INVESTIGATION OF RESOURCE USAGE AND VIDEO QUALITY WITH DIFFERENT FORMATS IN VIDEO BROADCASTING

HAZAR JALAL YAQO SHAMAYA

JULY 2014

INVESTIGATION OF RESOURCE USAGE AND VIDEO QUALITY WITH DIFFERENT FORMATS IN VIDEO BROADCASTING

A THESIS SUBMITTED TO

THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES OF

ÇANKAYA UNIVERSITY

BY

HAZAR JALAL YAQO SHAMAYA

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

IN

THE DEPARTMENT OF

MATHEMATICS AND COMPUTER SCIENCE

JULY 2014

Title of the thesis: **Investigation of Resource Usage and Video Quality with different Formats in Video Broadcasting.**

Submitted by **Hazar Jalal Yaqo Shamaya**

Approval of the Graduate School of Natural and Applied Science, Çankaya University.

Prof. Dr. Taner ALTUNOK
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Prof. Dr. Billur KAYMAKÇALAN
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. Subhi R. M. ZEEBAREE Co-supervisor	Assist. Prof. Dr. Ö. Tolga PUSATLI Supervisor

Examination Date: 31/07/2014

Examination Committee Members

Assist. Prof. Dr. Ö. Tolga PUSATLI	(Çankaya Univ.)	
Assist. Prof. Dr. Reza HASSANPOUR	(Çankaya Univ.)	
Assoc. Prof. Dr. Fahd JARAD	(THK Univ.)	

STATEMENT OF NON PLAGIARISM PAGE

I hereby declare that academic rules and ethical conduct have been observed in the obtaining and presentation of all the information given in this document. It is also declared that as per these rules and conduct, all the materials and results not original to this work are fully cited and referenced.

Name, Last Name : Hazar Jalal Y., SHAMAYA

Signature :

Date : 31.07.2014

ABSTRACT

INVESTIGATION OF RESOURCE USAGE AND VIDEO QUALITY WITH DIFFERENT FORMATS IN VIDEO BROADCASTING

SHAMAYA, Hazar Jalal Yaqo
M.Sc., Department of Mathematics and Computer Science
Supervisor: Assist. Prof. Dr. Ö. Tolga PUSATLI
Co-supervisor: Assist. Prof. Dr. Subhi R. M. ZEEBAREE
July 2014, 51 pages

Transfer of video through an Internet network is one of the greatest challenges faced by the users, especially when dealing with broadcasting e.g. online broadcasting, and large video sizes. Video codecs make this operation easier. In this thesis, we submit the video compression issue to bring us closer to a better outcome/solution by asking the question: “Does compressing videos before sending to the client have considerable benefit?”

The experiment methodology we follow is to capture video from multiple web cameras, encoding this video in one server in order to transfer it to another server, and from the second server we can transfer this video to all users wirelessly. Moreover, we can apply this operation to send more than one video to multiple users.

The literature surveyed has helped the researcher to view ideas such as using the MVC (Multi-View Video Coding) and UDMVT (User Dependent Multi-View Video Transmission) codecs to compress video, and how to carry out multi-video, in addition to see the effectiveness of wired and wireless modes of video transfer.

The researcher used the MVC codec to transfer video, and we can see the effectiveness of transferring video via wired and the effectiveness is reduced in the wireless mode. UDMVT was not used due to the fact that a number of algorithms required for this codec and they could not be obtained. However, the UDMVT codec may be an avenue of exploration in future studies.

Keywords: Multi-View Video, Wireless Network, Codec, MVC, UDMVT.

ÖZ

KAYNAK KULLANIMI VE ÇEŞİTLİ BİÇİMLERDE YAYINLANAN VİDEO KALİTESİ ÜZERİNE ARAŞTIRMA

SHAMAYA, Hazar Jalal Yaqo
Yüksek Lisans, Matematik-Bilgisayar Anabilim Dalı
Tez Yöneticisi Yrd. Doç. Dr. Ö. Tolga PUSATLI
Yardımcı Tez Yöneticisi Yrd. Doç. Dr. Subhi R. M. ZEEBAREE
Temmuz 2014, 51 sayfa

Genel ağ üzerinden video aktarımı, özellikle çevirimiçi işlemlerde olduğu gibi yayın aşamasında ve büyük boyutlara sahip videolar kullanırken, tüm kullanıcıların karşılaştığı en büyük sorunlardan biridir. Video kodu çözücüsüleri (codec) bu işlemi kolaylaştırır. Bu tezde, bu işlemle ilgili yardımcı olmak amacıyla sıkıştırılmış videoya ilişkin bilgiler sunuyoruz. Araştırma sorusu ise şudur: "istemciye göndermeden önce sıkıştırılmış videoların kayda değer bir faydası var mı?"

Deney yöntemi olarak, çoklu web kameralardan videolar elde etmek ve onları birinci sunucuda kodlayıp ikinci sunucuya ve oradan da istemcilere kablosuz aktarmayı seçtik.. Bu işlemi gerçekleştirerek bir videoyu birçok kullanıcıya da gönderebiliriz.

Taranan literatür, araştırmacının videoları sıkıştırmak için MVC ve UDMVT kod çözücüleri kullanma ve birçok videoyu nasıl kullanabileği fikrini gözden geçirmesine, bunun yanı sıra videoları kablolu ve kablosuz modda aktarmanın etkili yolunu görmesine yardımcı olmuştur.

Araştırmacı, video aktarmak için MVC kod çözücüyü kullanmıştır. Video aktarımının kablolu bağlantıda etkili olduğunu, kablosuzda ise etkisinin düştüğünü görüyoruz. UDMVT kod çözücüyü uygulamak için gerekli olan bazı algoritmalara sahip olunmadığından bu kod çözücüyü ileri zamanlarda yapacağı çalışmalarda kullanılması düşünülmektedir.

Anahtar Kelimeler: Çoklu Görüntülü Video, Kablosuz Ağ, Video Kodu, MVC, UDMV.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to my thesis advisor Assist. Prof. Dr. Ö. Tolga PUSATLI for the invariable guidance and support of my Master of Science studies and research in addition to the devotion, sincerity, motivation and immense knowledge on his part. His guidance helped me through the entire course of my research and writing of this thesis. I firmly believe that there could not have been a better advisor and mentor than him for my Master of Science studies.

Finally, I would like to express my deep gratitude to my family, my dear father, mother and brothers for their endless and continuous encouragement and support throughout the years.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURES

FIGURES

LIST OF TABLES

TABLES

LIST OF ABBREVIATIONS

PSNR	Peak Signal to Noise Ratio
SSIM	The Structural Similarity Index
MVC	Multi-View Video Coding
UDMVT	User Dependent Multi-View Video Transmission
AVC	Advanced Video Coding
USB	Universal Serial Bus
LAN	Local Area Network
QoS	Quality of Service

CHAPTER 1

INTRODUCTION

1.1 The Purpose

In the last two decades, the rapid development of digital video technologies had an impact on different areas which are important in our daily life including education, medicine and communications. We can watch a 3D show filmed from multi-video sites through the Internet, on sites such as *YouTube* as well as monitor medical surgical operations occurring in a distant country.

Many areas in digital video technologies have had a significant stake in the field of industry and scientific research. Because of their influence on this development, video compression technologies and video transmission are needed to be promoted.

The limitations of storage space and bandwidth transmission in digital video technologies have been reduced using video compression technologies.

Basically, two media – wired and wireless – are predominantly used in the network layer of video transmission. The wireless environment faces many challenges in the process of transferring video compression from exposure to the loss of video frames. These losses lead to poor quality images.

The image of a single scene taken from multiple cameras at various angles is a technology which is used in a number of different Areas such as scientific studies, medical operations and security systems. This technology is termed as multi-view videos.

Examples of using multi-view videos can be Free Viewpoint TV (FTV), remote security monitoring, undersea surveillance systems and remote medical surgery. The benefit of multi-view videos is that the appropriate images for the viewer can be chosen when multiple shots are taken by multiple cameras.

After multiple shots are taken and compressed, the compressed video is transmitted through the network and then the 3D video is observed by viewers. However, when the compressed video is transmitted, the bit-rate is remarkably larger than the bit-rate of a single video and this will cause an increased traffic in the bandwidth.

1.2 Purpose and Scope of Thesis

In this thesis, we aim to investigate effects of different formats of video transmission from server to client. This video can be encoded in the server by a video frame which can be taken from a camera. The encoded video may be transmitted through the network and later decoded by a decoder by clients. The video can then be played back and watched. This video can be transmitted in one step in a wired or wireless environment.

We build a working environment to process multi-view video codecs and transport; this working environment is to test two different codecs in a wireless transmission. In addition, we rely on certain criteria and algorithms.

With this aim and scope, we observe the strengths and weaknesses of video quality and CPU consumption during different test runs. By doing so, we are able to discuss the benefits and costs of compressing videos across a wireless environment on different client machines. Hence, the research question addressed in this thesis is as follows:

“Does compressing videos before sending them to the client have considerable benefit?”

CHAPTER 2

LITERATURE SURVEY AND BACKGROUND

2.1 Background

2.1.1 Data communications

Our communication is the sharing of information and this sharing may be either local or remote. Local communication is the mode such that two or more individuals usually communicate face to face, while remote communication takes place between individuals who communicate over a distance. The term telecommunication, which comprises telegraphy, television and telephony, stands for communication at a distance (*tele* is derived from Greek root meaning "far"). 'Data' is a reference to information demonstrated in whatever form which has an agreement by the parties creating and making use of the data. Data communications imply the interchanging of data between two devices through some medium of transmission such as a wire cable. In order to manifest data communications, two communication systems are needed in the communicating devices in order to transmit data communication. These systems are characterized by a combination of software (programs) and hardware (physical equipment) [1].

Characteristics of data communications: As for the efficacy of a data communications system, four fundamental characteristics are relied upon: delivery, accuracy, timeliness, and jitter.

1. Delivery: For the purpose of delivering data to the intended device or user, data must be sent by the system to the correct destination.
2. Accuracy: whereby data needs to be delivered accurately. Data which have been changed in transmission and left uncorrected is deemed unusable.
3. Timeliness: the time during which data is delivered as soon as they are produced, in the original order and without any significant delay. In the case of video and

audio, timely delivery is important and any delay in data delivery of the product (video and audio) is useless.

4. Jitter: This is the variance in the time of packet arrival or the irregular delay in the delivery of multimedia packets. To illustrate, if a video packet is delivered every 30 milliseconds and some of the packets arrive with a 30-millisecond delay while others arrive with a 40- millisecond delay, the result is an uneven quality in the video.

Components of data communications: There are five components of a data communications system (Figure 1).

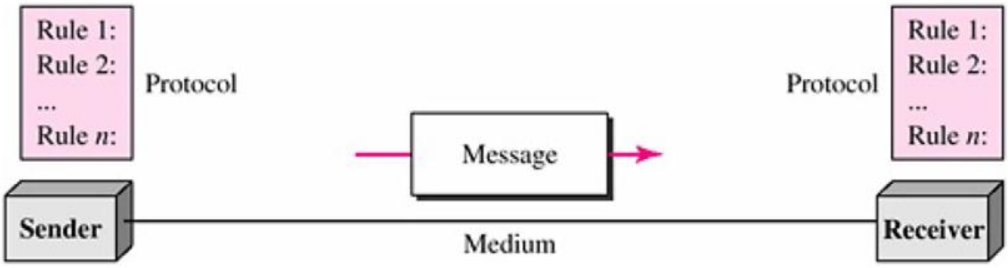


Figure 1 Components of data communications

1. Message. This is the communicated data such as pictures, numbers, text, audio and video.
2. Sender. This is the machine or device which is used to send the data message such as a computer, workstation, telephone handset or video camera.
3. Receiver. This is the machine or device which is utilized to receive the message such as a computer, workstation, telephone handset or television.
4. Transmission medium. This is the physical path whereby a message is transmitted from sender to receiver. Twisted-pair wire, coaxial cable, fiber-optic cable, and radio waves are examples of transmission media.
5. Protocol. This stands for a list of rules that govern data communications which also denotes an agreement between the communicating devices or machines. Connection between two devices can occur without communication when the protocol is not set up. For example, a person who speaks English are cannot be understood by a person how speaks Arabic. [1]

Data Representation: There can be different forms of data delivery such as numbers, images, text, audio and video.

2.1.2 Computer network

A computer network enables two or more devices to communicate and exchange data. Owing to the support these give to various applications, such devices are computer-based client devices, servers, and peripherals. Figure 1 shows a diagram of a computer network. With regard to client devices, a network can share data through a service called a cloud, which provides interconnections (such as routing of data, voice, and video) among client devices and servers.

Networks: these are sets of devices which originate, route, terminate data and are connected by communication links. Often referred to as nodes, these devices may be computers, printers, or any other device with the capability of sending and/or receiving data which produced by other nodes on the network.

Client/Server computing: this is a model which is a distributed application structure. It is structured such that it splits tasks or workloads between the providers of a resource or service known as servers, also in addition to service requesters and named clients. The communication on the part of clients and servers often takes place over a computer network on separate hardware while a single system may accommodate both client and server. [2]

Through the network a client can receive service from one or several servers without the need of both client and server to locate in the same location. Similarity, the service is able to supply service to many clients when the clients and service rely on two or more independent devices on the network (Figure 2) [2]

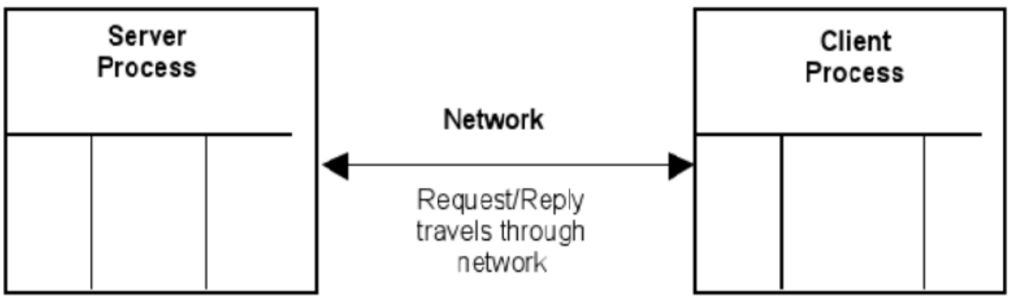


Figure 2 Client/server model

Network media: This is the media related to the Internet and uses the following types of media [3]:

1. Metallic wire: twisted pair cabling, the most common being category 5 or 6, which is a component of conventional network wiring. The form of the electrical current of digital data, video and voice signals can run through the metallic wire. Twisted pair cabling can support data rates toward the gigabit per second (Gbps) range. IEEE 802.3 (Ethernet) specifies the use of twisted pair wiring.
2. Optical fiber consisting of strands of glass is a cable that efficiently carries light between the two ends. Optical fiber has an advantage over metallic wiring such that the former may potentially support considerably higher data rates (up to terabits per second) without an electromagnetic field, and covering relatively longer distances. Moreover, connections are faster between buildings throughout cities in comparison to metallic wire. Optical fiber is nevertheless considered disadvantageous on account of its being relatively expensive to install for each client device within a company.
3. Void: It is a medium of transmission of electromagnetic waves, such as radio waves or infrared light, and to transport data, video and voice signals via electromagnetic waves. The most common signaling frequency for wireless networks is 2.4 GHz or higher frequencies for radio waves. Most IEEE 802 standards (such as 802.11a, 802.11b, 802.11g, 802.11n, 802.15, and 802.16) use radio waves, whereas an infrared version of 802.11 is used; however, there are very few, applications.

