Bandwidth Analysis of the Ubiquitous Video Conferencing Application

Neil Arellano, Aleksander Milshteyn, Eric Diaz, Sergio Mendoza, Helen Boussalis and Charles Liu
Structures, Propulsion, and Control Engineering University Research Center (SPACE URC),
California State University, 5151 State University Drive, Los Angeles, CA 90032, U.S.A.

Keywords: Ubiquitous Video Conferencing (UVC), Semantic Information System Network, Tuple Space, Client-server Model, Bandwidth Metrics.

Abstract: The CSULA SPACE Center has endeavoured to develop the Semantic Information System (SIS) Network for real-time project collaboration. However, the lack of uniform, real-time communication platform application poses an inconvenience to the project collaborators, as they would be driven towards third-party communication applications, such as Skype, MSN, Yahoo Messenger, etc. The use of these commercial products does not incorporate moderation features between the network participants. In addition, these applications have various conference capacity limitations and their simultaneous multi-device sign-in feature can lead to possible concerns with information security (Alegre, 2009). The Ubiquitous Video Conferencing (UVC) application has been designed specifically for the SIS Network in order to provide its participants with dedicated multimedia channels and interactive communication. It is built on the integration of Qt libraries, audio/video codec libraries of FFMPEG, and the image-processing library Open Computer Vision. This paper presents the UVC application within the Semantic Information System Model and focuses on issues related to real-time bandwidth regulation.

1 INTRODUCTION

The trends of increasing computing power, affordability of hardware, and emergence of embedded networked systems have enabled businesses, researchers, and students the capability of accessing the information through technologies such as the Internet, technical software, and peer-to-peer communication.

Although web-browsers, social network applications, and voice/video communication enable users to identify "birds of a feather" communities, information is often scattered among various networks often hindering the ability for information updates to be synchronized in a seamless fashion.

Figure 1 shows an innovative SIS model of software architecture. It provides facilitation and management of client requests across a distributed server network. It is particularity suitable for information dissemination between project collaborators, particularly for researchers, educators, and students; specifically the use of the network by groups where geographic location, time, and computing resources limit information exchange and collaborative efforts amongst each other.

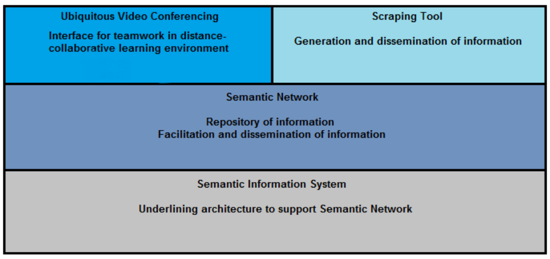

Figure 1: Semantic information system model.

The proposed Semantic Information System platform is best suited for educators, researchers, and team-project members with common interest. The platform allows these users to effectively generate, analyse, and disseminate information. SIS users will be able to "objectize" information nodes of their projects, which will generate a hierarchical tree structure to interrelate those nodes based on their semantic meaning and relationship to each other. These user-generated contents can be

Arellano N., Milshteyn A., Diaz E., Mendoza S., Boussalis H. and Liu C..
Bandwidth Analysis of the Ubiquitous Video Conferencing Application.
DOI: 10.5220/0004026103010304
In *Proceedings of the International Conference on Signal Processing and Multimedia Applications and Wireless Information Networks and Systems* (WINSYS-2012), pages 301-304
ISBN: 978-989-8565-25-9

accessed, updated, and shared with other network participants.

For the aforementioned SIS platform, a need for communication interface was born. In order for the project collaborators to interact with each other a Ubiquitous Video Conferencing (UVC) has been developed. It provides SIS users with real-time interactive application that supports audio, video, and textual means of communication for SIS project collaborations.

The paper is organized as follows: The Introduction of the SIS project establishes the need for UVC system and is followed by the detailed Facilitating Technology section. Then, the overview of the Client-Server Architecture and UVC Application Interface are discussed. The paper concludes with the Performance Analysis section, which presents bandwidth measurements.

2 FACILITATING TECHNOLOGY

The UVC application is utilized in conjunction with several underlying technologies that constitute to the development of the Semantic Information System. Figure 2 shows UVC application within the multi-server architecture, as well as structural component interdependency of the SIS platform. A Tuple Space programming paradigm allows initiation of client-server requests. Tuple Space supports automatic load balancing in parallel processing in a multi-thread/multi-core server. Active Directory Service System uses Tuple Space to provide server-to-server communication infrastructure for distributed server network configurations (Tolksdor, 2004).

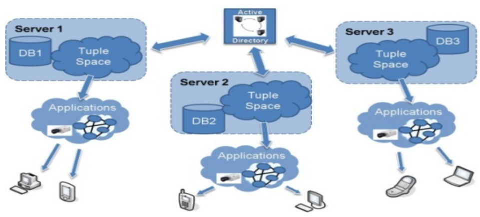

Figure 2: Overview of SIS scheme and applications.

