Design, Implementation, and Performance Analysis of In-Home Video based Monitoring System for Patients with Dementia

BY

Santosh Gondi

Submitted to the graduate degree program in Electrical Engineering & Computer Science and the Graduate Faculty of the University of Kansas School of Engineering in partial fulfillment of the requirements for the degree of Master of Science

Thesis Committee:

Dr. James P.G. Sterbenz: Chairperson

Dr. Victor S. Frost

Dr. Bo Luo

Dr. Russ Waitman

Date Defended: Sept 30, 2014

The Thesis Committee for Santosh Gondi certifies
that this is the approved version of the following thesis:

Design, Implementation, and Performance Analysis of In-Home Video based Monitoring System for Patients with Dementia

Committee:

Dr. James P.G. Sterbenz: Chairperson

Dr. Victor S. Frost

Dr. Bo Luo

Dr. Russ Waitman

Date Approved: Sept 30, 2014

Abstract

Dementia is a major public health problem affecting 35 million people in USA. The caregivers of dementia patients experience many types of physical and psychological stress while dealing with disruptive behaviors of dementia patients [1–4]. This will also result in frequent hospitalizations and re-admissions. In this thesis we design, implement, and measure the performance of an advanced video based monitoring system to aide the caregivers in managing the behavioral symptoms of dementia patients. The caregivers will be able to easily capture and share the antecedents, consequences, and the function of behavior, through a video clip, and get the real-time feedback from clinical experts. Overall the system will help in reducing the hospital admission/readmission, improve the quality of life for caregivers, and in general, result in reduced cost of health care systems. The system is developed using Python scripts, open source web frameworks, FFmpeg [5] tool chain, and commercial off-the-shelf IP camera, and mini-PC. WebRTC [6] is used for video based coaching of caregivers. A framework has been developed to evaluate the storage and retrieval latency of video clips to public and private clouds, video streaming performance in LAN and WLAN environments, and WebRTC performance in different types of access networks. The InstaGENI [7], a GENI rack in KU is used as the private cloud infrastructure for the evaluation. OpenSSL [8] utilities are employed for secured transport and storage of captured video clips. We conducted trials on the Google fiber [9] ISP in Kansas city, and compared the performance with other traditional ISPs.

I like to dedicate this work to my wife Deepa and my daughter Shreya for their continuous support and patience.

Acknowledgements

I would like to thank to my committee members, especially my advisors Dr. James P.G. Sterbenz, and Dr. Russ Waitman for their support and guidance. I would also like to thank all ResiliNets group members, and KUMC medical informatics team for their help and kindness. Finally, many thanks to my family and friends, for their constant support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction and Motivation

This chapter presents the motivation, usecases, and proposed solution.

1.1 Motivation

Dementia is a major public health problem in United States, currently affecting 35 million people, and is estimated to affect 115 million people by 2050 [1–4]. The physical and psychological stress experienced by caregivers, while taking care of Dementia patients is also a growing public health problem. This also results in frequent hospitalizations and re-admissions. Identifying antecedents, consequences, and the function of behavior is considered to be very effective in preventing and managing disruptive behaviors in Dementia patients. We have designed and implemented a video based monitoring system to address the following items

- Improvement in the care and treatment of Dementia patients
- Avoidance and reduction of frequent hospital admissions
- Improvement in the quality of life of home caregivers

- Overall efficiency and cost reduction in health care system.

1.2 Usecases

Following are the two use cases which are addressed in this system

- Primary usecase
  - A caregiver establishes the provider relationship, authorizes review by providers, marks the event, transmits the event, receives the feedback, and ends the relationship.
  - A provider establishes the relationship with the patient, receives the review events, consults events to a multidisciplinary team, and sends feedback.
- Secondary usecase
  - A caregiver can establish a real-time continual monitoring and initiate for real-time coaching.
  - A provider can accept or request real-time sessions for monitoring and coaching.

1.3 Proposed Solution

We design and implement an in-home video based monitoring system as an aid to caregivers to help them to easily capture and share the behavioral changes in patients with the clinical experts, and to get timely feedback.

1.4 Contributions

The contributions of this thesis are:

- System Design

  Designed VMS (Video Management Software) to handle the continuous recording by using commercial-of-the-shelf IP camera and local storage device. Designed browser-based applications for managing the caregiver and provider relationship, and for capturing behavior changes and sharing with providers. HTTPS/TLS based secured transport protocols are employed for transferring captured clips between public and private cloud storage.

- Implementation

  The framework and applications are developed using well-known Python frameworks and scripts. FFMpeg [5] tool chain is used for video recording and for creating clips.

- Measurements and Analysis

  Evaluated the system performance by measuring, real-time recording performance, public/private storage and retrieval latency, bulk data transfer latency to private cloud, and basic WebRTC parameters.

1.5 Organization

The remaining sections are organized as follows: In chapter 2 we explain the related work. Chapter 3 covers the system architecture, design, and implementation. In chapter 4 we explain the measurement methodology and analysis of results. Finally, in chapter 5 we discuss the conclusions and future work.

Chapter 2

Background and Related work

In this chapter, background and related work of thesis is presented.

2.1 Background

A video capture and personal electronics health records platform has been developed [12] to help parents of autism children to capture behavioral changes in them, and to communicate it electronically with health care providers. Autism is a most prevalent and fastest growing development disorder among children in the USA. Apprehending the behavioral changes is considered to be effective in detecting and managing the autism disorders. This study has shown that, the use of technology in capturing and managing the behavioral disorder has been favorably received by parents, providers, and teachers. It has been perceived to be equally applicable in home and hospital environment. The platform adequately addresses the privacy, security, and the control requirements associated with US healthcare systems. This project has been precursor in perceiving the idea of the current work presented in this thesis. The focus of the current thesis is managing the behavioral changes in dementia patents with advanced video, and networking technologies.

The importance of Telehealth systems is highlighted while presenting the findings of a project for managing the autism symptoms and behavior by using a new video capture technology [13]. The general use of video capturing and electronic personal health records is not only useful in reducing the the time and cost involved in visiting the physicians in person, but also, in cases it helps diagnoses and treatment in times of disasters, such as. hurricanes, when actual visit to doctors is impossible. Most of times the behavioral changes are context based and may not show up in children while visiting the doctors. So, capturing the behavioral changes in natural environments, such as, home and schools becomes necessary for accurate treatment and diagnosis. Practically, Telehealth systems reduce the cost of diagnosis, at the same time, potentially increasing the accuracy of diagnosis. The current thesis builds on the idea of this system, by providing similar tools for managing the dementia in home and hospital settings.