2.1.3 Bandwidth

Bandwidth: shows a measurement of the available bit-rate or consumed data communication resources which are expressed in bits per second or multiples thereof [1]. The difference between two numbers is normally denoted by bandwidth. For instance, the bandwidth of a composite signal is 5000 – 1000, or 4000.

The difference between the highest and lowest frequencies of a signal is the component of the signal bandwidth. [1].

The signal bandwidth can be classified into two signal types: periodic signal and nonperiodic signal. Figure 3 shows the difference between periodic and nonperiodic signals. Periodic signal contains frequencies from 1000 to 5000 whereas the nonperiodic signals have the same frequencies from 1000 to 5000 but the frequencies are continuous. [1]

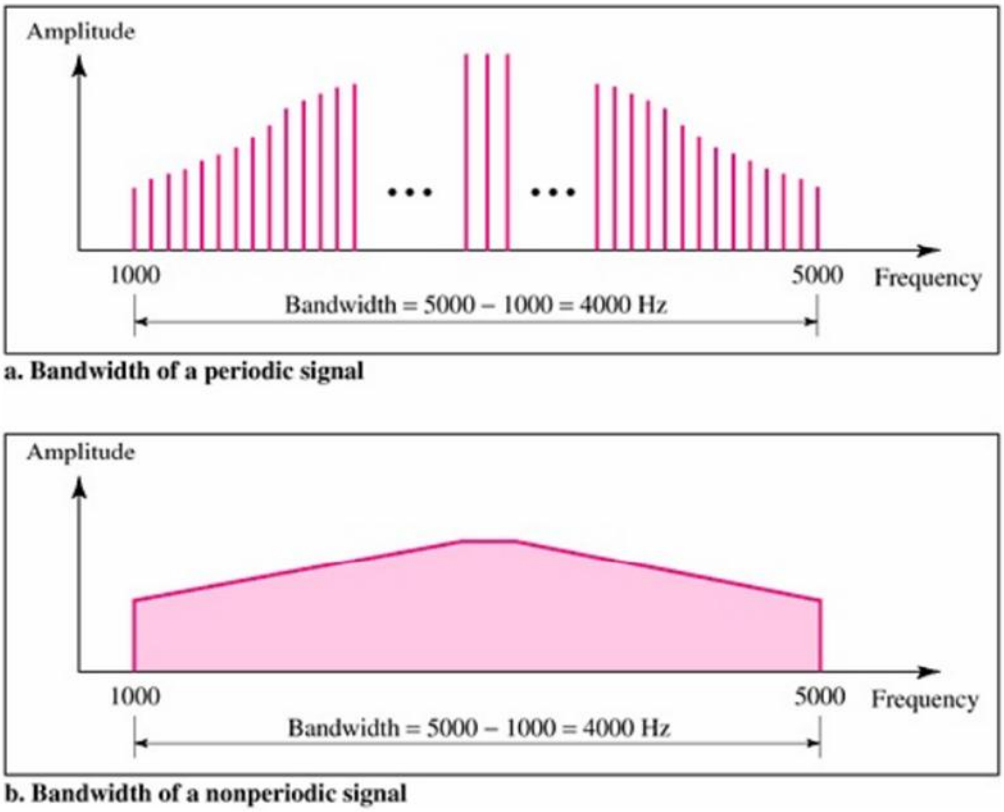


Figure 3 Bandwidth of periodic and non-periodic composite signals [1]

Bit Rate: The number of bits conveyed or processed per unit time (in seconds) is called the bit rate. This is expressed in bits per second (bps). A bit rate of two signals is shown in figure 3.16.

An example for bit rate is shown by [1]: What would be the required bit rate of a channel if we were to download text documents at a rate of 100 pages per minute؟

Solution: A page comprises approximately 24 lines with each line carrying 80 characters. Assuming that that one character consumes 8 bits, the bit rate is

$100 \times 24 \times 80 \times 8 = 1{,}636{,}000$bps = 1.636 Mbps.

2.1.4 Video sequence

A video sequence is defined as a sequence of images which is captured and represented in its basic form. These images are commonly referred as *frames*. The frames are shown at a stable rate, called the frame rate (measured in frames/second). 30 and 25 frames per second are most commonly used. Analogue video signals are produced by scanning a 2-D scene which is then converted to a 1-D signal. The analogue video signal is digitized by the process of filtering, sampling and quantization. [4]

2.1.5 Compression

Compression is the process where data is compacted from a large number of bits into a smaller number and this compression is used in digital video to reduce the number of bits. Digital video can be converted into a format which is suitable for transmitting and storing, leading to a reduction of bit number. This process is termed as *video compression* or *video coding*. For example, 'raw' or uncompressed digital video is approximately 216Mbits for 1 second (which is a large bitrate) of uncompressed standard definition video, and in order to store and transmit this digital video, compression is needed. [5]

A complementary pair of systems, a compressor (encoder) and a decompressor (decoder) is what comprises a compressor. The source data is converted by the encoder into a compressed set of data (or compressed data). This is achieved through occupying a reduced number of bits, before storage or transmission. The compressed data is then converted by the decoder into a representation of the original video data. CODEC (enCOder/DECoder) is the term commonly used for the encoder/decoder pair (Figure 4).

Figure 4 Encoder / decoder

2.1.6 Video resolution and bandwidth requirement

The video bandwidth requirement is related to the video resolution. [4] shows data in Table 1 in which the bandwidth requirement for selected video formats is based on the Common Intermediate Format (CIF), 25 frames per second for example and obviously the higher video resolution requires higher video bandwidth. [4]

Format	Resolution (horizontal × vertical)	Bits per frame	Bandwidth
Sub-QCIF	128 × 96	147456	3.68 Mbit/s
QCIF	176 × 144	304128	7.6 Mbit/s
CIF	352 × 288	1216512	30.42 Mbit/s
4CIF	704 × 576	4866048	121.65 Mbit/s
16CIF	1408 × 1152	19464192	486.6 Mbit/s

Table 1 Bandwidth Requirement for Selected Video Formats

On account of the immense amount of data, turning communication and storage capabilities for these data is both expensive and limited. Uncompressed (raw) video is considered disadvantageous. The raw data rates of a number of typical video formats are shown in Table 2, while Table 3 shows various typical video applications and their respective bandwidths. [6]

Format	Raw data rate
HDTV	1.09 Gbits/s
CCIR-601	165.89 Mbits/s
CIF @ 15 f.p.s.	18.24 Mbits/s
QCIF @ 10 f.p.s.	3.04 Mbits/s

Table 2 Raw Data Rates of Typical Video Formats

Application	Bandwidth
HDTV (6 MHz channel)	20 Mbits/s
Desktop video (CD-ROM)	1.5 Mbits/s
Videoconferencing (ISDN)	384 kbits/s
Videophone (PSTN)	56 kbits/s
Videophone (GSM)	10 kbits/s

Table 3 Typical Video Applications

For example, a 2-h CCIR-601 is color move and storing this move without compression in the optical disk (CD) of 5-Gbit would take 30 s. Therefore, 36 days is the time which is needed to deliver the entire move to other end of a 384 Kbits/s Integrated Services Digital Network (ISDN) if the move was not compressed. [4]

[4] Shows that compression of large amount of video data is required to enable this data to be transmitted or stored. However, there is a challenge to use video compression for reduce the size of video data. "Video coding" and "video compression" are therefore terms used interchangeably and these are important components of digital video services.

2.1.7 Video format

1. Intermediate formats: These are compressed and converted formats for video compression algorithms before compression and transmission. Listed in the Table 4 [5] is a popular set of formats which are the CIF.

Format	Luminance resolution (horiz. × vert.)	Bits per frame (4:2:0, 8 bits per sample)
Sub-QCIF	128 × 96	147456
Quarter CIF (QCIF)	176 × 144	304128
CIF	352 × 288	1216512
4CIF	704 × 576	4866048

Table 4 Video Frame Formats

To illustrate, for standard-definition television and DVD-video, 4CIF is appropriate. Moreover, CIF and QCIF are popular for videoconferencing applications, as well. With regard to mobile multimedia applications, QCIF or SQCIF are deemed favorable due to the fact that these codecs offer limited display resolution and bitrate.

2. High definition: There are several high definition (HD) video formats and the most commonly used are shown in Table 5 [5]. Standard Definition (SD), however, is a format used widely for digitally coding video signals in television production.

Format	Progressive or Interlaced	Horizontal pixels	Vertical pixels	Frames or fields per second
720p	Progressive	1280	720	25 frames
1080i	Interlaced	1920	1080	50 fields
1080p	Progressive	1920	1080	25 frames

Table 5 HD Display Formats

Clearly, the requirements of the HD format are an even larger uncompressed storage or transmission rate in comparison to SD formats. SD video has (720 × 576 × 25 =) 10368000 displayed pixels per second. 720p HD video has (1280 × 720 × 25 =) 23040000 displayed pixels per second and at the top end, 1080p HD has (1920 × 1080 × 25 =) 51840000 displayed pixels per second. These resolutions in uncompressed video require large storage or large transmission capacity with high resolutions, very large storage or transmission capacity; therefore, an essential compression for video is needed in practical applications (5).

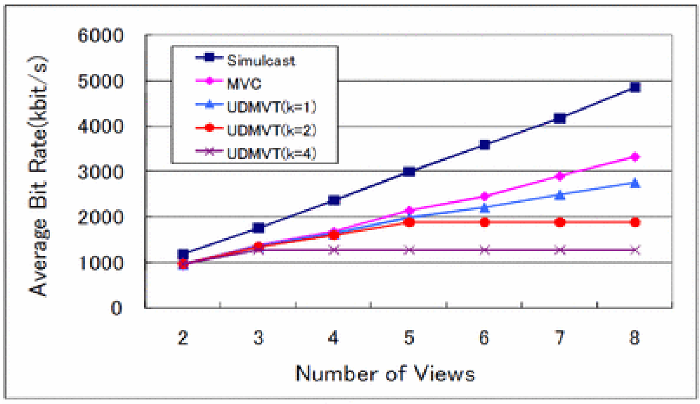
2.1.8 Measurement of quality

1. Peak Signal to Noise Ratio (PSNR): Being a very popular quality measure, it is both easy and quick to measure. It is widely used to compare the 'quality' of compressed and decompressed video images. [17]
2. The Structural Similarity (SSIM) Index: For the purpose of measuring the similarity between two images, this method is used. One may view the SSIM

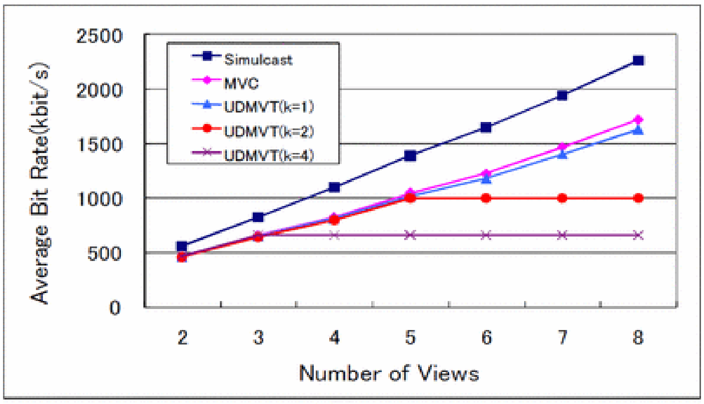
index as a quality measure of one of the images being compared provided the other image is considered to have perfect quality. [17]

2.2 Literature Survey

Pan and colleagues [7] showed that multi-view video is made up of multiple video sequences. It is from multiple angles and positions that these video sequences are taken by multi-cameras. On account of the fact that the multi-view video comprises multiple video sequences, a several times more traffic is required than is the case for traditional multimedia, which therefore dramatically increases the bandwidth requirements. Pan et al made performance evaluations between simulcast, MVC (Multi-view Video Coding) and UDMVT (User Dependent Multi-view Video Transmission) in order to reduce the rate-bit of multi-view video transmission. They proved that unlike simulcast and MVC, UDMVT greatly reduces the average bit-rate. The multi-view video test sequences “Ballroom” and “Vassar” with 8 views presented in [7] showing that some frames are repeated in MVC. In order to encode the multi-view video sequences, encoders implemented by the modified open source project JMVC were brought into use. 250 frames were encoded with a frame rate of 25 f/s. Figure 5 shows the Average Bit Rate of traffic. Figure 6 shows Peak Signal to Noise Rate (PSNR), and table 6 shows Qualitative Comparison.

