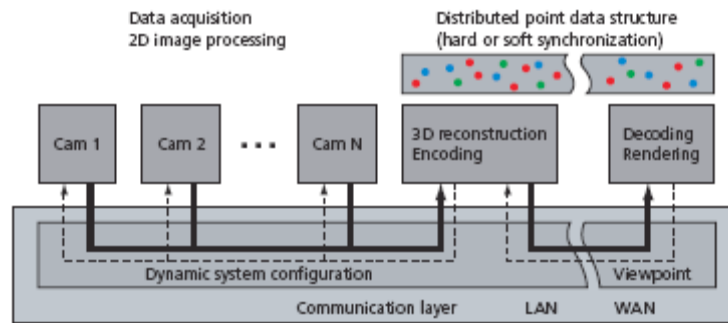


Fig. 2: Video System Architecture

As explained, rendering and reconstruction nodes need to share a common data structure, and, depending on the 3D video streaming procedure, this data structure must satisfy different consistency requirements. The overall system architecture is depicted in Fig. 2.

The main contribution of this paper consists in proposing a communication framework for distributed real-time 3D video reconstruction and rendering and in analyzing the transmission of the subsequent streams with respect to changing networking conditions.

2.2. Data Transmission

The communication layer API offers channel interfaces for transmitting and receiving data. The shared application data is organized in message objects which provide an interface for writing their payload into a transmission buffer. At the receiver, the transmission buffer is parsed, the messages are decoded and a callback mechanism informs the application of the arrival of a new message of a given type. Flow control and retransmission algorithms are applied to the transmission buffers only. A single transmission buffer may contain a set of messages of the same type or a collection of messages of different types. If the message size is greater than the size of a transmission buffer, a single message can be fragmented and distributed in multiple transmission buffers. All buffers are handled by memory pools such that all memory allocation is performed during start-up. The communication software also handles issues of cross platform interoperability, e.g. little-/big-endian conversions.

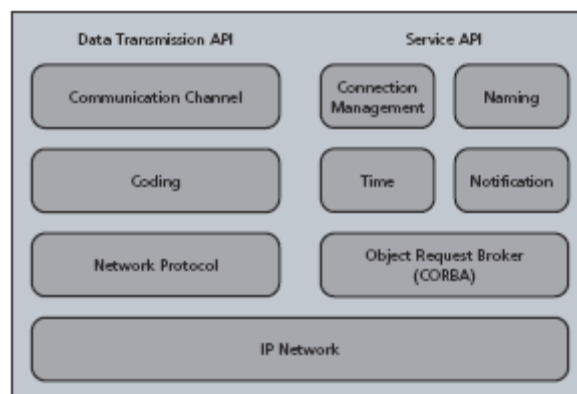


Fig. 3: Communication Framework Architecture

Real-time streaming data is transmitted by a communication channel consisting of a forward channel for payload data and a backward channel for control information and application feedback. This approach is inspired by the RTP/RTCP protocol suite, where RTP is used for unreliable streaming of payload data and RTCP is used for periodical exchange of control messages, indicating packet loss rates, packet jitter values

and possibly application specific data. The communication framework also offers a collection of codecs. Typically the data is entropy encoded before network transmission.

2.3. Video System

Fig.4 shows how the 3D video system uses various services offered by the communication framework. The camera nodes run a daemon which allows for the remote start-up and shut-down of applications. The reconstruction node identifies the camera nodes using the Naming Service and remotely starts the acquisition applications. The transmission channels between camera and reconstruction nodes are configured using the Connection Management Service. This service also keeps track of the state of the connections, and hence allows dealing with camera node failures. The pre-processed data from the camera nodes, i.e. silhouette and texture information, is streamed over reliable transmission channels. The associated backward channels carry the dynamic system configuration which is determined for each frame at the reconstruction node. Furthermore, the distribution of a global frame ID enables the synchronization of the running 3D video process with late joining camera nodes.