Countries across the world are experiencing the growing aging population which will put stress on the current elderly care systems. An experience with a system called smart home technology is presented is presented in [14]. This system is developed to monitor elderly people, using motion sensors, video sensors, and bed sensor to capture the sleep restlessness, pulse, and respiration levels. These sensor data is used to algorithmically detect the changing patterns in physical and physiological activities. The cues from such algorithm can be used to predict the conditions that may commence in future. They aim to reduce the functionality reduction in elderly so that they can perform the activities independently with out much help, and lead better quality of life.

The department of Veterans Affairs (VA), one of the the largest health system operators, increasing makes use of Internet and online personal health record system called My HealtheVet [15]. The study reveals that 71% of participating veterans used Internet and about fifth used My HealtheVet. In conclusion, majority show willingness to use the Internet for health services and need some training and and guidance to effectively use specific system. In this thesis, a specialized system is developed to aid the caregivers in managing the behavioral symptoms in dementia patients through advanced video and networking technologies. High speed Internet is used for transferring the the captured video clip, and WebRTC is used for two-way video based coaching.

Cloud platforms have been explored for implementing a centralized VMS system [16]. Various cloud platforms, such as, SaaS (software as service), PaaS (platform as service) and IaaS (infrastructure as service), and two commercial cloud providers, Windows Azure, and Amazon, along with internal infrastructure is considered for comparison. Based on the current state-of-art IP camera and bandwidth requirements, the cloud solution for VMS is feasible, but very expensive compared to the internal infrastructure based solution. Cloud based servers have been used for video surveillance [17], for efficient large scale storage, considering, privacy, reliability, and fault tolerance. Feasibility of running complex event detection algorithms in cloud on encrypted content is highlighted. In this thesis, public and private cloud providers are evaluated in terms of latency in secured storage and retrieval of captured video clips. Also, latency in moving large quantities of data from home to private cloud for complex algorithmic analysis in different ISP connections is measured.

A federated cloud model has been used [18] to select multiple cloud destinations based on the cost, probability of outage, and reliability requirements. A system called Cloud4Home [19] is unveiled, which combines the capabilities of in-home devices with data center resources to build a low latency, scalable, and accessible cloud computing platform. It utilizes visualization to achieve location agnostic storage, access, and sharing services. A data management framework is presented [20], which is suitable for distributed bio-medical research environments. Bio-medical datasets are often large and may contain large number of of small objects making it challenging to handle in the distributed environment. Firewalls often present hurdle to accessibility to data at the edges of the network. They have proposed a data management framework which is suitable for distributed bio-medical research in the context of FBRIN (a distributed data-sharing system for medical researchers). The current thesis has overlapping aspects with above related work in this paragraph. A local storage device is used to store the private and sensitive data at home, and, on demand basis transferred to the cloud premises for providing access to healthcare providers. GridFTP based services are utilized to transfer multiple hours of recorded video to private cloud for further analysis.

The performance of storage and retrieval of MER (mars exploration rover) data to public cloud storage is evaluated [21]. Cloud solutions provide high availability, geographical redundancy, durability, and fine grained access controls. With archiving and encryption, 70MB/s performance is achieved using cloud option, which is higher than traditional backup strategies using external hard drive and DVD archiving. The present work utilizes some aspects of cloud based backup strategies.

The evolution of video quality measurement techniques and their current state of the art is reviewed [22]. Various subjective (MOS, DSCQS, DSIS, SSCQE, ACR), and objective (PSNR, MSE, BER, PLR) metrics, and how they are employed is described. Also, V-Factor, a hybrid metric using both transport and bit-stream information is presented. Though many approaches and improvements are proposed, there is no metrics for measuring video quality which can be used universally. In our current work we have developed a use-case specific metrics to measure the quality of recorded video clips.

A study of access link performance using data from 4000 gateway devices, across 8 ISPs, from over 4200 devices is presented [23]. The study reveals that, there are many factors contributing to the performance seen at end device, including, the modems in home, and traffic shaping measures used by different ISPs. So, it is difficult the universally compare different ISPs for performance. A white paper from SamKnows [24], describes the methodology for collecting the metrics for different types of services over Internet. In the current work we have used the ideas from above papers to decide the metrics suitable for our application.

2.2 WebRTC

WebRTC is a free open source project. It enables real-time communication between browsers without requiring to install any plug-ins. Figure shows the high level architecture of WebRTC. WebRTC has two distinct API layers, a Web API, which exposes real-time communication features to Javascript based applications in the browser, and a Native C++ API, which is used by browser developers to manage the underlying audio, video, and network components.

GCC (google congestion control) algorithm is evaluated in an experimental

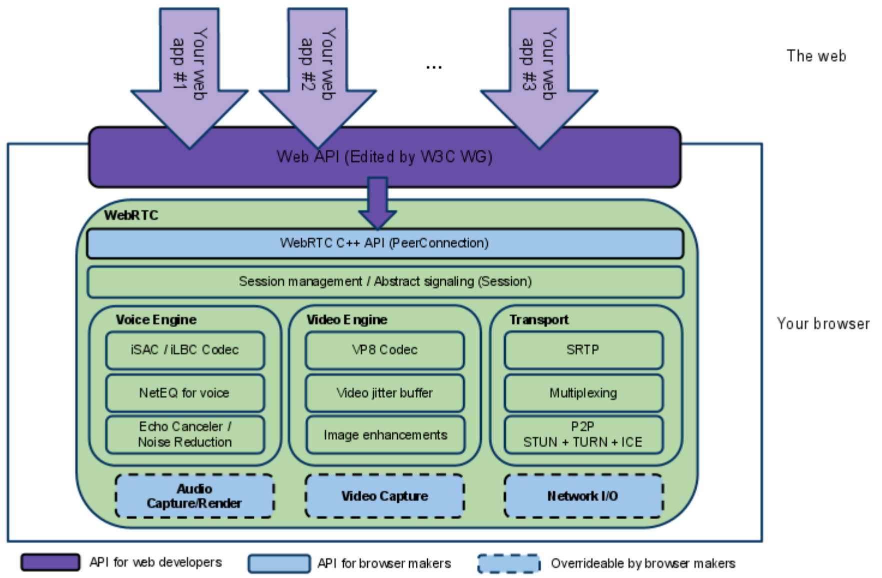

Figure 2.1. WebRTC architecture. adapted from [6]

testbed [25]. GCC is part of IETF RTCWeb WG proposed standards for transport of real-time flows over Internet and is used in WebRTC, an open source real-time communication protocol for web browsers. GCC performs well without any cross traffic. But, friendliness with TCP and fairness with other GCC flows is poor on low bandwidth paths. The friendliness and fairness improves with the increase in the available bandwidth. In the current work, WebRTC is employed for two-way video coaching of caregivers. The performance on different access networks, namely, 4G(LTE), Wi-Fi(802.11n), and wired LAN is measured.

2.3 Data Transfer Protocols

FTP is a standard data transfer protocol on the Internet. It has separate control and data channels for exchanging command and data. GridFTP [26] is a high-performance, secure, reliable protocol, optimized for high bandwidth networks. GridFTP extends the standard FTP protocol by implementing additional features, such as, pipelining, parallelism, concurrent transfers, etc.