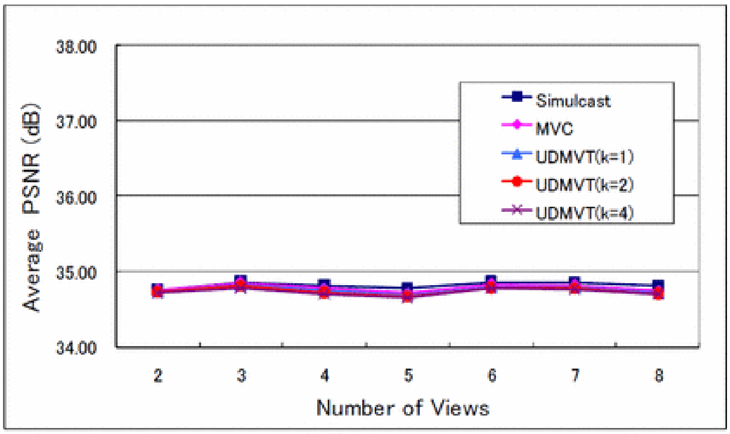


(a) Ballroom

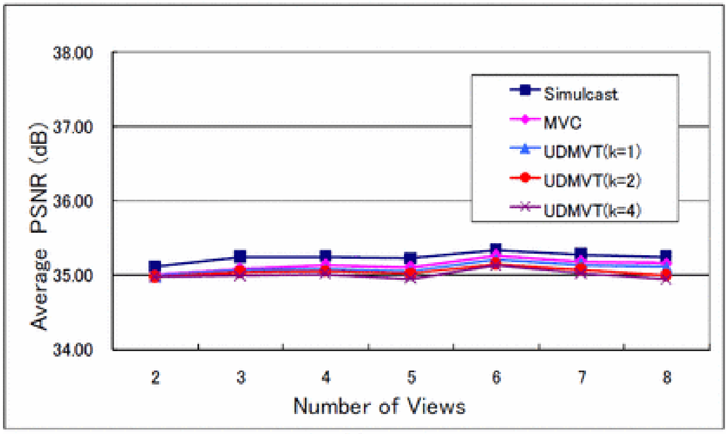


(b) Vassar

Figure 5 Average bit rate (ABR) of each scheme



(a) Ballroom



(b) Vassar

Figure 6 Average PSNR of each scheme

	Spatial and Temporal redundancy	Inter-view	Low traffic	High hit rate	Quality control	Tradeoff transmission and computation
Simulcast	YES	NO	NO	YES	NO	NO
MVC	YES	YES	NO	YES	NO	NO
CDSS [7]	YES	YES	YES	NO	YES	NO
IMVS [10]	YES	YES	YES	NO	NO	YES
UDMVT	YES	YES	YES	YES	NO	YES

Table 6 Qualitative Comparison

Moreover, they [8] studied the application of UDMVT in FVT, which showed that several problems should be taken into account in the application of FVT in terms of

User Interface, Storage, Encoder and Networks. Ziyuan Pan et al. considered the following steps to use UDMVT and FVT:

1. User Interface: There should be compatibility between the user interface and the successive motion model.
2. Storage: For storage purposes, the prediction structure of UDMVT is not appropriate since the captured raw data of FTV is massive.
3. Encoder: When multiple users are viewing the same program simultaneously, a special encoder should be used to encode frames of various ranges.
4. Networks: In cases of limited bandwidth, in a network such as a wireless network, UDMVT is suitable as it reduces the traffic. However does not suit P2P networks and multicast.

Nem̌ić, O. et al. [9] showed the advantages of multi-view video coding (MVC) by examining the encoding of two data sets (MVC and H.264/AVC reference software) captured with multiple cameras. The peak signal-to-noise ratio (PSNR) and Structural Similarity (SSIM) Index were used as objective quality metrics.

Figure 7 respectively represents the PSNR Y and SSIM Y results of encoding the ballroom sequence. The solid lines on both graphs represent the averages of the PSNR Y and SSIM Y values across all camera views obtained by MVC for a set of targeted average bitrates. The dashed lines represent the average PSNR Y and SSIM Y values obtained using H.264/AVC at roughly the same average bitrate as MVC.

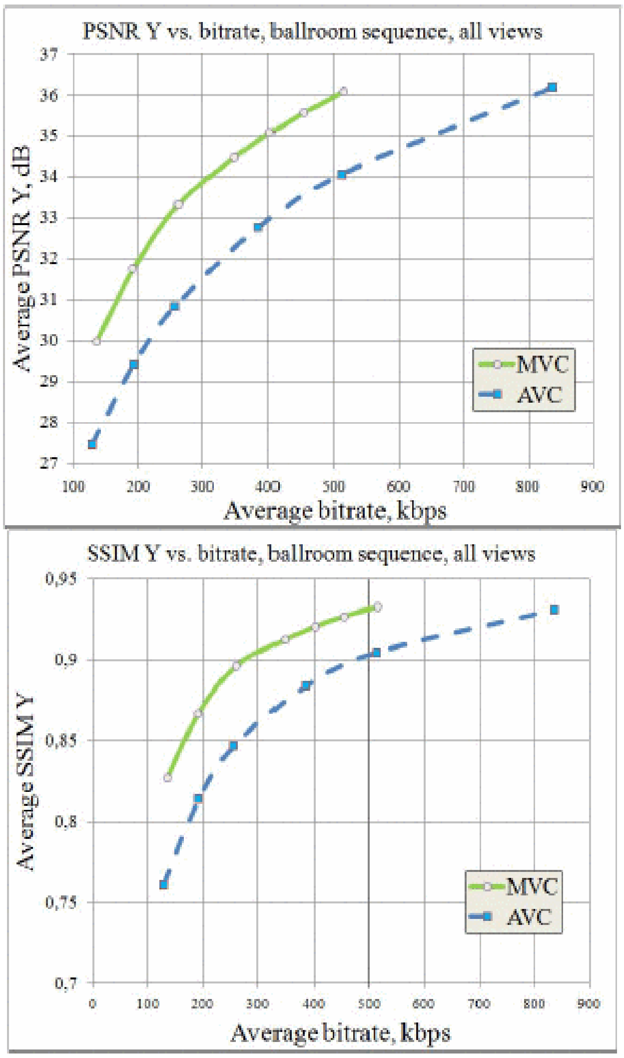


Figure 7 Ballroom sequence coding results (PSNR Y) and (SSIM Y)

Neṁić and colleagues in [9] show that MVC achieved higher compression efficiency at lower bitrates. MVC should be taken into account for services and applications of using limited capacity communication channels such as video for mobile services. They found that (MVC) provided the same quality of the coded signal using approximately 30% (sequence exit) and 40% (ballroom) less bitrate on average. The highest bitrate savings are emphasized in the lower bitrate values.

[10] Micallef and Debono demonstrated the effect that transmission errors have on the current H.264-MVC standard in the wireless environment and the studies on the

quality of transmitted multi-view video when the corrupted packets are not discarded by the underlying protocols of the decoder.

In [10] it is shown that the channel loss mechanisms affect the video which is transmitted over a wireless channel. These may result in bit-stream errors. The sensitivity of the bit-stream errors rises with the compression. The use of predictive coding adds to the effects of errors.

The results showed that:

1. Reference software with corrupted slice decoding: The results of Figure 8 show a large degradation even for small bit error rates. This implies that in the absence of necessary error-resilience tools, the reference software bit-streams are unsuitable for the wireless environment.

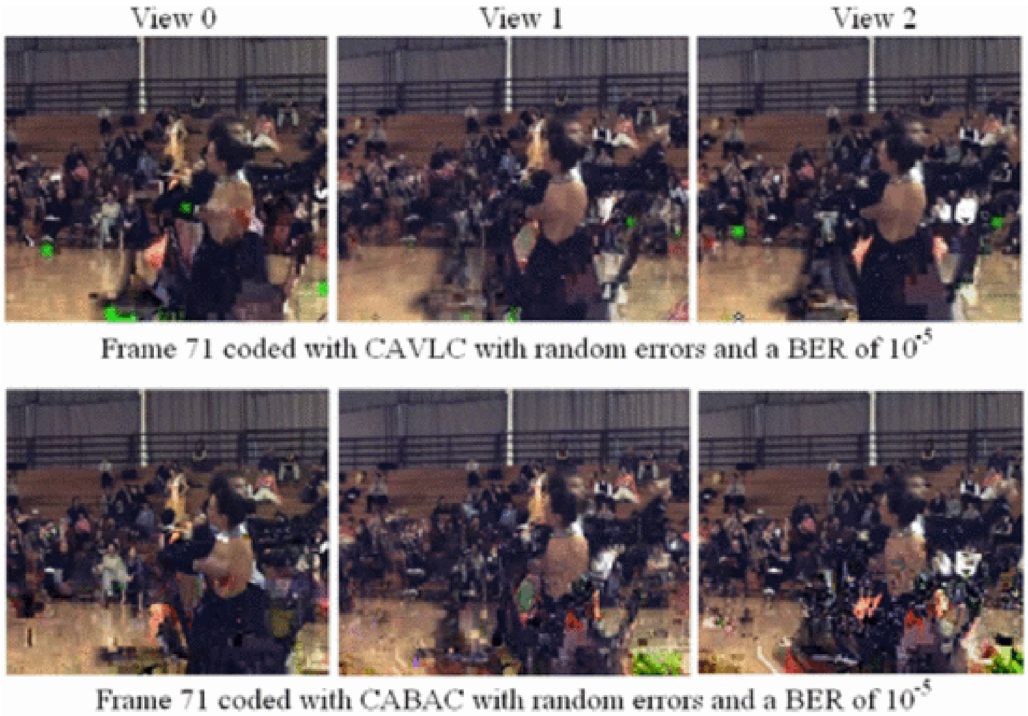


Figure 8 Subjective results for the CAVLC and CABAC entropy encoders

2. JMVM entropy encoders: Figure 9 shows that CAVLC (Context Adaptive Variable Length Coding) is better than CABAC (Context Adaptive Binary Arithmetic Coding) in an entropy encoder for an unprotected bit-stream in the wireless environment.

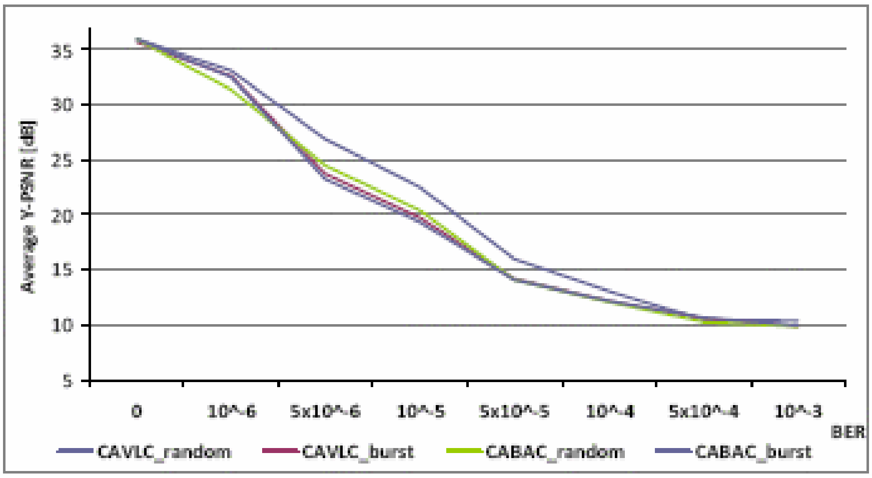


Figure 9 Comparing the entropy encoders with different type of errors

3. Performance with different slice sizes: Table 7 shows that spatial error propagation is lower and the probability of an error in a slice is lower.

Error type	BER	Original	200 bytes	150 bytes	120 bytes
Random	1×10^{-3}	0	0	1	3
	1×10^{-4}	1	4	5	6
	1×10^{-5}	5	8	8	9
	1×10^{-6}	8	10	10	10
Burst	1×10^{-3}	0	0	0	2
	1×10^{-4}	0	3	4	5
	1×10^{-5}	4	8	8	9
	1×10^{-6}	8	10	10	10

Table 7 Mean Score Opinion for Subjective Testing

4. Smaller Cyclic-Intra coded period: Figure 10 illustrates that two other video sequences, “Exit” and “The reconstructed video,” which contain a smaller sequence of degraded images after an artefact occurred, shows faster recovery from the errors.

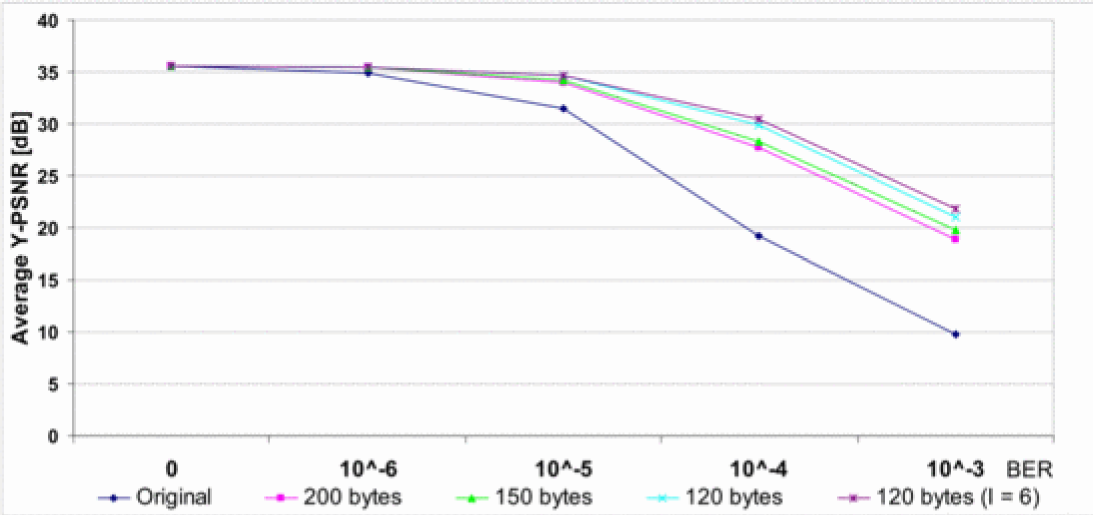


Figure 10 Objective results for the 'Exit' sequence

5. Rate-quality graph: Figure 11 shows the variations in the quality of multi-video sequences with differences in bit-rates. It was found that the quality of the multi-view video sequences improved on transmitting low-quality video and investing the saved bit-rate in error resilience.