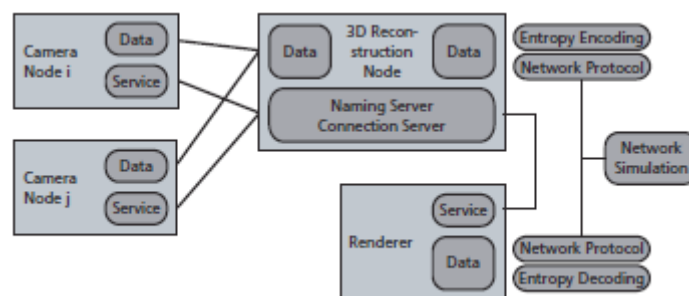


Fig. 4: Services for Real-time Video

2.4. Object-based Video Streaming Technique

To link the sender, receiver, transmission channel for real time displaying video sequences, a standard RTP based network streaming technique is adopted. The streaming module adopts two categories of network protocols containing the network-layer protocol and transport protocol. Based on IP network, the network layer protocol using network address to serve the basic network support. The majority of transport protocols perform over an RTP stack, which is implemented on top of UDP/IP to provide an end-to-end network transport for video streaming. The encoded bit stream is packetized into RTP packets and then transmitted using UDP. The receiver de-multiplexed RTP packets into the packet buffer for decoding and playback. Streaming QoS is not the scope of this paper and will be considered in future work. The MPEG-4 standard addresses a new object-based technique, which allows the different audio-visual objects in the scene to be encoded independently. In order to utilize the bandwidth efficiently, a mechanism is proposed to take benefits from the object-based compression standard. Important can be real-time encoded and transmitted with higher quality.

A streaming-based surveillance system needs efficient storage, transmission and manipulation of video data in multimedia environments. To achieve a best transporting of video sequence in a limited bandwidth, classification of important objects in a traffic scene into groups with high quality is required. The MPEG-4 standard addresses a new object-based technique, which allows the different audio-visual objects in the scene to be encoded independently.

3. User Interface Specifications And Features

Administrator Interface Specifications:

- Kick users
- Ban IP's (1 hour, 1 day, 1 week, etc...)
- Full access to private rooms (no password required)
- Option to be visible or invisible (can be disabled)
- Enhanced users list with IP display
- Enhanced rooms list with id display (for dropping users into certain rooms)

- Broadcasting a message to the clients over the network

End User Interface:

- Web camera and microphone automatic detection (best detection script)
- The names of the devices used for capturing audio/video are displayed.
- Rooms support with rooms list, join room and create room functions.
- Private (password protected) and public rooms support.
- Private messages (send messages to only one user) in separate windows.
- Public messages (messages seen by all users in a room).
- Bad words filter.
- One person can view 2, 3 or more video streams.

4. Acknowledgements

There are some key personalities involved, whose role has been very vital to pave way for success of the paper. We take the opportunity to express our sincere thanks and gratitude to them.

I would like to thank all the professors of Department of Information Technology of STES'S, Sinhgad College of Engineering, Pune. We are indebted to Prof. Mrs. K. S. Thakare my project guide who was very generous in sharing her time and knowledge with me and provided helpful and constructive criticism of early drafts of material. Last but not the least, I would like to extend my heart felt regards to all those who helped me directly or indirectly in the accomplishment of the paper.

5. References

- [1] Edouard Lamoray, Stephan Würmlin, and Markus Gross, authors of IEEE paper on "Real-Time Streaming of Point-Based 3D Video", Proceedings of the 2004 Virtual Reality (VR'04) 1087-8270/04\$20.00 IEEE. Virtual Reality 2004 March 27-31, Chicago, IL USA 0-7803-8415-6/04/\$20.00©2004 IEEE.
- [2] Ming-Ho Hsiao, Hui-Ping Kuo, Hui-Chun Wu, Yu-Kai Chen, Suh-Yin Lee, authors of IEEE paper on "*Object-Based Video Streaming Technique*", Proceedings of the 2004 IEEE International Conference on Networking, Sensing & Control Taipei, Taiwan, March 21-23, 2004, 0-7803-819 3-9/04/\$17.00©2004 IEEE.
- [3] Hulya Seferoglu, Ozgur Gurbuz, Ozgur Ercetin, Yucel Altunbasak, authors of IEEE paper on "*Video streaming to multiple clients over wireless local area networks*", Proceedings of the 2006 IEEE International Conference on Multimedia, 1-4244-0481-9/06/\$20.00©2006 IEEE.
- [4] Jozsef Vass, Xinhua Zhuang, authors of IEEE paper on "*Video streaming for wireless ATM networks*", Proceedings of ICSP2000 IEEE international conference on Multimedia, 0-7803-5747-7/00/\$10.00©2000 IEEE.
- [5] Neil J. McCurdy, William G. Griswold, and Leslie A. Lenert, authors of IEEE paper on "*A Robust Abstraction for First-Person Video Streaming: Techniques, Applications, and Experiments*", Proceedings of the Eighth IEEE International Symposium on Multimedia (ISM'06) 0-7695-2746-9/06 \$20.00 © 2006 IEEE.
- [6] Serkan Kiranyaz and Moncef Gabbouj, Senior Member, IEEE, authors of IEEE paper on "*Hierarchical Cellular Tree: An Efficient Indexing Scheme for Content-Based Retrieval on Multimedia Databases*", IEEE Transactions on Multimedia, vol. 9, no. 1, January 2007, 1520-9210/\$20.00 © 2006 IEEE