2.4 Cloud Storage Options

Generally, public cloud storage option has less upfront cost, high accessibility, high resilience to hardware failures, and geographic redundancy. Also, with public cloud we have less control over the saved objects, which in turn might lead to privacy and security concerns. Performance wise, public clouds have higher latency for storage and retrieval, compared to private cloud storage. Table 2.1 shows the comparison of public and private cloud storage options. We have considered Google storage [27] and Amazon S3 [28] as public cloud storage providers. Google storage provides two options, standard and DRA [29] (durable reduced availability). Compared to standard, DRA is low cost option with tradeoff of reduced availability. InstaGENI rack is used as private cloud storage provider.

Table 2.1. Public cloud vs Private cloud

Public	Private
Low up-front cost Higher access latency High resilience to failures High risk to privacy and security Easily accessible from any place on any device Easily maintainable and upgradable	Higher up-front cost Lower access latency Less resilient Inherently private and more secure

2.5 Firewall/NAT Traversal

Many middle boxes on the Internet, such as, NAT [30], Firewalls, and ALG (application level gateways), will block the incoming TCP [31] connections into local network. Moreover, some middle boxes will block the whole UDP traffic, practically making it impossible to achieve UDP based direct streaming. Even if we are able to achieve this streaming in some cases through a mechanism called hole punching [32], the continuous streaming of recorded content consumes constant bandwidth, creating steady cross-traffic for other network applications in LAN. For example, a 1280X780 resolution video consumes roughly 3 Mb/s bandwidth. So, it will constantly consume 30% on a 10 Mb/s upload connection.

2.6 Dropcam

Dropcam [33], a popular cloud based video monitoring solution, continuously streams to the cloud consuming constant bandwidth. It might waste lot of bandwidth if the recorded clip is never viewed. In our solution the recording is local, and the video clip is shared only based on the need. This way, there is no unnecessary wasted bandwidth. At the same time, the user video is stored locally, providing more privacy. Generally the video quality achieved locally is better than what is possible through cloud streaming. Because, the WAN bandwidth fluctuation is not present in the local setup. Also, on low upload bandwidth connections (which is very typical in DSL based broadband), the dropcam solution is not feasible, whereas, our solution will work by trickling the clip through slow connection.

2.7 InstaGENI rack

GENI [10], Global Environment for Network Innovation, is a NSF funded international testbed for conducting large-scale networking experiments. It is a federated infrastructure, providing programability at each network layer, and, with programable node cluster at each participating institute. GENI racks are a network of distributed clusters supporting GENI aggregate manager API. Figure 2.2 to shows the architecture of GENI rack.

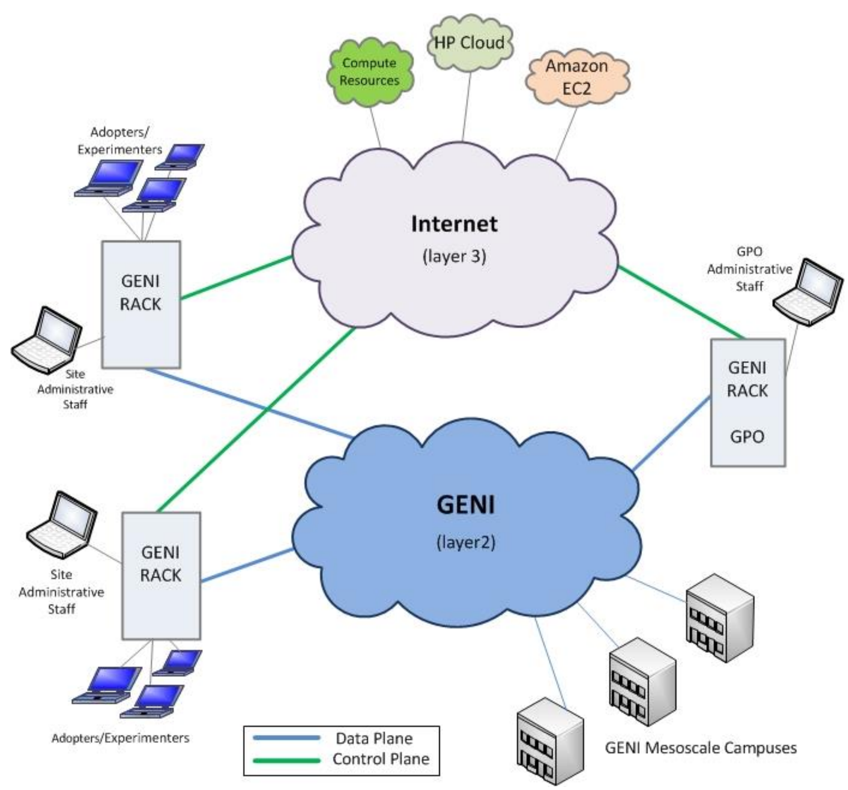

Figure 2.2. GENI rack. adapted from [10]

InstaGENI rack is a type of GENI rack, which is a lightweight, expandable, and standalone cluster. The InstaGENI rack at KU is part of the national GENI testbed. The InstaGENI rack connections are shown in the figure 2.3.

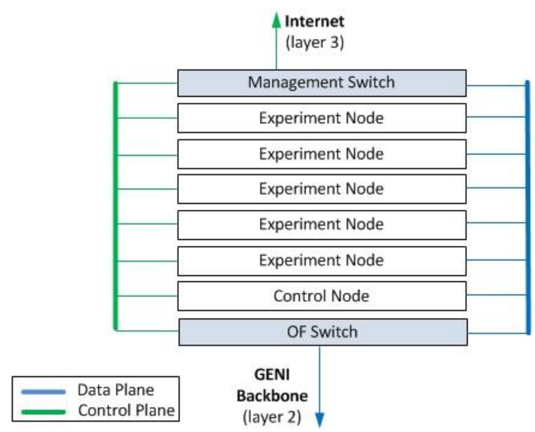

Figure 2.3. InstaGENI rack connections. adapted from [10]

Chapter 3

Design and Implementation

In this chapter, system architecture and implementation details are presented.

3.1 System Architecture

The Figure 3.1 shows the high level system architecture in Google fiber environment. We have COTS(commercial off-the-shelf) high resolution IP cameras continuously recording the patient behavior at home into a local storage device(shown in the figure as Google fiber network box). The caregiver can trigger a capture event which will create a video clip of required duration in local device. The caregiver will then choose to transfer the created clip to public or private cloud storage, for easy accessibility by clinical experts. Caregivers will establish prior relationship with clinical experts for consultation through telehousecalls (Section 3.3) application. The clinical experts will then review the video clip and provide feedback, either through a phone call or an email. Also, the clinical expert or the caregiver at any time can request and schedule a real coaching, to effectively manage the behavioral symptoms in the patient. In this thesis we model the publicly accessible IntaGENIrack as private cloud storage. Google storage and amazon S3 are used as public cloud storage platforms for evaluation of trade-off between public and private cloud storage.