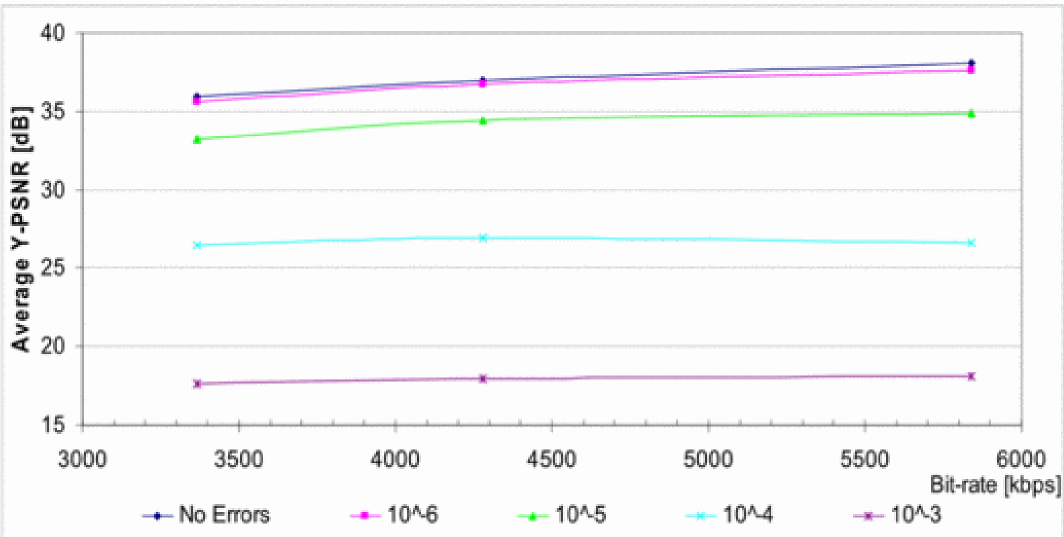


Figure 11 Quality versus bit-rate graph

These results can help the work of our project by understanding the impact of transmission errors in real-time H.264-MVC bit-streams.

In addition, in [11] Ebara, with tiled display wall has constructed the remote communication environment. Also, by transmitting each video image captured by four cameras, Ebara tended to display ultra-resolution video streaming on tiled displays. This was done in order to study the possibility of realistic remote communication. (Figure 12)
The tiled display wall environment consisted of a master node, two display nodes and four LCD panels. The LCDs are an example of tiled display walls located in Formula arrays as demonstrated in Figure 12. The master node and all display nodes are linked by a Gigabit Ethernet network, and the two LCDs are linked to each display node with DVI cables.

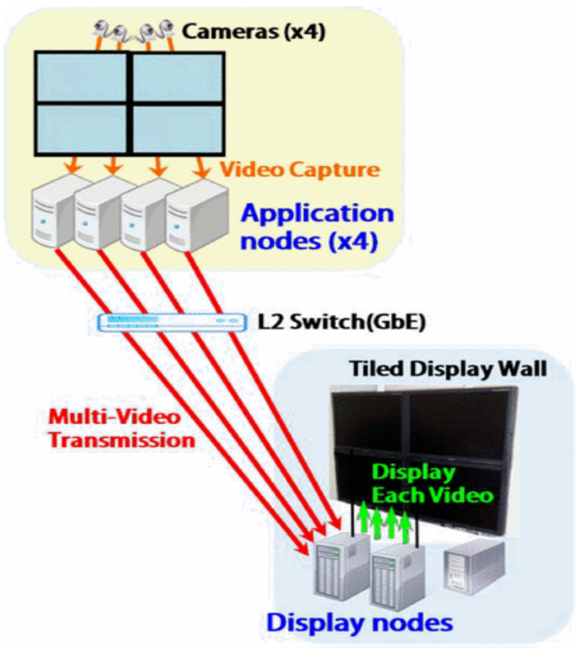


Figure 12 Configuration of implemented environment

In [11] VLC media player is used as an application for video capture and transmission.

Figure 13 shows the result of a video image which was generated by compositing transmission multi-video images. These have been captured by a set of four small cameras. On the tiled display wall, the resolution of the composited video is formula pixels' as the displayed resolution of each transmitted.

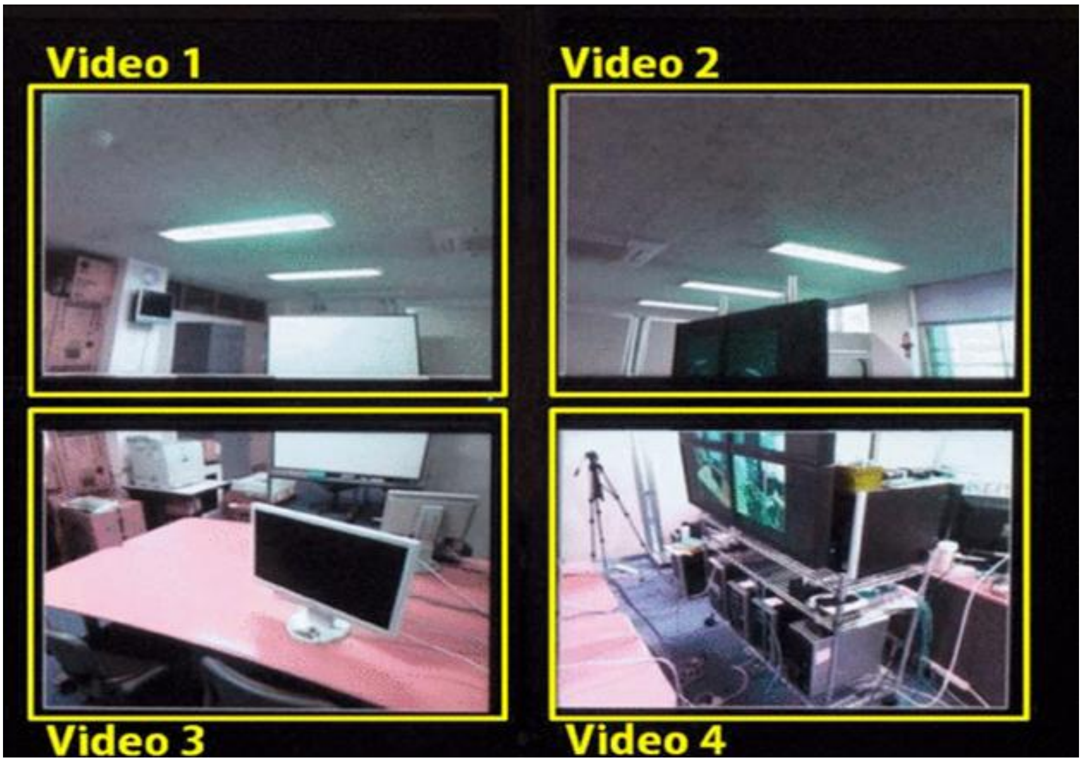


Figure 13 Display results of four transmission video images on a tiled display wall (resolution formula 2,560 × 1,920 pixels)

Figure 14 shows the results of video images captured by a single small camera. On the tiled display wall, the video image is projected by expanding it to the same display resolution as that of video streaming of Figure 13.

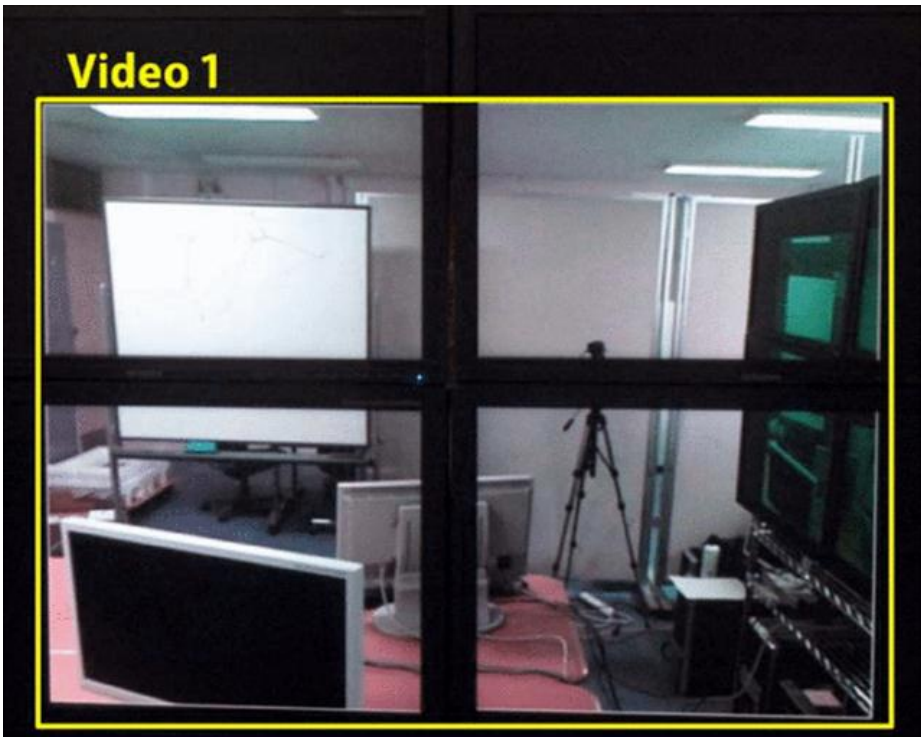


Figure 14 Display results of single transmission video images on a tiled display wall (resolution formula 2,560 × 1,920 pixels)

In [11] it is found that on being captured by a single small camera the video image captured had problems in displaying as video streaming with realistic high-resolution on a tiled display wall. It is difficult to obtain video streaming with a satisfactory image quality since the tiled display wall shows a video image that is grainy. On the other hand, the composited image video with four cameras (as shown in Figure 13) has the capacity to be of higher quality than the expanded video image given by a single camera.
From experimental results, [11] one might obtain a better quality than that achieved in the case of an expanded video using a single camera. Thus we see the feasibility of realistic remote communication by virtue of ultra-resolution streaming on tiled displays through multi-transmission video.

In [12] Arsan holds the view that streaming video applications do not only have a good end-to-end transport performance but also should have a quality of service (QoS) provisioning network infrastructure in addition to a bandwidth adaptive video

streaming architecture. By using two different controllers, it is possible to implement a special architecture for bandwidth adaptive video streaming. These are the Bandwidth Adaptive Streaming (BAS) and QoS controllers.
In the commercial field, we can see the use of several common media streaming architectures. The typical architecture which includes a BAS controller and a QoS controller, consists of a single sender-multiple receivers streaming system as shown in Figure 15 for wired and wireless networks.

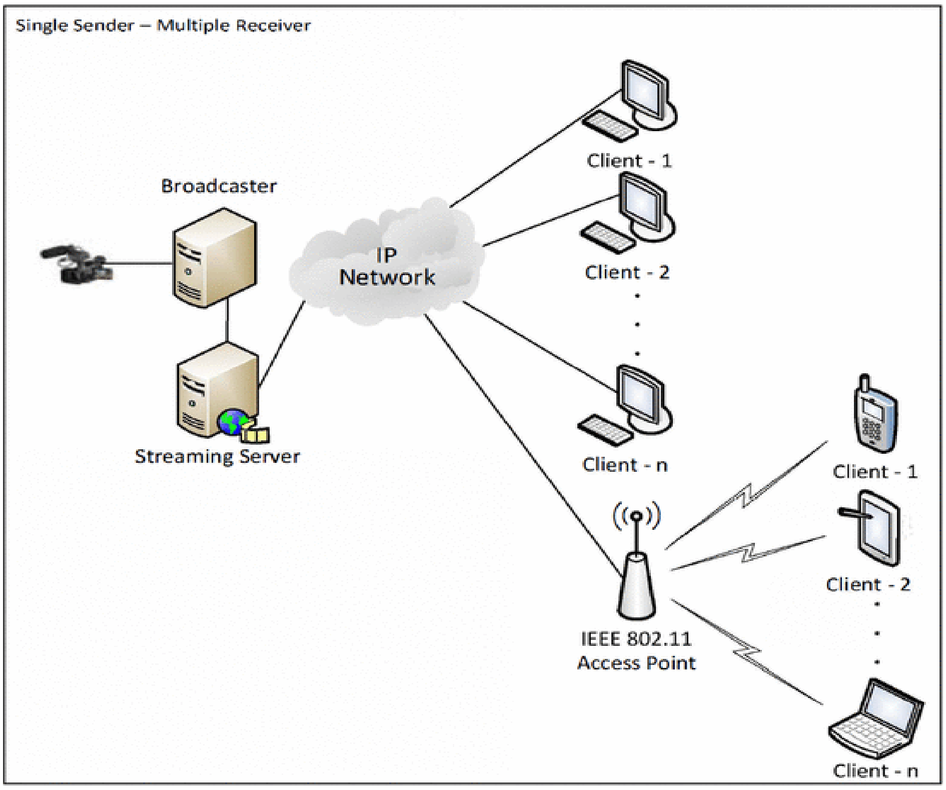


Figure 15 Video streaming for wired and IEEE802.11 networks: single sender-multiple receivers

In [12] it is given a categorization of publicly available bandwidth estimation tools for wired networks as compiling those tools reported to the literature (Table 8).

Tool	Author	Measurement Metric	Methodology
pathchar	Jacobson	Per-hop Capacity	Variable Packet Size
clink	Downey	Per-hop Capacity	Variable Packet Size
pchar	Mah	Per-hop Capacity	Variable Packet Size
bprobe	Carter	End-to-End Capacity	Packet Pairs
nettimer	Lai	End-to-End Capacity	Packet Pairs
pathrate	Dovrolis Prasad	End-to-End Capacity	Packet Pairs & Trains
Sprobe	Saroiu	End-to-End Capacity	Packet Pairs
Cprobe	Carter	End-to-End Available BW	Packet Pairs
Pathload	Jain Dovrolis	End-to-End Available BW	Self-Loading Periodic Streams
IGI	Hu	End-to-End Available BW	Self-Loading Periodic Streams
pathchirp	Ribeiro	End-to-End Available BW	Self-Loading Packet Chirps
Treno	Mathis	Bulk Transfer Capacity	Emulated TCP Throughput
Cap	Allman	Bulk Transfer Capacity	Standardized TCP Throughput
Sting	Savage	Achievable TCP Throughput	TCP Connection
Ttcp	Muuss	Achievable TCP Throughput	TCP Connection
Iperf	NLANR	Achievable TCP Throughput	Parallel TCP Connections
Netperf	NLANR	Achievable TCP Throughput	Parallel TCP Connections

Table 8 Bandwidth Estimation Tools in Wired Networks

Furthermore, for wireless networks, [12] has made another compilation for a categorization of bandwidth estimation tools. (Table 9)

Tool	Author	Measurement Metric	Methodology
Wbest	Claypool	Per-hop Capacity	Packet Pairs & Trains
DietTOPP	Johnsson	Per-hop Capacity	Packet Pairs & Trains
AdhocProbe	Sun	Path Capacity	Packet Pairs & Trains
ProbeGap	Lakshminarayanan	Path Capacity	Packet Pairs & Trains
iBE	Yuan	End-to-End Capacity	Packet Dispersion
IdleGap	Lee	End-to-End Capacity	Link Idle Rate
MBE	Yuan	End-to-End Capacity	Model Based Algorithm

Table 9 Bandwidth Estimation Tools for Wireless

From Arsan's work, using bandwidth estimation tools, the BAS controller tries to obtain the best path, an acceptable rate and an optimized resource allocation. Delay and packet loss values are calculated using pathchirp (Table 8), Wbest and MBE (Table 9), and the QoS controller.