A PostgreSQL Relational Database Management System performs data warehouse storage allocation and unique key assignments to the content within the network. The database keeps track of registered users and their access privileges play an important role in security.

Qt API is an open-source development environment that is geared toward design of GUI applications with extensive set of libraries available to the programmer. To facilitate audio and video compression, Open Computer Vision (OpenCV), FFMPEG, and Speex libraries were integrated to create the UVC Application. As mentioned before, the UVC application provides its participants with dedicated multimedia channels. Data streams through those channels can be compressed or encrypted based on different level of quality of service (QoS) or security demands. The mechanism of facilitating such needs is based on a reconfigurable embedded subsystem. The related technologies are beyond the scope of the paper and will be published elsewhere.

3 CLIENT-SERVER ARCHITECTURE

The UVC Application by default is designed to operate in a client-server environment. The client-server communication is handled through a Tuple Space model. In a Tuple Space, an "OUT" command deposits a tuple in, while an "IN" command retrieves a tuple based on the keyword matching. The tuples, described in XML, can be exchanged between a client and its local server.

A Hyper-Threading architecture defines different roles among the threads of each server. Figure 3 shows a controller and multiple worker threads. The controller thread deposits and manages the tuple into the Tuple Space. A worker thread retrieves a tuple and performs the task based on the requests described in a tuple. This delegation of roles allows for efficient communication and task scheduling, as well as automatic processor load balancing (Alegre, 2010).

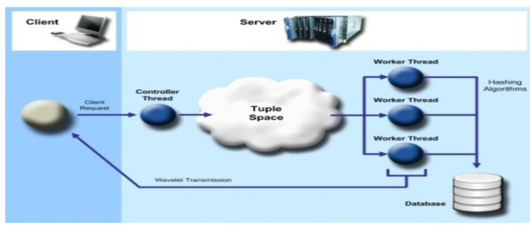

Figure 3: Tuple space architecture with controller and woker threads shown.

4 APPLICATION INTERFACE

The UVC is embedded in the Semantic Information Network, where conference participants are able to interact with each other. In project environments, a human moderation is necessary to organize and lead the videoconferencing sessions. In moderated sessions, an initiator of the videoconferencing is automatically assigned with Forum Coordinator privileges. When establishing a videoconference link, the Forum Coordinator of the group will initiate a session, thus relevant group members can join through an authentication process. Members will have their own user name and password, and will be required to enter an additional verification code for a specific video conferencing room. Its login panel GUI is displayed on Figure 19. Upon joining, members' data such as the user name, IP address, application capabilities (video/audio/text chat only) will be logged. The Forum Coordinator will be in charge of placing and lifting restrictions on each participant such as who can use video/audio channels. These operations will be done in the main tab of the UVC software shown on Figure 4.

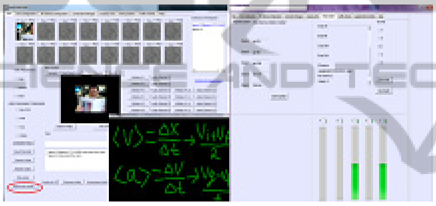

Figure 4: UVC Main GUI with video/audio, chat, finger-painting pad, and voting system shown.

Individual participants will also be able to regulate their own bandwidth, such as changing the video resolution, closing certain incoming video streams, adjusting their video frame rates

5 PERFORMANCE ANALYSIS OF VIDEO/AUDIO CHANNELS

5.1 Video Performance Characteristics

The server bandwidth is determined by $BW_s = (P * N) * S$, where BW_s is the bandwidth of the server, P is the number of senders, N is the number of receivers, and S is an average stream bit-rate of the encoded Audio/Video content (Prasad, 2003). The client bandwidth is calculated using the following formula: $BW_c = P * S$, where BW_c is the bandwidth of the client.

Network utilization is used to determine the activity of the network port on the client system. Three experiments have been performed in order to find the average ratio of the network usage to the maximum traffic. If the network utilization exceeds the threshold of 50%, it causes system instability. The sender is able to vary the FPS (frames per second) rate and hence, influence the network utilization. The first bandwidth experiment determined the percentage of network utilization for one video transmission with frame rates of 1, 5, 10, 15, 20, and 25 fps. Figure 5 displays a graph of incremental bandwidth usage as different frame rates are being established.

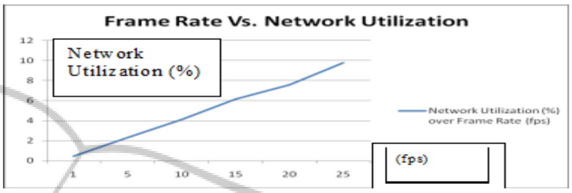

Figure 5: Frame rate versus network utilization.

This feature assists to the application users in determining the appropriate frame rate settings during various videoconferencing sessions. Depending on the amount of conference participants, as well as the number of active personal bandwidth consuming applications, users can vary their video frame rate, thus having an instant impact on their bandwidth and video quality. As the frame rate gets closer to the mark of 25 fps, the quality of the video improves, while decreasing frame rate frees up bandwidth resources.