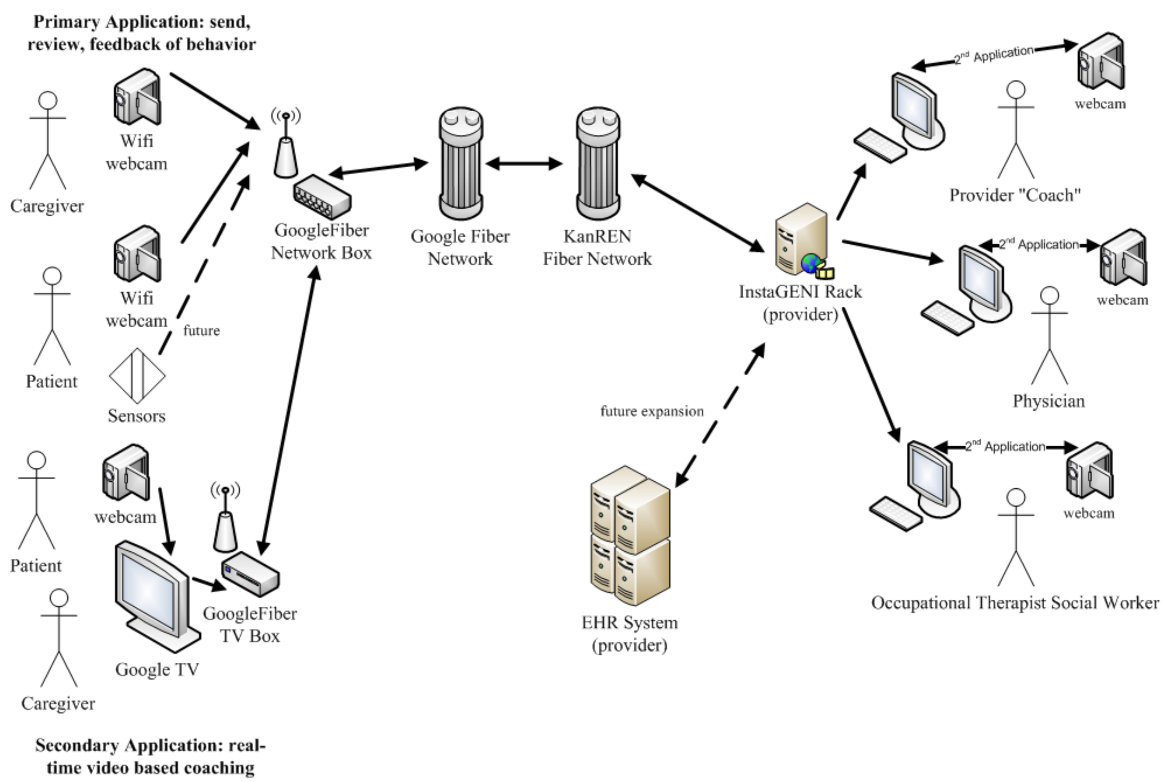

Figure 3.1. System architecture. adapted from [11]

3.2 Detailed Design

Figure 3.7 shows the internal details of each component in the system. We have used D-Link DCS-2132L IP camera for continuous recording purpose. A ZOTAC ZBOX mini-PC with Ubuntu 12.04 is used as local storage and compute

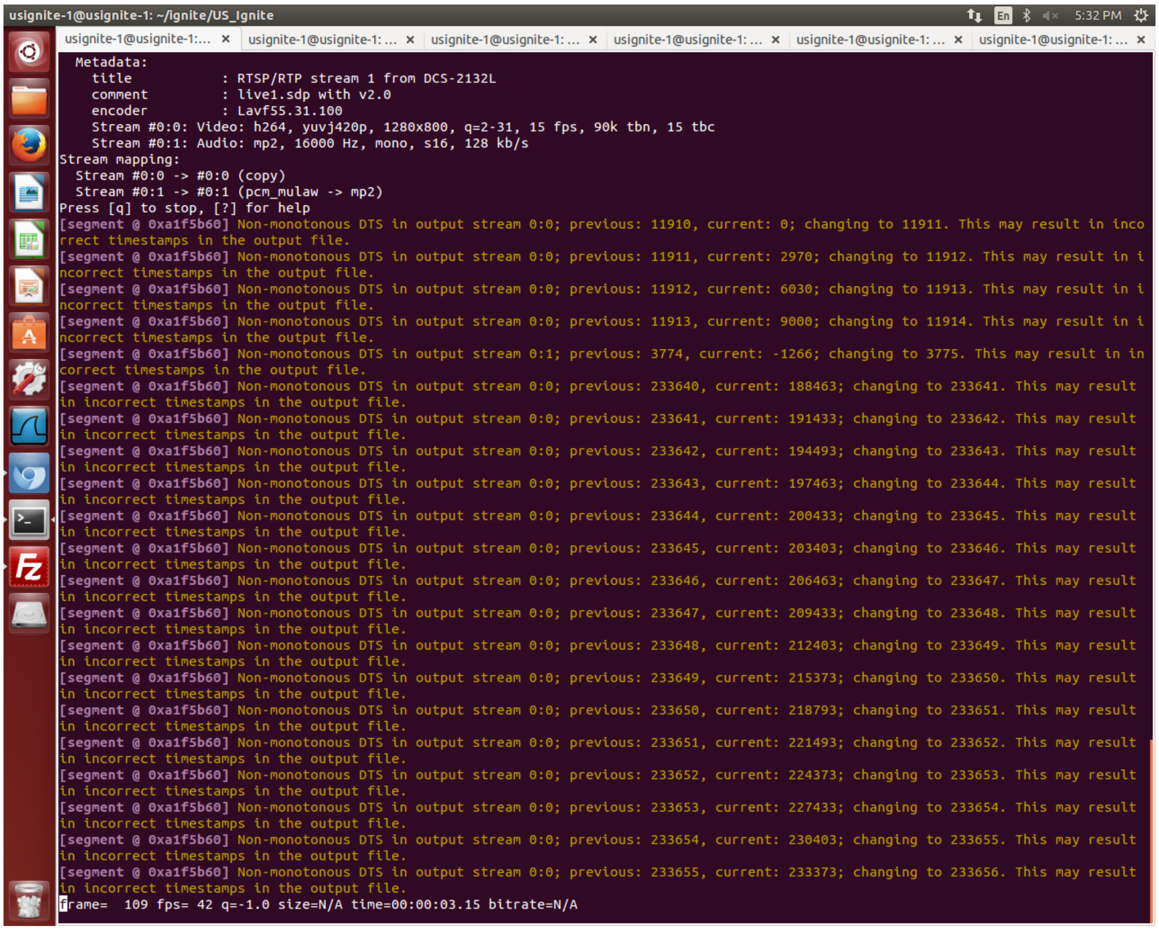

Figure 3.2. Recording snapshot

device. An OpenSSH client was installed on the mini-PC for remote login and configuration of recording and capturing application. During first installation, the camera IP address is detected by a D-LINK application, and the IP address is used by the recording application to start continuous recording. The mini-PC is configured to auto start the recording, and capturing application on the boot up.