The path is determined by the decision center and client side buffers. These also optimize the rate under the delay and packet loss constraints. In our project, we take into consideration the types of bandwidth estimation tools.

In addition, the quality of the audio-video streaming service on the network Keerom, Papua was determined by Gondokaryono et al. [13] who studied a number of comparisons of some video codecs, audio codecs, audio bit rates and video bit rate. The average capacity in the network Keerom, Papua was 1.5 Mbps. Also in this study MPEG and AC3 audio were chosen because of their characteristics while the video codec was MPEG4 and H.264. Audio bit rates of 64 and 128 kbps were used, while the video bit rate was 64, 128 and 256 kbps.

Gondokaryono et al. took QoS as the quality of audio and video streaming. There was a specific method on the part of each to determine its value. Their methods included the perceived evaluation of speech quality (PESQ) for audio streaming quality and Peak Signal to Noise Ratio (PSNR) for video streaming quality.

At 4 schools located in the Arso district, Keerom and Papua, the Test site/testbed audio-video streaming service was used. This was carried out with the application of the start network topology in which all the other points would be served by one of them as the central node.

The measurements show that average of the PSNR values lie between 10.48 to 18.39dB, which implies the poor quality of the video. Selection of MPEG audio 64kbps and 128kbps MPEG-4 produced PSNR = 16.60dB and PESQ = 3.802. Although the PSNR value is lower than expected (PSNR 20 ≥ dB), subjectively however, a good quality was produced by this configuration of the audio-video streaming. The unbroken video frames attest to this.

When [13] H.264 codec is used at a 64 and 128 bit rate (kbps), the PSNR was approximately 17 – 16 dB; however, when a rate of 256 bits (kbps) was used, the PSNR was about 14dB. Therefore, MPEG 4 yielded/yields a higher bit rate (kbps) in order to increase PSNR, and this will produce a clear picture. The H.264 codec works well at a low bit rate, which reverses to MPEG4.

The video quality control system (VQCS) was proposed by Hyun and Seong [14]. This system controls video service quality by monitoring end-to-end available bandwidth for video streaming services such as IPTV in NGN (Next Generation Network) convergence networks. Video quality degradation such as video jerkiness (description perception of still images in film), block distortion (loss of video quality) and blurring (video disorganized not like the vision in fog) is caused when network available bandwidth is insufficient.

In [14] the authors configured test network to for the just verification of proposed model, and measured the video quality using the MSU video quality measurement tool.

The interworking system controlled between transport QoS parameters and media quality, especially available bandwidth for adding to quality level of the service user.

Figure 16 shows the operating environment for VQCS that is proposed to control video quality using available bandwidth. VQCS measures available bandwidth in real-time at the terminal to consider scalability. The scalability extractor considers various terminal performances and network environments based on the measured available bandwidth.

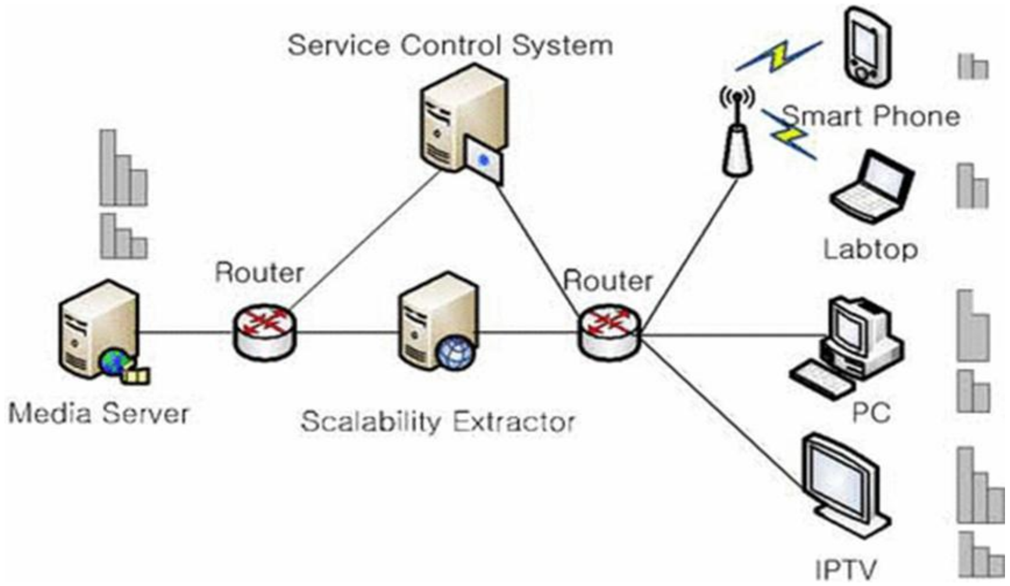


Figure 16 The proposed VQCS operating environment

The scalability extractor can minimize packet loss and delay due to insufficient bandwidth because it selects and forwards the appropriate video streaming data rate to the network environment.

It can also adaptively control the data rate of the stream according to the available network bandwidth.

Constructed test network in [14] is shown in Figure 17 to experiment the proposed VQCS performance. This media server in the test network sent a video stream to the client.

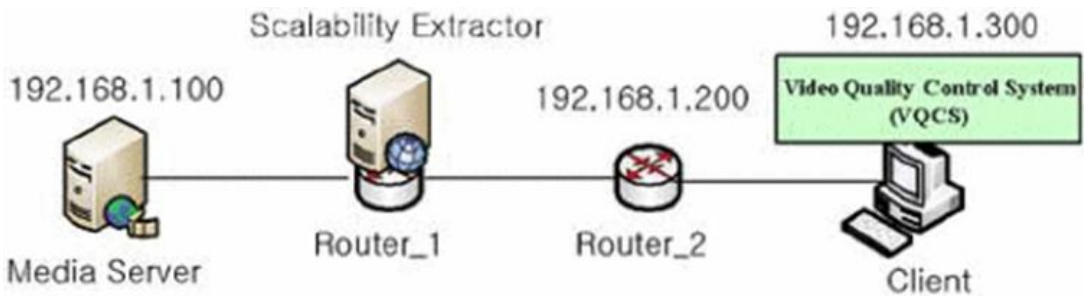


Figure 17 Test network configuration for VQCS performance verification

In their study [14] Linux network emulator commands were used to select the video stream data rate according to network resource (available bandwidth) changes. The following commands can be used to control network delay, jitter, packet loss and bandwidth in the router based on Linux (kerne1 2.6 distribution support).

Authors have selected SSIM (structural similarity) score to accurately measure video quality. VLC media player was used for video streaming with a de-jitter buffer size set to 100ms. VLC media player was also used for video stream delivery in an experimental network set to a certain delay, jitter and loss rate. They compare 300 original images and 300 test images that were delivery through different network environments such as Figure 16 using the full-reference` method.

Figure 18 shows video streaming at 2.5 Mbps with the original video frame sample data rate and a high packet loss rate with 2 Mbps available bandwidth network; serious degradation results, as shown in Figure 18(c). If a video stream at a data rate below 900 kbps is transmitted in the same network environment, the video stream service would improve. This is carried out by preventing packet loss in spite of the image resolution being decreased to 0.962 of SSIM score as shown Figure 18(b).

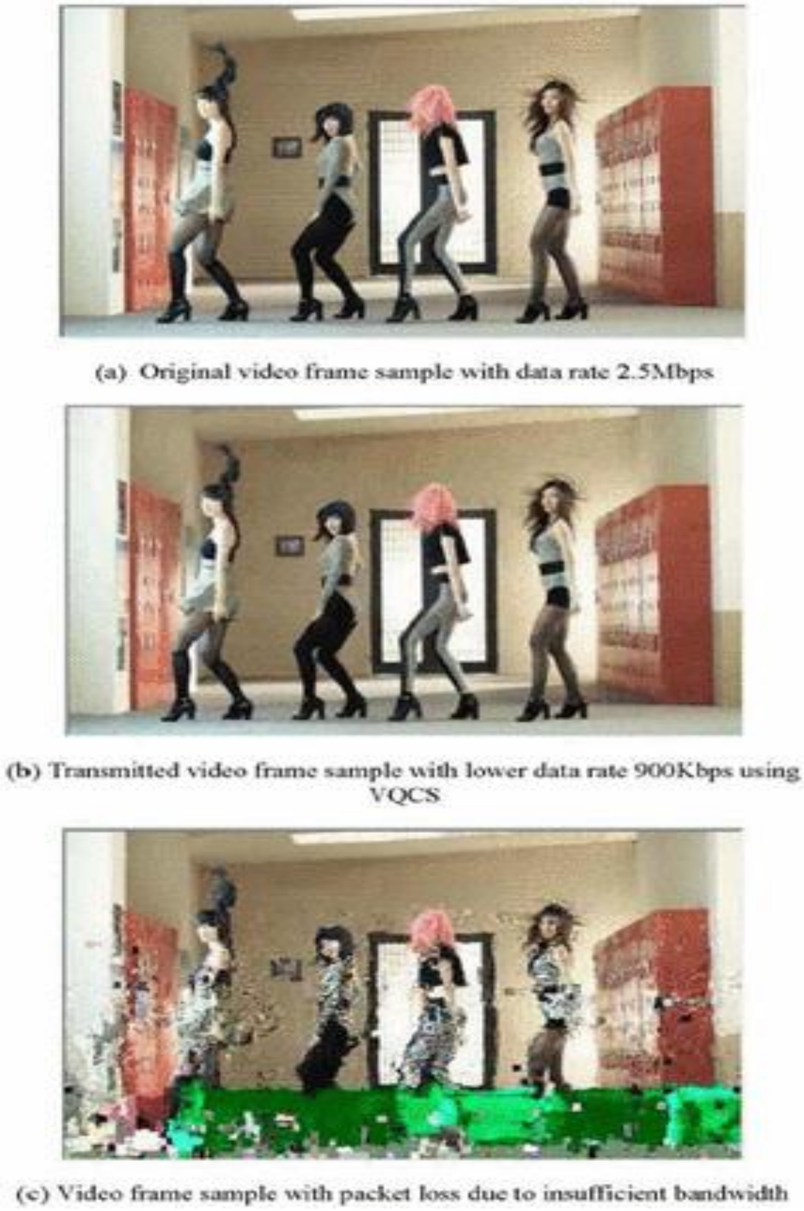
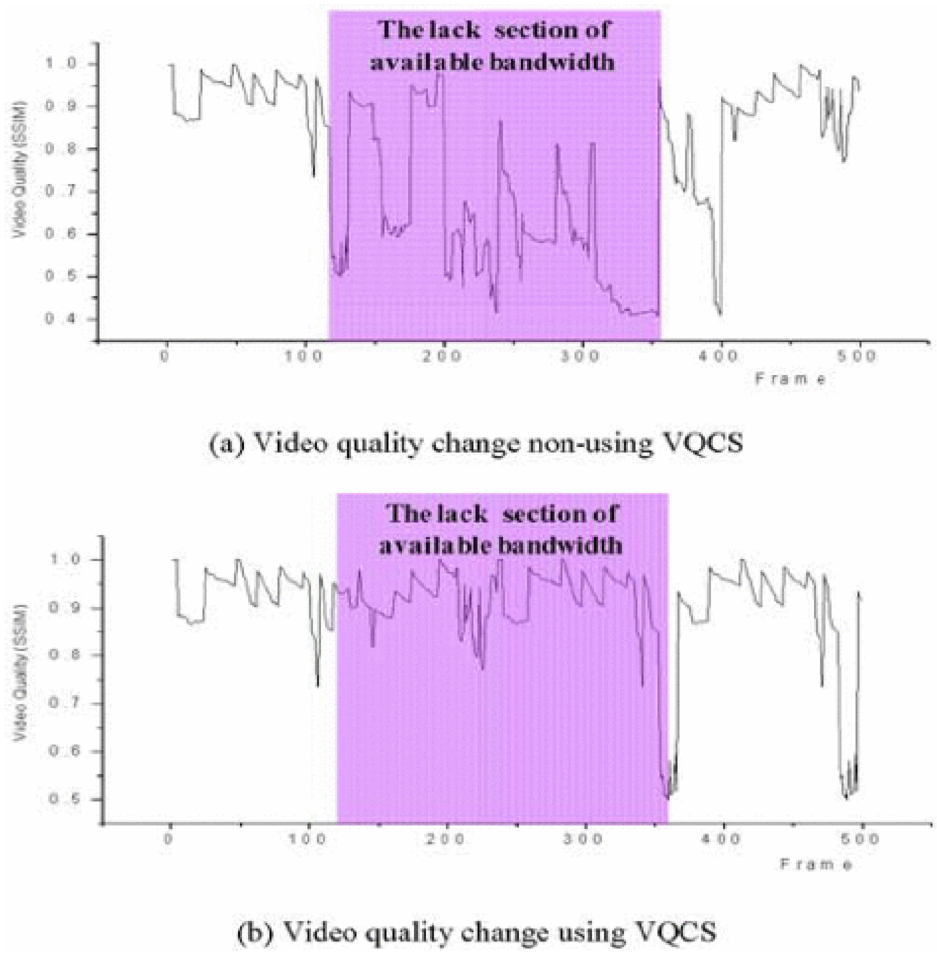


Figure 18 Video frame samples according to network resources and streaming data rate [14].

Figure 19 shows that the average SSIM score is improved from 0.768 to 0.906 when VQCS is used. Therefore, [14] could improve the quality of video service by controlling the stream data rate through real-time available bandwidth monitoring when VQCS is applied at the terminal.



(a) Video quality change non-using VQCS

(b) Video quality change using VQCS

Figure 19 Video quality changes when selecting adjusted streaming data rate with VQCS [14].

The authors have been able to determine the quality of the (VQCS) environment, which is sufficient to add their thoughts to the research in terms of the practical by control video service quality.

In addition, Nakamura et al. [15] show streaming video broadcast services such as *YouTube* and *Ustream* that allow users to watch streams in which they are interested. In the current streaming video broadcast services, users search for streaming videos by title and tag information. However, since many streaming videos provided insufficient information, it was often difficult to find any video. Because many streaming videos have insufficient and incorrect tag information, viewers could waste their time in searching for content. Moreover, since only one streaming video can be watched in one browser, comparing two or more simultaneously streaming videos is difficult.