The second bandwidth measurement experiment was conducted in order to determine the average amount of bandwidth utilized on a 100Mbps network, with the transmission frame rate of 25 fps. Figure 6 shows the network bandwidth usage with 25 fps rate for multiple UDP video connections.

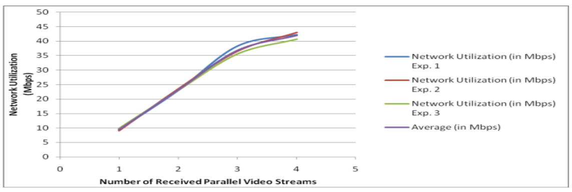

Figure 6: Average Network Utilization versus Number of Parallel Video Streams.

5.2 Audio Performance Characteristics

This section presents and discusses the experiments performed to analyze the video and audio bandwidth. In UVC, the video and audio channels run independently from one another. The user gets to decide which channels of communication to open or close. The UVC GUI has various options that provide users with an easy and efficient control panel. Video frame rate control and selection of audio transmission type are primary bandwidth regulators that UVC GUI offers to the end user. A client may choose from the three following types of audio transmission: Pulse-Code Modulation (PCM) raw audio streaming, Zlib codec, and Speex audio codec. The experiments on audio performance were done over IEEE 802.11g -54 Mbps network.

Audio transmission requires much less bandwidth then video. However, in a collaborative environment with multiple participants, who are actively involved in performing various data transferring operations, audio bandwidth conservation can also be application-critical. To offer audio compression solutions within UVC application, the Zlib and Speex audio codec were integrated to reduce the amount of bandwidth during transmissions. Since Zlib codec has been already incorporated within Qt library as its standard data compression scheme, it was chosen as the first compression option for audio transmission (Roelofs). Zlib furnishes users with a lossless scheme so the uncompressed output on the receiver's end is equal to the sender's raw data input. It was determined that, at least for this particular sequence of samples, the data compressed yielded a 29% of the original PCM data.

In order to have a more vigorous encoder, Speex codec library was selected to provide optimal compression ratio within the UVC application (Xiph Org.) Upon successful implementation of the codec, it was determined that on average Speex utilizes only 0.08% of the network bandwidth. Speex encoder on average is able to compress 75% of the original PCM data. Table 1 shows bandwidth utilization comparison between three methods of UVC audio transmission, while Table 2 displays bandwidth consumed by both audio and video channels.

Table 1: Average Network Utilization for Audio Transmission over IEEE 802.11g (54 Mbps).

Codec Type	Ave. Network Utilization (%)	Compression Ratio to Raw PCM
PCM	0.31%	1
Zlib	0.22%	0.71
Speex	0.08%	0.25

Table 2: Average Network Utilization for Audio and Video Transmission over IEEE 802.11g (54 Mbps).

Video at 25 fps with Audio Codec Type	Ave. Network Utilization (%)
Video + PCM	19.32%
Video+Zlib	19.24%
Video + Speex	19.08%

6 CONCLUSIONS

SIS participants are offered with a repertoire of video and audio controls while using UVC application. These transmission options provide flexibility for bandwidth control, as the Forum Coordinator and other participants can regulate their network bandwidth in order to accommodate more conference participants or enhance the quality of the video/audio streams. The multithreading processes of providing individual audio and video channels, along with real-time video frame rate control and audio transmission selection - add versatility to the Ubiquitous Video Conferencing application. These features enhance efficiency within the whole SIS platform, as participants have direct control over real-time communication channels. The performance analysis of UVC application shows the benefits of incorporated application options that allow flexible real-time bandwidth regulation during videoconferencing sessions.

ACKNOWLEDGEMENTS

Acknowledgement to NASA University Research Center Program, GRANT # NNX08A44A.

REFERENCES

A. Alegre 2009. Aerospace Information Server. California State University Competition, 2009. *Structures, Propulsions, and Control Engineering (SPACE) Center* – Research Topic: Ubiquitous Computing and Embedded Architectures.

A. Alegre, S. Beltran, J. Estrada, A. Milshteyn, A. Lam, JP Adigwu, C. Liu, H. Boussalis May 2010. Ubiquitous Video Conferencing for Obectizing Contents in a Semantic Network. *4th International Conference on New Trends in Information Science and Service Science (NISS2010)*, Gyeongju, Korea.

R. Tolksdorf, F. Liebsch, and D. Minh Nguyen, 2004. XMLSpaces.NET: An Extensible Tuplespace as XML Middleware.

R. S. Prasad, M. Murray, C. Dovrolis, K. Claffy, 2003. Bandwidth estimation: metrics, measurement techniques, and tools.

G. Roelofs, J, Gailly and M. Adler. ZLIB - A Massively Spiffy Yet Delicately Unobtrusive Compression Library.

Xiph Org. Speex - an Open Source/Free Software patent-free audio compression format designed for speech.