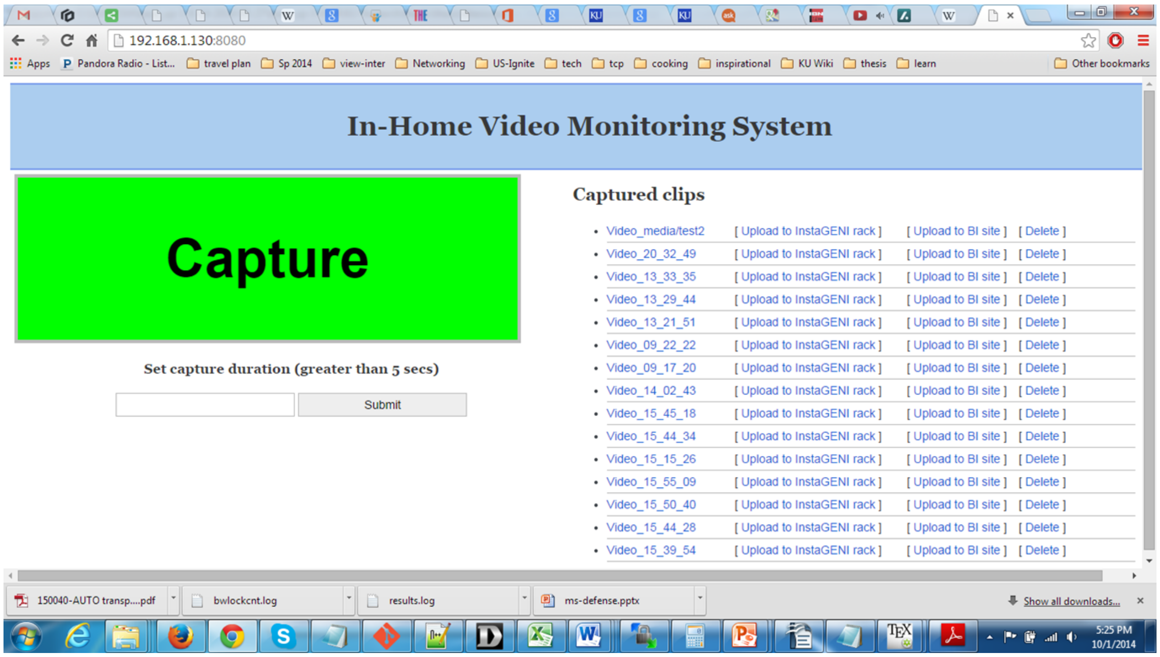

Figure 3.3. Capture application snapshot

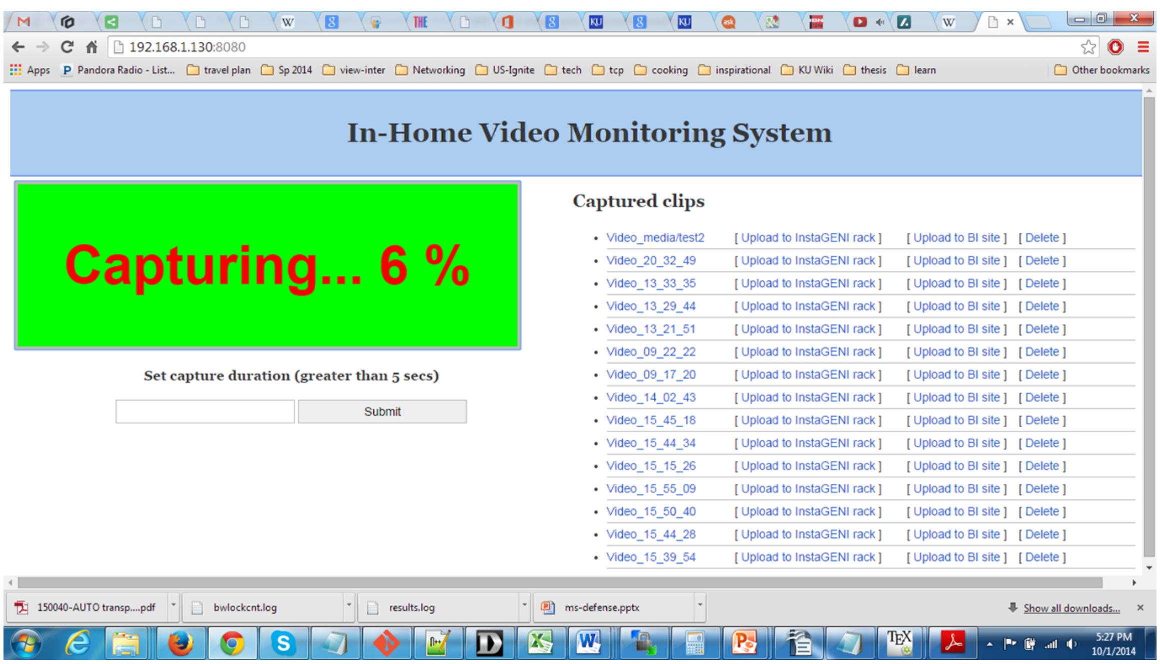

Figure 3.4. Capture in-progress snapshot

3.2.1 Recording

Although the camera supports FTP [34], NAS [35], HTTP [36] based transport, we choose to use RTSP/RTP [37, 38] based transport to able to perform continuous recording at high resolution, and also to avoid Firewall/NAT traversal issues highlighted in the background section. While using RTSP/RTP, UDP [39] is the default transport protocol, and camera acts as the RTSP server. The video processing module in the local storage device consists of recording and capturing components. Both are python scripts which use FFmpeg tool chain for recording and creating clips. Capture application hosts a local webserver based on python-pyramid [40] framework. We have configured the camera to employ H.264 [41] and G.711 [42] as video and audio codecs, respectively. The continuous recording is stored in circular fashion, as media segments, each one of them 5 seconds long. The media segments allow us to flexibly create the video clips of required size(in the granularity of 5 seconds), very quickly. The default circular buffer is configured in such a way to hold the recording for length of 2 days. All these configurations can be changed programmatically based on the requirements. Figure 3.2 shows the log of recording application.

3.2.2 Capture

When a capture event is triggered through a capture application running on local webserver, an event is recorded and video clip is created. The size of video clip is configurable through a field in the application. After creating the video clip the capture application allows the user to review the clip, upload it to a configured server, or delete it in case it is not captured correctly. Figures 3.3 to 3.5 shows the capture application in working.