Due to these problems, [15] studied Multi-Stream Viewer as a simultaneous viewing system for streaming videos with tag information is added to the time axis to enable users to watch streaming videos. Still the same work showed that two delivery methods exist: delivery on demand, which delivers streaming videos saved on servers, and on live streaming, which delivers the images in real time. They implemented viewer interface that enables users to search and watch streaming videos as a web application using Action Script and MXML (an XML-based user interface markup language). A discussion can be done that in [15] the author implementation viewer interface enables users to search and watch streaming videos as a web site; however, the researcher used an on-demand format.

Flash Media Development Server 4.0, which is a real-time media server, was used to deliver streaming videos to two or more platforms, to manage streaming videos and to send them to viewers. The streaming server prepared the client API for these services, and the viewer interface with the API was implemented [15].

Within the streaming server, by dividing and managing the linked streaming videos with separate folders, viewers can receive streaming videos in the folders from the streaming server and are also able to receive two or more linked streaming videos.

With Multi-Stream Viewer, the average time where subjects discovered one streaming video out of 20 streaming videos was about 12 seconds. With *YouTube*, the average time where subjects discovered one streaming video was about 101 seconds.

[15] Concluded how this work can run more than one video within a single interface and on live streaming.

Live televised sports events are tending to become more instrumented; therefore, supporting more camera views were needed. At Berlin 2011, at least 25 cameras were set up in the stadium to enrich the live coverage of the event. An alternative user scenario is that viewers could be allowed to direct their own viewing schedules of live sports events through scheduling the switching of video streams and cameras themselves. Zhenchen Wanghe [16] proposed and evaluated a video stream adaptation scheme which was able to coordinate multiple streams through reducing bandwidth contention (a bottleneck occurs when two or more files are simultaneously transmitted over a single data line. The effect is to slow delivery of each). The main novelty of the system is that dispensing the available bandwidth amongst multiple input video streams can automatically coordinate and preserve the video quality across multiple live video streams. A personalized camera switching feature is also proposed and evaluated, which may be consistently used across users and is scalable as the number of system users increases.

The author faced two challenges: First, the terminal network connection becomes congested when streams are received from several cameras concurrently. Second, end users need an interface that is usable by non-professionals to view live camera feeds in a timely manner.

[16] Found that the video stream adaptation scheme needs to address both a single stream viewing and a multi-stream viewing scenario.

In this paper, an Internet based video streaming player called ePlayer is reported to be able to meet these challenges through researching, developing and evaluating:

1. a video stream adaptation scheme for the adaptation of multi-video streams in terms of video quality and streaming smoothness.
2. Personalization of camera switching.

The system architecture consists of three key components: camera switching video stream adaption, user profile data store and camera switching personalization control (Figure 20).

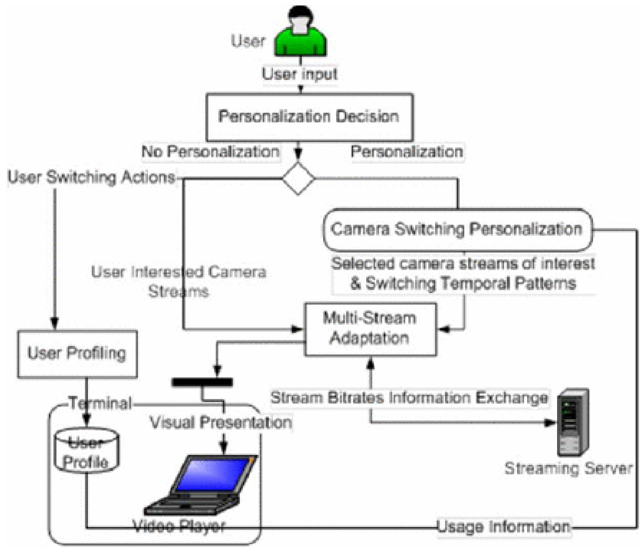


Figure 20 A personalised Camera Switching System

A comparative approach is taken for the evaluation, i.e. a system with multi-stream adaptation versus a system without multi-stream adaptation. The video stream download speed is limited to 2.5Mbps. Prepared video clips were encoded with the following bitrates: 300Kbps, 427Kbps, 608Kbps, 866Kbps, 1.23Mbps, 1.64Mbps and 2.43Mbps. Three copies of the same streams rendered with same video height and width were sequentially streamed by the system.

The results obtained for comparing systems with and without multi-stream adaptation are illustrated in Figures 21 and 22 with the shaded regions highlighting potential bandwidth.

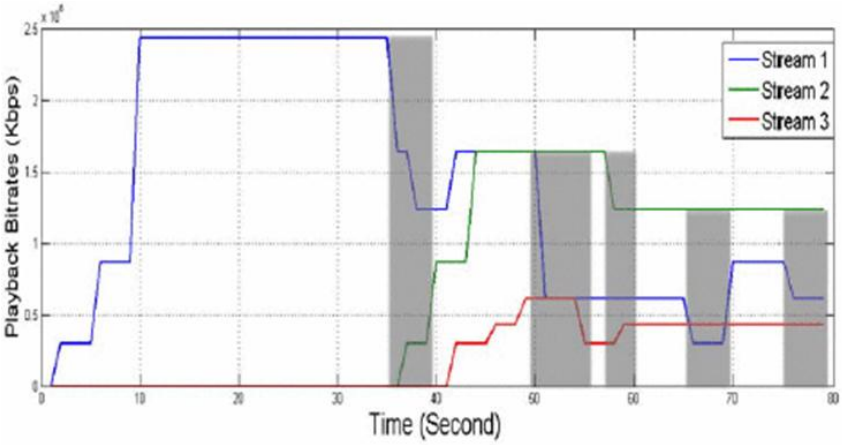


Figure 21 Comparing system results I

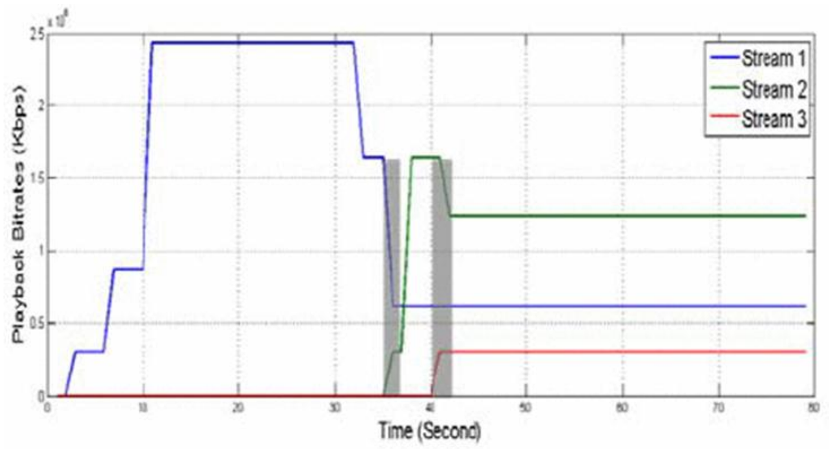


Figure 22 Comparing system results II

Results of this experiment suggest that when the multi-stream adaptation mechanism is used, bandwidth contention becomes less intensive and a more stabilized playback bitrate arises.

The researcher can use this idea to select any camera in future studies.

2.3 Summary

The work presented in [8] leads us to consider UDMVT as a compression process before transmission. For this reason, we aimed to conduct our experiments in this format; however, we could not find the source, hence the algorithm of UDMVT as discussed in the limitation section of the conclusion chapter.

Similarly, [9] present successful analyses of the importance of the MVC codec; however, the researcher did not analyze the detail in the wireless performances. Thus, we take efficiency of MVC into account for the wireless performances in this study.

[10] Shows that they developed a program to simulate the wireless environment, whereas our research will be practical and use a real wireless network.

In addition, in [11], the researcher attempts to analyze the behavior and role of the camera in his work in order to maintain accuracy of the images obtained.

The classifications of bandwidth estimation tools in wire and wireless networks are shown in [12] in details that can be selected in order to measure the QoS of the streaming video.

Additionally, in the literature, some video codecs are compared as in [13]. The results of the experiments reveal the fact that there was a better quality of the video streaming service when the MPEG4 256kbps video codec was used.

Video quality was improved by using the proposed VQCS scheme in [14] because VQCS can adaptively control the streaming data rate according to network resource status in order to prevent packet loss.

As discussed in [15], the authors used a program to build a single interface to watch more than one video.

CHAPTER 3

IMPLEMENTATION

In this chapter, we explain the steps of implementation which followed suit to addressing the research question.

3.1 Implementation Requirements

1. Two web cameras: used to capture video frame which needs to be sent, as shown in Figure 23.

Figure 23 Web camera

2. USB-HUB: is a device that collects several USB ports into one port, as shown in Figure 24.

Figure 24 USB-HUB

3. Primary and secondary servers: we used two servers; the first one was to save the video frames which were captured from the web camera. The second server was

connected by cable to the first server and used to decode video and then be sent to the client wirelessly, as explained in Section 3.3.

4. Ethernet crossover cable: is used to connect two devices of the same type, (such as the Primary and Secondary servers) as shown in Figure 25. Moreover, 10 Gbps Ethernet LAN Adapters were used.

Figure 25 Ethernet crossover cable

5. Client: This may include one or more computers used to receive the video frame from the second server wirelessly, as shown in Figure 26.

Figure 26 Client

6. Software: three software packages were programmed in c# language and these packages were used for primary, secondary servers and client.

3.2 Binding System

1. Connect the cameras to the USB-HUB, as shows in figure 27.

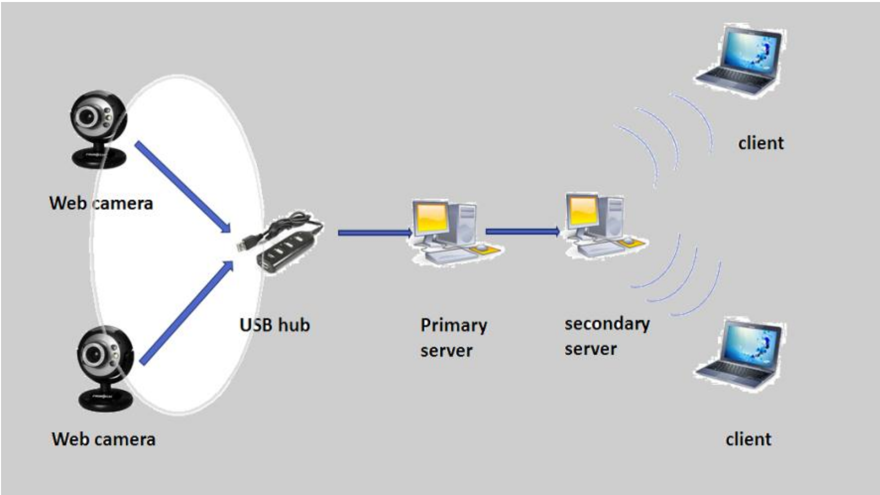


Figure 27 Cameras connected with USB-HUB

2. Connect the USB-HUB to the primary server, as shows in Figure 28.

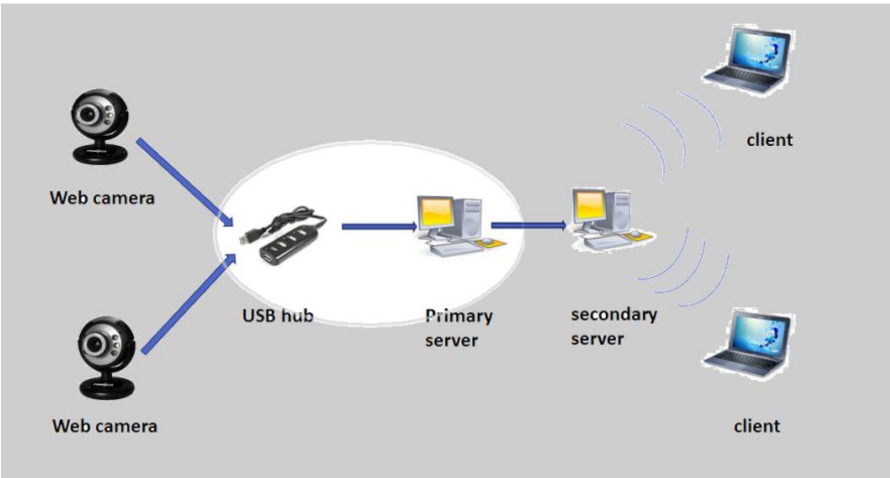


Figure 28 Connecting USB-HUB and primary server

3. Connect the primary server to the secondary servers using the Ethernet crossover cable, as shown in Figure 29.

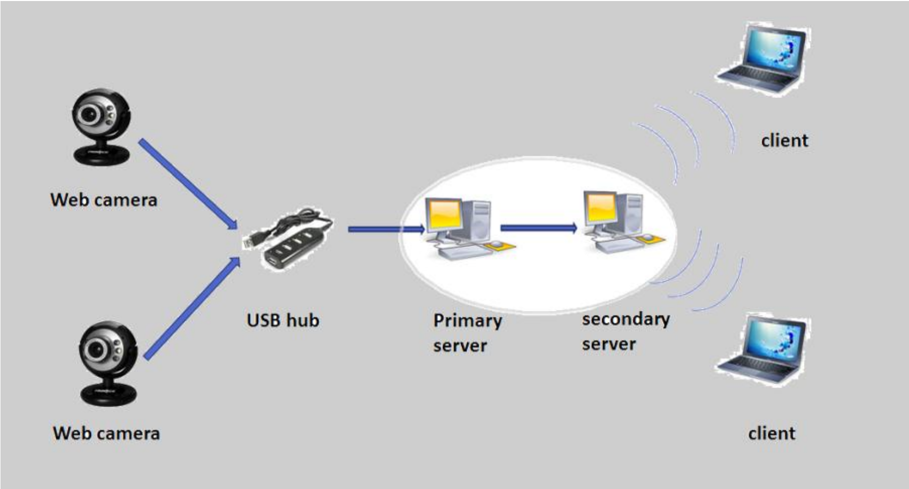


Figure 29 Connecting primary and secondary servers

4. Connect the secondary server to the client via the wireless link, as shown in Figure 30.

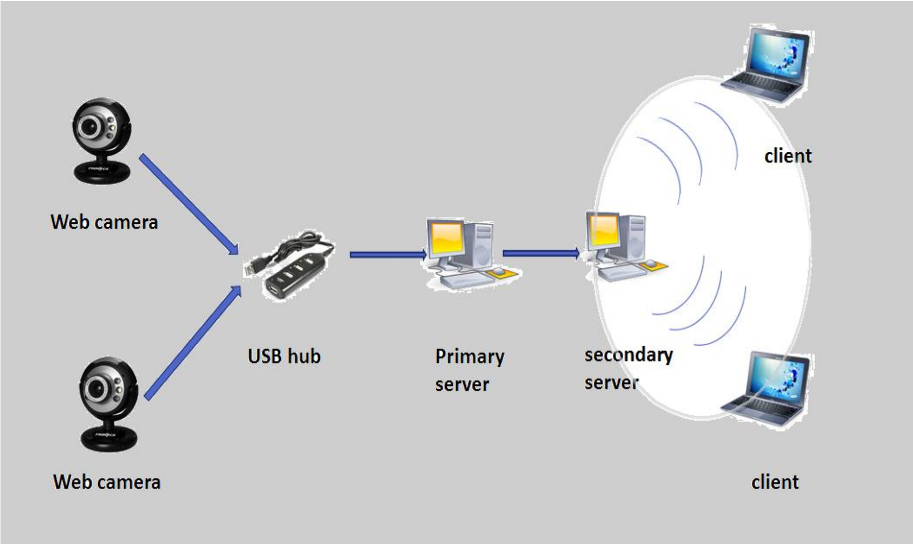


Figure 30 Connecting secondary server and client

5. Installing the software (program coding) and running it on the Primary server, Secondary server and clients.

3.3 Implementation

1. Capture a video with webcams for approximately 1 minute and transfer the video to the primary server.
2. In the second step, we face a problem regarding the size and quality of the video frame, and then by using the software (program coding package) we encode the video frame via MVC (Multi-view Video Coding).
3. By using the Ethernet crossover cable, we can transmit the compressed and uncompressed video from the primary to the secondary servers.
4. Finally the secondary server sends the compressed wirelessly to the client in the MVC and uncompressed form.
5. At the client, we play the video to ascertain whether the video works or not.