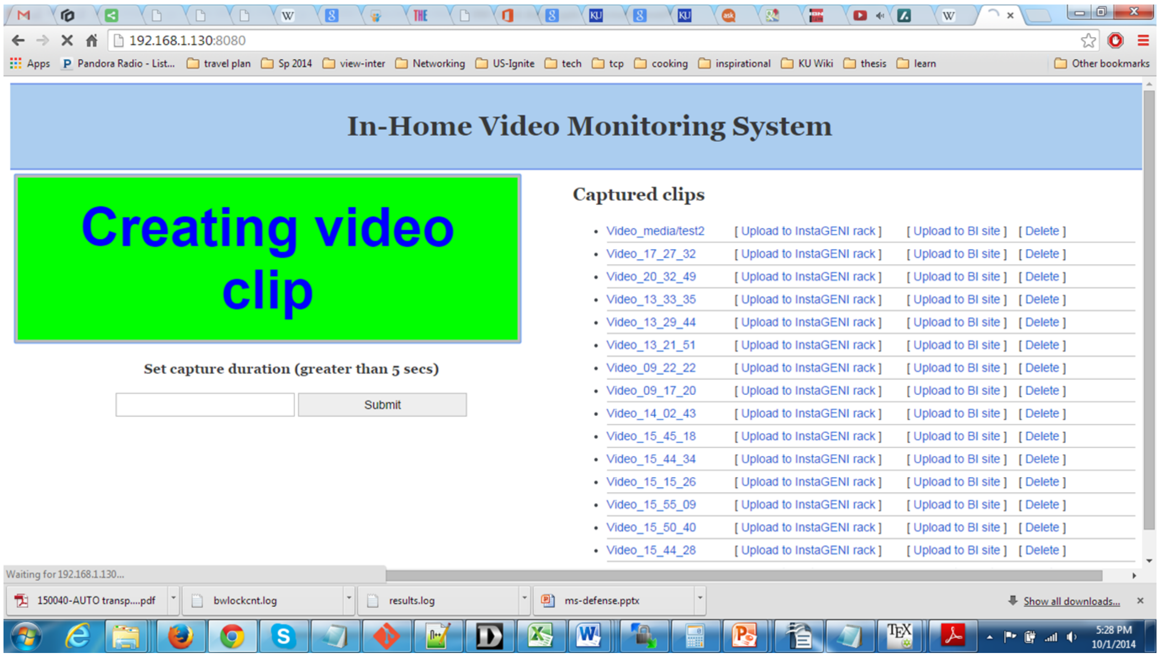

Figure 3.5. Creating clip snapshot

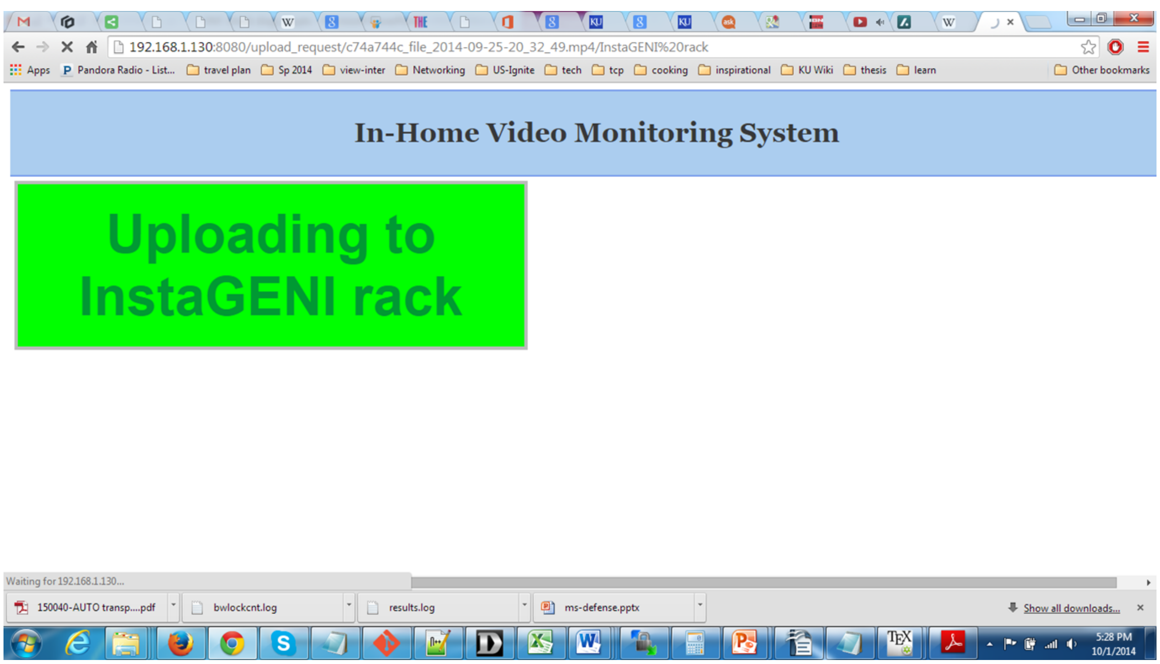

Figure 3.6. Uploading snapshot

3.2.3 Sharing

The sharing of video clips from caregivers to providers is established through telehousecalls application. A many to many relationship could exist between caregivers and the providers. The clinical experts acting as providers could review the video clips shared with them, and provide a timely feedback either through an email or phone call. At any point of time caregiver or the provider can request and schedule a two way video coaching. WPA2 based 802.11n connection is employed while local recording, and HTTPS while uploading of video clips to public and private clouds. Figure 3.6 shows the snapshot of uploading to InstaGENI rack

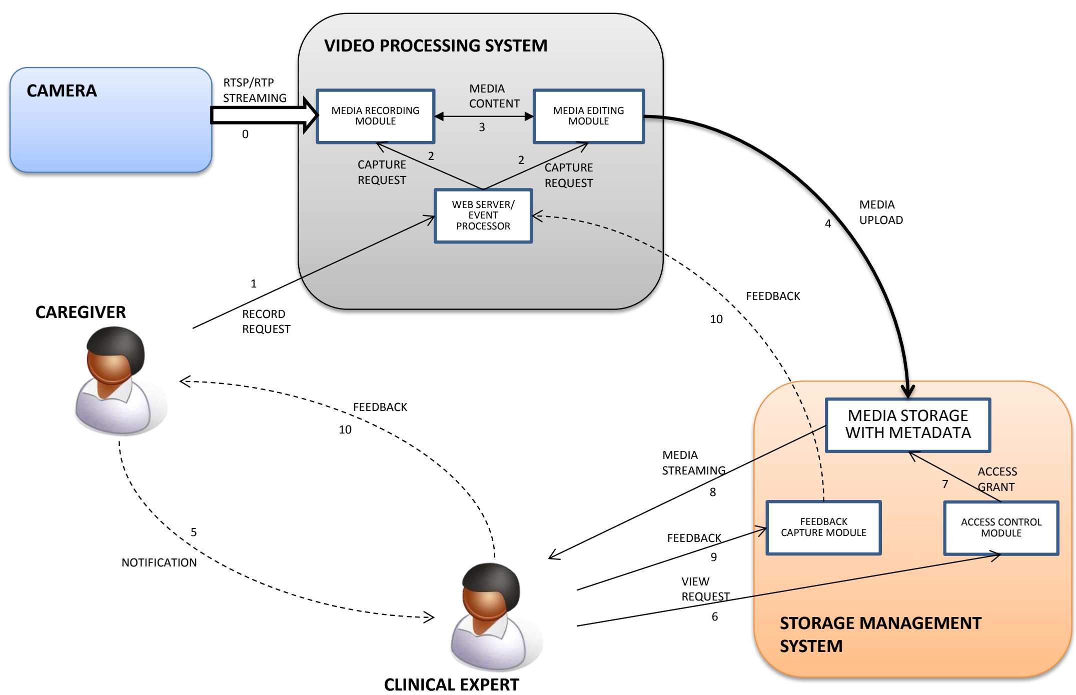

Figure 3.7. System block diagram

KU InstaGENI rack in GpENI/KanREN [43] network is utilized as a private cloud, and FTP server in our evaluation framework. We employ it to compare the performance with public cloud, and as destination for bulk transfer evaluations. An experiment is setup using Emulab/Omni interface to reserve a node on rack. L2TPv3 tunneling mechanism is used to interconnect the InstaGENIrack data plane with mini-PC with publicly accessible IP address. A publicly accessible GpENI node is used facilitate the tunneling between InstaGENI rack data plane, and the host device in the Internet. Basically, a secure, multi-threaded, python based HTTP server is deployed on the rack on which the content can be uploaded from mini-PC in the home.

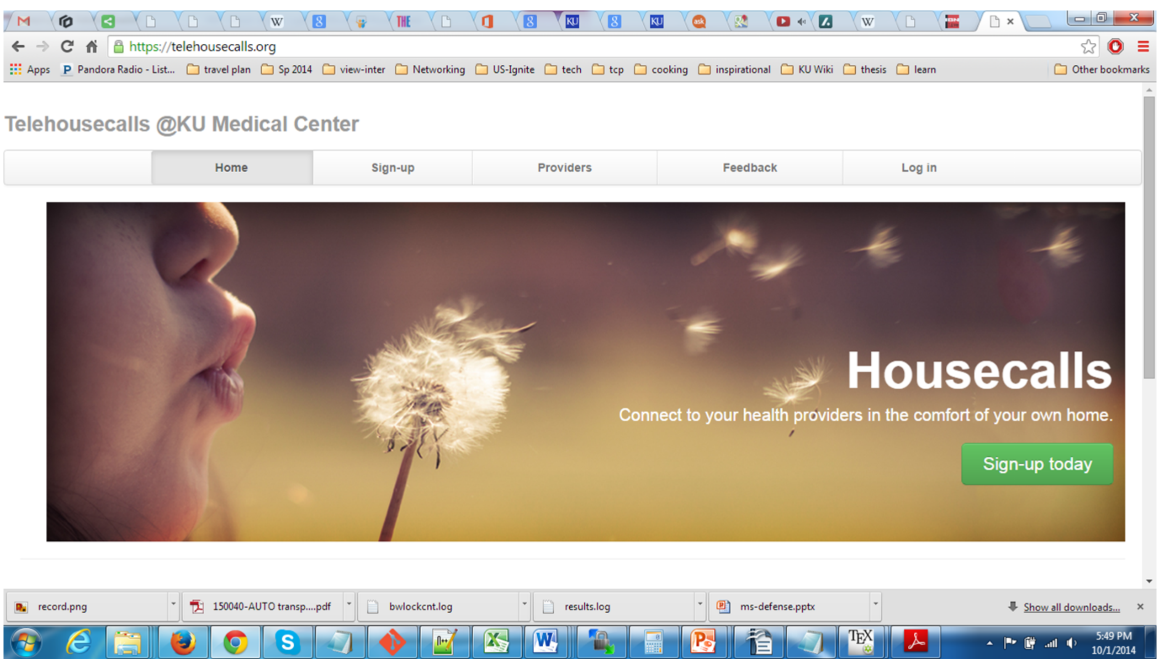

Figure 3.8. Telehousecalls application snapshot

3.3 Telehousecalls Application

Telehousecalls [44] is a browser based application through which caregivers and providers can log-in, and establish relationships. Many to many relationship can exist between caregivers and the providers. A caregiver can create video source for each of the monitoring equipment at home, and selectively share the particular video source with a provider. A video source is nothing but a camera and mini-PC combination at home, and is identified by a unique 8 character identifier. Each video clip created through capture application will have this unique identifier as prefix. Through this identifier a uploaded clip is uniquely associated with the caregiver account. This application also facilitates scheduling and starting the two way video coaching. We have integrated WebRTC into this application to enable two-way video coaching of caregivers.

We have utilized GCS (Globus Connect Server) and GCP (Globus Connect Personal) services from Globus [45] for bulk data transfers. These services internally use GridFTP.

Latency involved in storing and retrieving the video clips to and from the public and private cloud storage is measured. Google storage (both standard and DRA), Amazon S3 are employed as public cloud storage options. KU InstaGENI rack is utilized as private cloud. The complex cost models employed by different providers makes it difficult to uniformly compare them in terms of overall cost of deployment. In our evaluation we mainly focus on latency. gsutil [46], boto [47], wget [48], and general http upload utilities are employed to upload from, and, download to, public and private cloud.

WebRTC has been integrated into telehousecalls application as a two-way video chat solution for real-time coaching. Figure 3.8 shows the snapshot of telehousecalls home page. Telehousecalls application is developed by KUMC medical informatics team, and it is a HIPAA compliant application. HIPAA ((Health Insurance Portability and Affordability Act, 1996) provides generic guidelines for electronics health care transactions.

Chapter 4

Measurements and Analysis

This chapter presents the methodology for performance evaluation, and analysis of results.

4.1 Methedology

This section explains the environment, measurements and analysis part. The recording, capture, and telehousecalls application was pioleted in Kansas City homes with 1 Gb/s Google fiber connection. Healthy volunteers were recruited to play out a predefined kit. We setup three different video recording configurations, namely, Wired, 802.11n and 802.11n with poor signal strength (setting in which the camera and router were few meters away) , in 3 different homes. The measurements were carried out at 3 different times of the day – 11.00 am, 1.00 pm, and 3.00 pm – to uniformly spread the measurement instants. The following sections explain the measurement methodology for each component in the evaluation, as shown in figure 4.1.