3.4 Evaluation

In this section, the researcher evaluates CPU time and video quality results.

1. We can measure the CPU time and utilization in primary, secondary servers, and clients:

	Client / sec	First server /sec	Second server /sec
Without compression	64	62+30=92	64
With compression (MVC)	60	120+1.5=121.5	60

Table 10 CPU Time With Resolution 1280×720 Duration 60 Sec.

	Client	First server	Second server
Without compression	60-65	84-90	60-65
With compression (MVC)	90-93	90-94	60-65

Table 11 CPU Usage With Resolution 1280×720 Duration 60 Sec.

Frame (295) Frame (606) Frame (932)

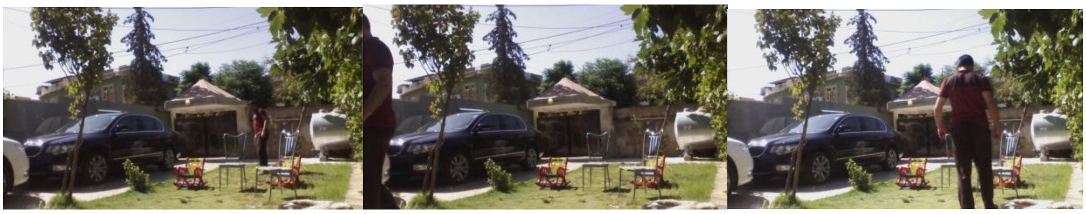
Figure 31 View 1 with resolution 1280x720 without compression

Frame (295) Frame (606) Frame (932)

Figure 32 View 2 with resolution 1280x720 without compression

Frame (295) Frame (606) Frame (932)

Figure 33 View 1 with resolution 1280x720 with compression (MVC)



Frame (295) Frame (606) Frame (932)

Figure 34 View 2 with resolution 1280x720 with compression (MVC)

a. Without Compression

In uncompressed mode, in (Table 10) the first server does not need to bring any effort into its side for using a compression technique; therefore, it only records a 60-

second video from the USB camera and saves the video on the disk and sends it to the second server. The time consumed for recording a 60-second video theoretically will also require 60 seconds. Because we have two views in our system, there will be two USB cameras positioned at different angles. Here, the operation of recording is suggested to be in parallel; that is, the two views are captured simultaneously by the first server. Eventually, the process of capturing two views and saving them on disk at the same time leads to the process becoming delayed slightly more than 60 seconds, which is the origin length. In reality, we have seen that it has lasted about 62 seconds. There is another important point to be mentioned which is the engagement of the first server with the process of sending the two views over the network when it is required by second server. Therefore, when we come into calculating the time required, it shows an additional 30 more seconds. The total time will therefore be 92 seconds. The reason behind the time consumed on transferring files over network is due to disk latency.

The second server needs to read the two saved views from disk and then send them simultaneously and continuously to the client through the network. At this resolution, sometimes the client falls into a standby status for the video packets to be fully received. We have noticed that the time needed for the video to be displayed at the client side from start to end is 64 seconds.

As we can see, the video is not so stable using the uncompressed mode at a resolution of 1280×720.

b. With Compression

In compressed mode, when the first server tries to capture video from two USB cameras, the video needs to be compressed as well. There is therefore an additional effort added to that main operation. The server captures the videos and saves them on disk then compresses them.

So, the time needed to capture, compress, and save the two views by the first server is approximately 120 seconds. We have successfully reduced the size of the two views from more than 2GB for each view to about 200MB as a complete conversion result.

The time required to transfer the small compressed video from the first server to the second server is about one to less than two seconds, so the added value is approximately 1.5 seconds. The total time used by first server is 121.5 seconds.

The second server and the client both work in a real time such that the video is broadcasted from the second server and displayed at the client side in a stable manner and in real time.

To sum up, by using a compressed technique we produce stable and real time video streaming in spite of the fact that the compression has its own load on the server. Nevertheless, we were able to overcome this by using a good distribution of tasks between servers. Moreover, we see that the system without compression eliminates the added load. However, the results are not acceptable due to the instability of the video streaming between the server and clients.

	Client / sec	First server / sec	Second server / sec
Without compression	62	62+18=80	62
With compression (MVC)	60	80+1=81	60

Table 12 CPU Time With Resolution 1024×576 Duration 60 Sec.

	Client	First server	Secant server
Without compression	57-61	81-87	57-61
With compression (MVC)	90-93	84-91	57-61

Table 13 CPU Usage With Resolution 1024×576 Duration 60 Sec.

Frame (295) Frame (606) Frame (932)

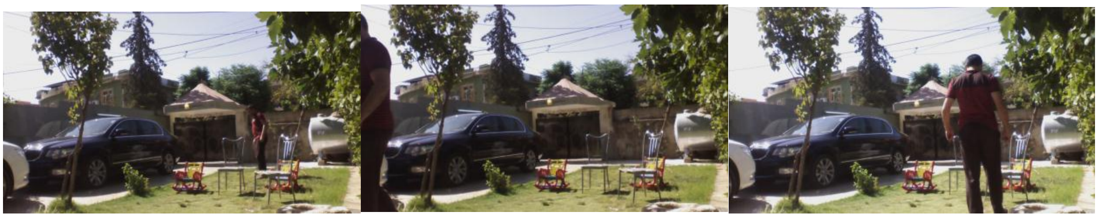
Figure 35 View 1 with resolution 1024x576 without compression

Frame (295) Frame (606) Frame (932)

Figure 36 View 2 with resolution 1024x576 without compression

Frame (295) Frame (606) Frame (932)

Figure 37 View 1 with resolution 1024x576 with compression (MVC)



Frame (295) Frame (606) Frame (932)

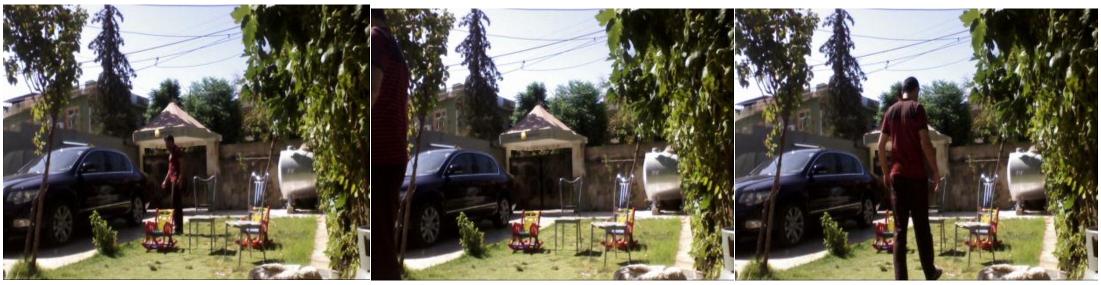
Figure 38 View 2 with resolution 1024x576 with compression (MVC)

	Client / sec	First server / sec	Second server / sec
Without compression	60	61+15=76	60
With compression (MVC)	60	65+0.5=65.5	60

Table 14 CPU Time With Resolution 768×576 Duration 60 Sec.

	Client	First server	Second server
Without compression	55-59	79-83	55-59
With compression (MVC)	90-93	82-84	55-59

Table 15 CPU Usage With Resolution 768×576 Duration 60 Sec.

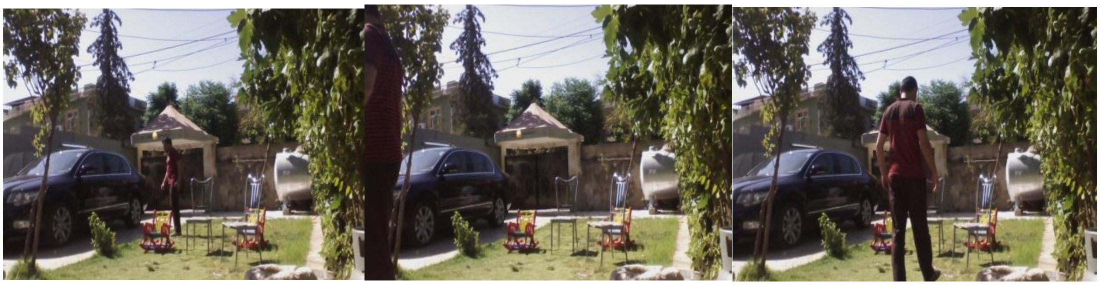


Frame (295) Frame (606) Frame (932)

Figure 39 View 1 with resolution 768x576 without compression

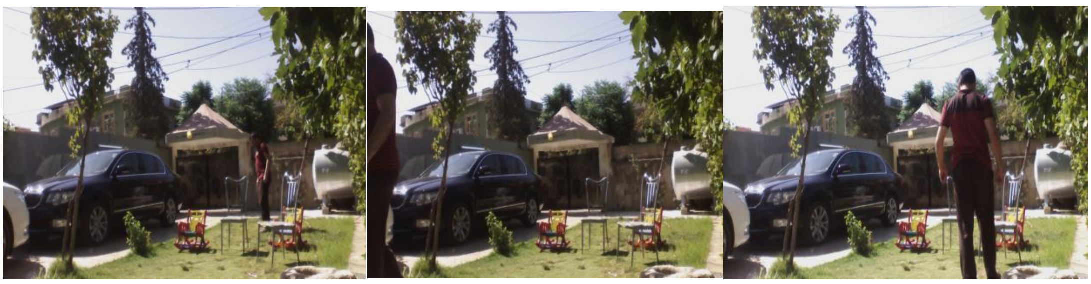
Frame (295) Frame (606) Frame (932)

Figure 40 View 2 with resolution 768x576 without compression



Frame (295) Frame (606) Frame (932)

Figure 41 View 1 with resolution 768x576 with compression (MVC)



Frame (295) Frame (606) Frame (932)

Figure 42 View 2 with resolution 768x576 with compression (MVC)

As seen, the video at resolution 768×576 (Table 14) is acceptable in both modes (uncompressed and compressed). Moreover, in all resolutions discussed so far, we can easily detect that compression mode works in real time and at a stable state compared to the uncompressed mode, which is only acceptable at a resolution of 768×576 and to some extent at a resolution of 1024×576 in spite of the fact that it requires 2 seconds to start in order to fill its buffer with some frames before sending the video.

2. the researcher measures the quality of the video frame in order to the Frame/second:

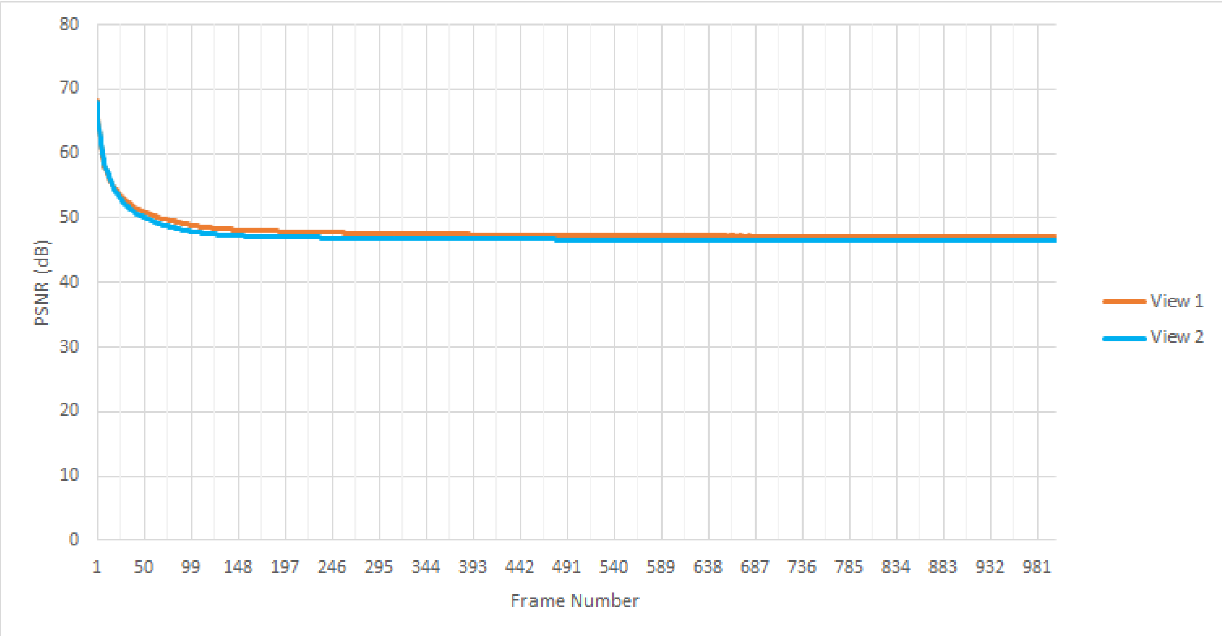


Figure 43 PSNR for view one and view two.

In Figure 43, View 1 and View 2 are compressed versions of a video. To start with providing some analyses of the results of the chart, we see that the difference between the two videos is small at the beginning for one main reason. The video has very small movements at the very early start of its frame sequence, which deemed it highly suitable for good compression of the video at that specific part. The remaining frames come with many movements and actions hence the compression process being highly intense. As a result, we see that there is some of decrease in quality in the two views compared to the original video; however, the reduction in quality is acceptable and even suitable for broadcasting in network environments. Whenever the result of the PSNR is larger, the quality is better, that is, there is a positive correlation between the PSNR value and image quality. On the other hand, we see that View 1 is slightly better than View 2 in quality as this measurement technique attests its results and that returns that the View 2 has different view of the scene that contains more movements than View 2.

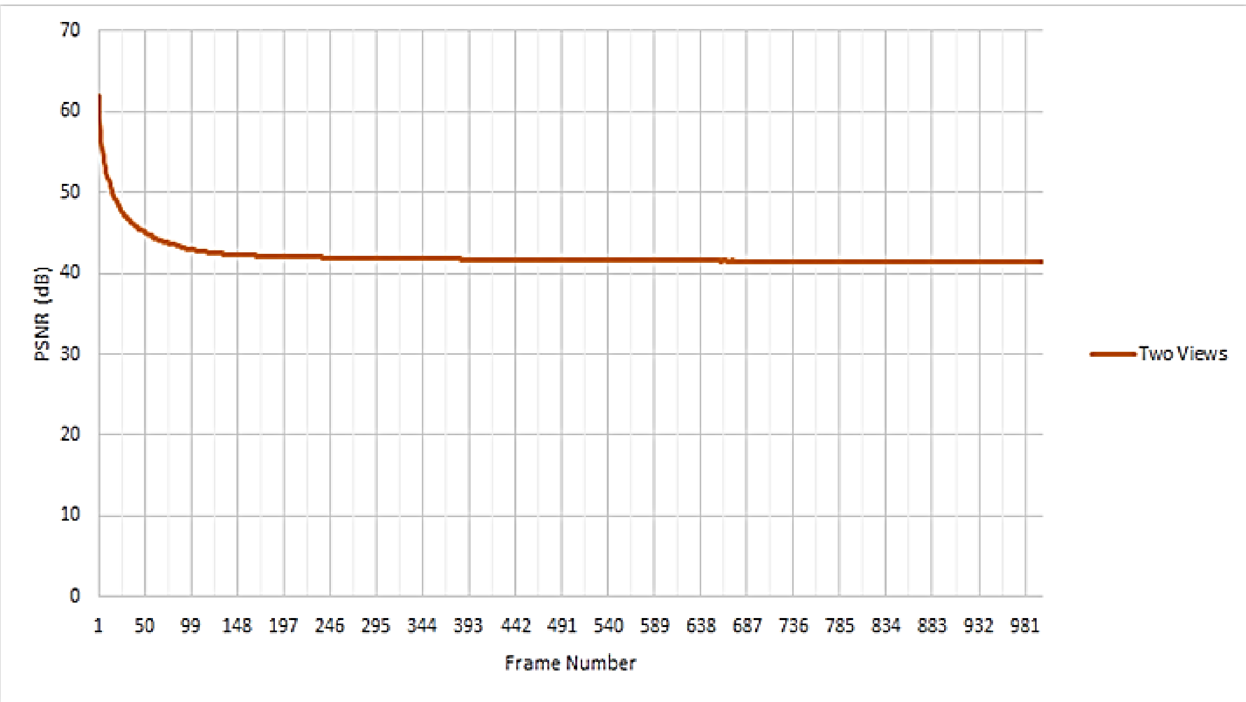


Figure 44 PSNR for a system (two views).

This is the combination of the two views in Figure 3-10. The compression parameters are as follows:

Frame rate = 30

Minimum bitrate= 28 mbit/s ; Maximum bitrate=40 mbit/s

GOP length = 24 and it is of type "OPEN GOP"

Uncompressed file = YUV 4:2:0

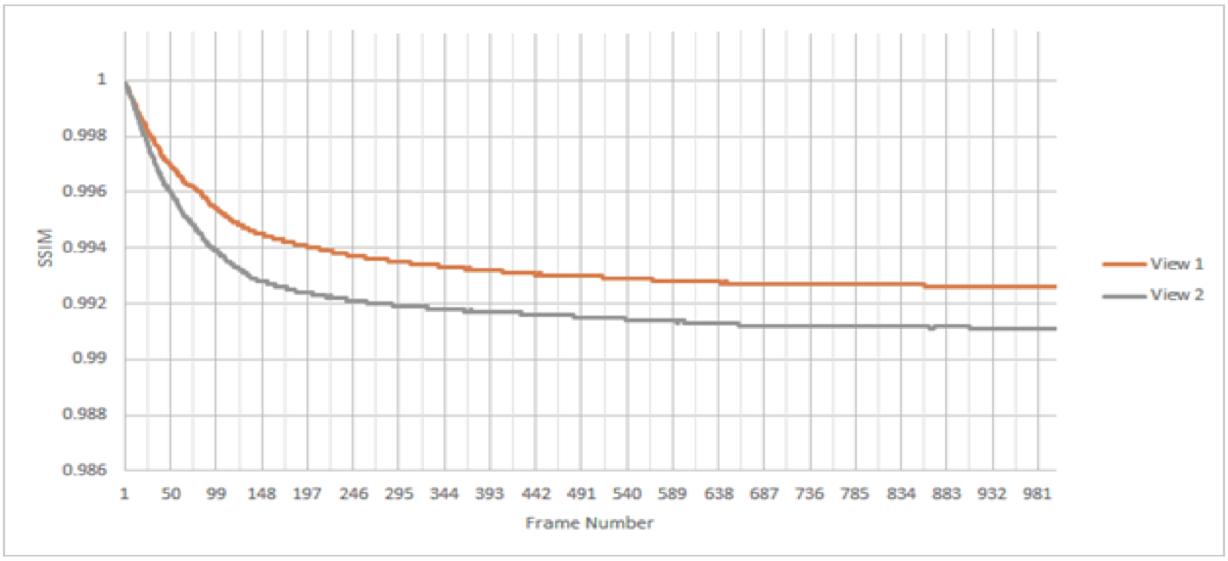


Figure 45 Structural similarity for view one and two.

To read the SSIM values shown in Figure 45, we have to know how those values can be explained in the first place. High values of SSIM are better. Whenever the value approaches number one, the quality is the best possible. In other words, it approaches the original video or image in quality. At the beginning of the image frame sequence, View 1 and View 2 are at best quality (i.e. close in quality to the original frames), the reason behind this is that the contents of the early frames of the video have little movement or action inside them, so the structure of the information of the original and the encoded sequence is almost identical. When the video is in progress and reaches the later frames, the quality reduces gradually due to the existence of more movement and action in that section of the frame sequence; hence we see that the curve changes its direction downward. Furthermore, from the figure we can see that the quality of View 1 is better than View 2 due to the angle at which each view is recorded, such that View 1 has slightly less movement in it than View 2, which leads to more changes in structure of information than View 1.

CHAPTER 4

CONCLUSION

4.1 Findings of Research

As a result of surveying the literature in section 2.3, we have more than one finding:

1. UDMVT gives enhanced results on the client side: we have inferred this finding from viewing the literature. It is reported that removing repeated frames increases video quality and consumes fewer network resources.

2. MVC software provided lower bitrate with average to good quality in multiview video: as mentioned, that particular coded signal uses on average an approximately 30% and 40% lower bitrate in multi-view video, which produced a better quality result than the H.264/AVC software.

3. The quality of transmitted multi-view video in wireless: the CAVLC (Context Adaptive Variable Length Coding) entropy encoder gives better performance in wireless environment simulations.

4. Measuring the QoS of streaming video: there are many bandwidth estimation tools in wired and wireless networks to measure the quality of video.

5. The researcher's successful use of this idea to design a C# project, and to study the difference results.

6. The researcher's system uses a method to control the video streaming speed transferred over a network by controlling its bitrate. This way is more applicable because the size of the target file is known and hence it is considered as suitable, thereby being able to make calculations in advance to acquire an appropriate and

yet more accurate bitrate for video streaming in order to avoid negatively affecting the quality of the video or making the video unstable at the player side, the client. Moreover, the user is able to obtain from this approach the buffer size used to encode the received compressed video since the buffer used will be as small as possible, especially when there is more than one video is received and require simultaneous decoding.

4.2 Limited Experiments

Due to unexpected and extreme changes in the place where our experiments are conducted, it was necessary to limit the number of experiments.

1. Finding (1) suffers such that we could not conduct our own experiments to our satisfaction with what has already been reported in the literature. Therefore, we have taken the results of the surveyed literature as they are presented.

2. Related to finding (6) the sending of encoded video over a network on the second server should need a much shorter time to broadcast it in a compressed case.

4.3 Future Study

Conduct more experiments to enlarge the research question.

In this thesis, we were only able to test two video formats in a limited working environment. A further study of a series of tests to observe and ascertain the effects of different formats would be a complementary study to this research. This future study will overcome the limitation.

It is recommended to make a combination among the proposed techniques in this thesis such as using fiber-optic cables especially at the server side.

4.4 Conclusion

One of the most important points that can be concluded is that an efficient system was proposed and implemented for resource usage evaluation related to wireless video signals broadcasting depending on client/server principles and applying video compression techniques. The evaluation relates to both consumed CPU time and CPU usage. Moreover, another conclusion is related to the ability of working with more than one video broadcasting format.

Coming back to the research question, “Does compressing videos before sending them to the client have considerable benefit?” we can say that the video which is compressed before being sent to clients, have more benefits than the video which is not compressed.

REFERENCES

1. **Forouzan A. B., (2007),** *"Data Communications & Networking"*. Tata McGraw-Hill Education, vol. 4, pp. 3-5, 69-70.

2. **Yādava, S. C., (2009),** *"An Introduction to Client/Server Computing"*, New Age International, pp. 1-2.

3. **Geier, J., (2008),** *"Implementing 802.1 x Security Solutions for Wired and Wireless Networks"*, John Wiley & Sons, pp. 12-14.

4. **Sun L., Mkwawa I. H., Jammeh E., Ifeachor E., (2013),** *"Guide to Voice and Video Over IP"*, Springer, pp. 54-58.

5. **Richardson I. E. (2011),** *"The H. 264 Advanced Video Compression Standard"*, John Wiley & Sons, pp. 25-26.

6. **Metkar S., & Talbar S., (2013),** *"Motion Estimation Techniques for Digital Video Coding"*, Springer, pp. 3-4.

7. **Pan Z., Ikuta Y., Bandai M., & Watanabe T., (2011),** *"User Dependent Scheme for Multi-View Video Transmission"*, In Communications (ICC), IEEE International Conference, pp. 1-5.

8. **Pan Z., Ikuta Y., Bandai M., & Watanabe T., (2011),** *"User Dependent Scheme for Multi-View Video Transmission"*, In Communications (ICC), IEEE International Conference, pp. 1-5.

9. **Neṁić O., Rimac-Drlje S., Vranjeˇs M., (2010),** *"Multiview Video Coding Extension of The H. 264/AVC Standard"*, In Elmer, 2010 Proceedings, IEEE, pp. 73-76.

10. **Micallef B. W., & Debono C. J., (2010),** *"An Analysis on The Effect of Transmission Errors in Real-Time H. 264-MVC Bit-Streams"*, In MELECON 2010-2010 15th IEEE Mediterranean Electrotechnical Conference, pp. 1215-1220.

11. **Ebara Y., (2012),** *"Approaches to Display of Ultra-Resolution Video Streaming by Multi-Transmission on Tiled Display Environment"*, In Network-Based Information Systems (NBiS), 2012 15th International Conference, pp. 540-545.

12. **Arsan T., (2012),** *"Review of Bandwidth Estimation Tools and Application to Bandwidth Adaptive Video Streaming"*, High Capacity Optical Networks and Enabling Technologies (HONET), 2012 9th International Conference on, pp. 152-156.

13. **Gondokaryono Y. S., Bandung Y., Wibowo J. A., Nugraha A. A., Yonathan B., & Ramadhianto D., (2011),** *"Performance Evaluation of Audio-Video Streaming Service in Keerom, Papua Using Integrated Audio-Video Performance Test Tool"*, Telecommunication Systems, Services, and Applications (TSSA), 2011 6th International Conference on, IEEE, pp. 145-148.

14. **Kim H. J., & Choi S. G., (2011),** *"Design of Video Streaming Service Quality Control System Using Available Bandwidth Information"*, In Advanced Communication Technology (ICACT), 2011 13th International Conference on, IEEE, pp. 1540-1544.

15. **Nakamura K., Tanigawa R., Shimada H., & Sato K., (2013),** *"Multi-Stream Viewer: Simultaneous Viewing System for Streaming Videos with The Time Tag"*, In Consumer Communications and Networking Conference (CCNC), 2013 IEEE, pp. 629-632.

16. **Wang Z., (2013),** *"Personalising Multi Video Streams and Camera Views for Live Events"*, In Complex, Intelligent, and Software Intensive Systems (CISIS), 2013 Seventh International Conference on, IEEE, pp. 122-127.

17. **Rao K. R., Kim D. N., & Hwang J. J., (2014),** *"Video Coding Standards and Video Formats"*, In Video coding standards, Springer Netherlands, pp. 37-50.

APPENDICES A

CURRICULUM VITAE

PERSONAL INFORMATION

Surname, Name: Shamaya, Hazar Jalal Yaqo

Nationality: Iraqi (IQ)

Date and Place of Birth: 27.12.1979, Baghdad

Phone: +964 770 344 80 06

Email: hazar.shamaya@hotmail.com

EDUCATION

Degree	**Institution**	**Year of Graduation**
M.Sc	Çankaya University, Mathematics and Computer Science, Ankara, Turkey	2014
B.Sc.	Al-Rafidain University College, Computer Science, Baghdad, Iraq	2001
High School	Zawraa School, Baghdad, Iraq	1997

WORK EXPERIENCE

Year	**Place**	**Position**
2001	Worked as trainer developer	
2006	Federal Board of Supreme Audit (FBSA) of Iraq	Specialist

LANGUAGE SKILLS:

Arabic: Native Speaker

English: Very Good Command

Turkish: Good Command

HOBBIES: Books, Swimming, Travel, Football.