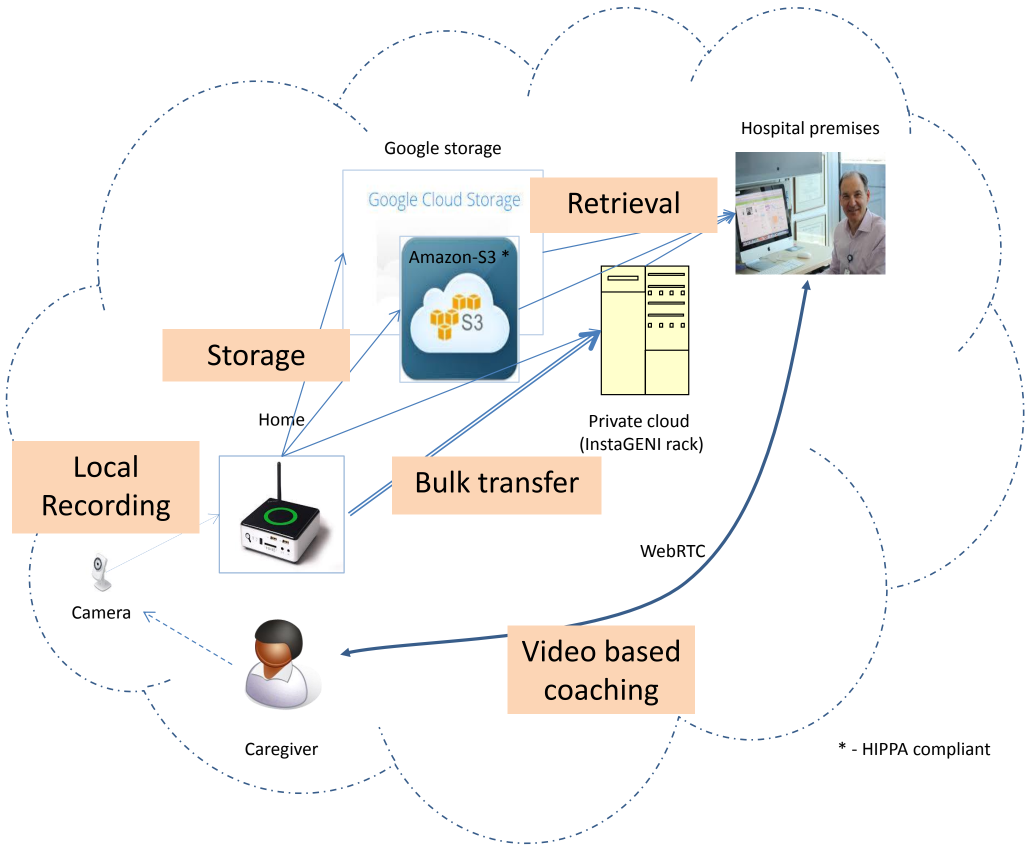

Figure 4.1. Measurement components

4.1.1 Bulk Transfer

The latency in transferring large number of media segments using FTP and GridFTP protocols is measured. The InstaGENI rack at KU is employed as server in all the schemes. We evaluate the performance when the client is in Google fiber, and Apogee ISP [49]. For standard FTP evaluation vsftpd server and mput on client are utilized. The latency is monitored through /var/log/vsftpd.log syslog on the Linux platform. For GridFTP evaluation, we utilized GCS (Globus Connect Server), and GCP (Globus Connect Personal) services from Globus. Transfer status is monitored asynchronously through globus online status command and used for computing the transfer latency. Default configurations of 2,2,20 are used

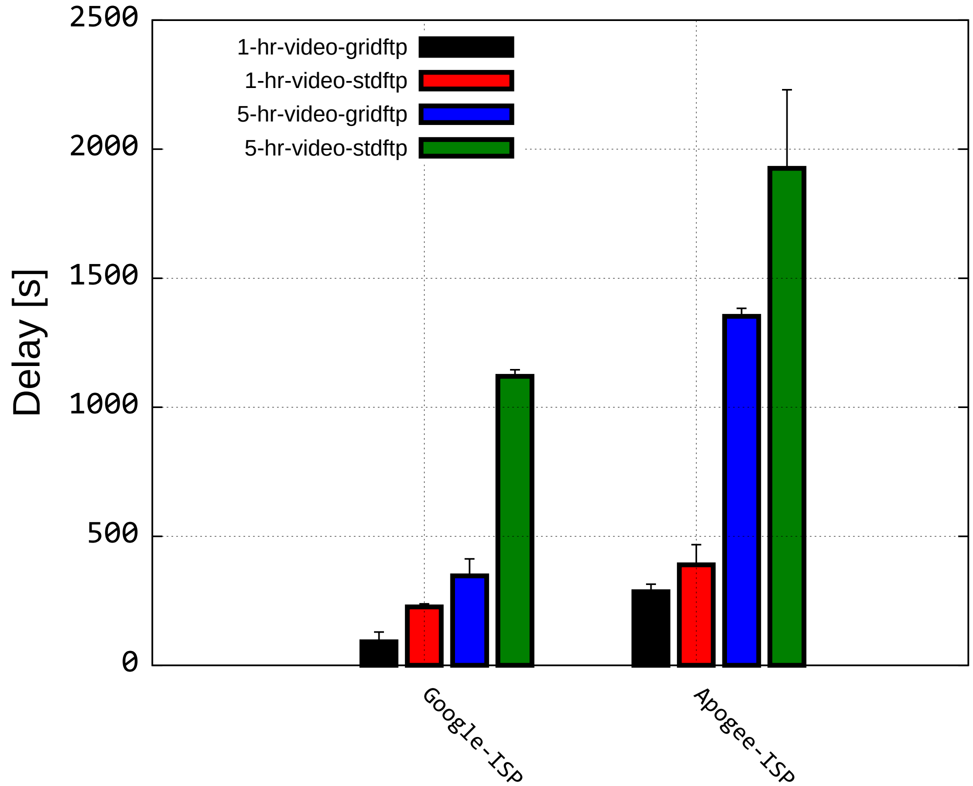

Figure 4.2. Bulk data transfer delay with FTP and GridFTP

for concurrency, parallelism, and pipelining of GridFTP transfers. These default are computed based on the segment size and number of segments in the transfer. Concurrency is applicable in multi-homed scenarios and falls back to parallelism otherwise. So, in our evaluation, 4 parallel connects were established and the data was striped across the 4 connections.

4.1.2 Cloud Storage/Retrieval

The standard utilities provided by cloud vendors, namely, boto, and gsutil tools are used for communicating with public cloud storage. Python standard library http server and client modules are used for communicating with private cloud storage. Also, we have used open source OpenSSL utility to encrypt and

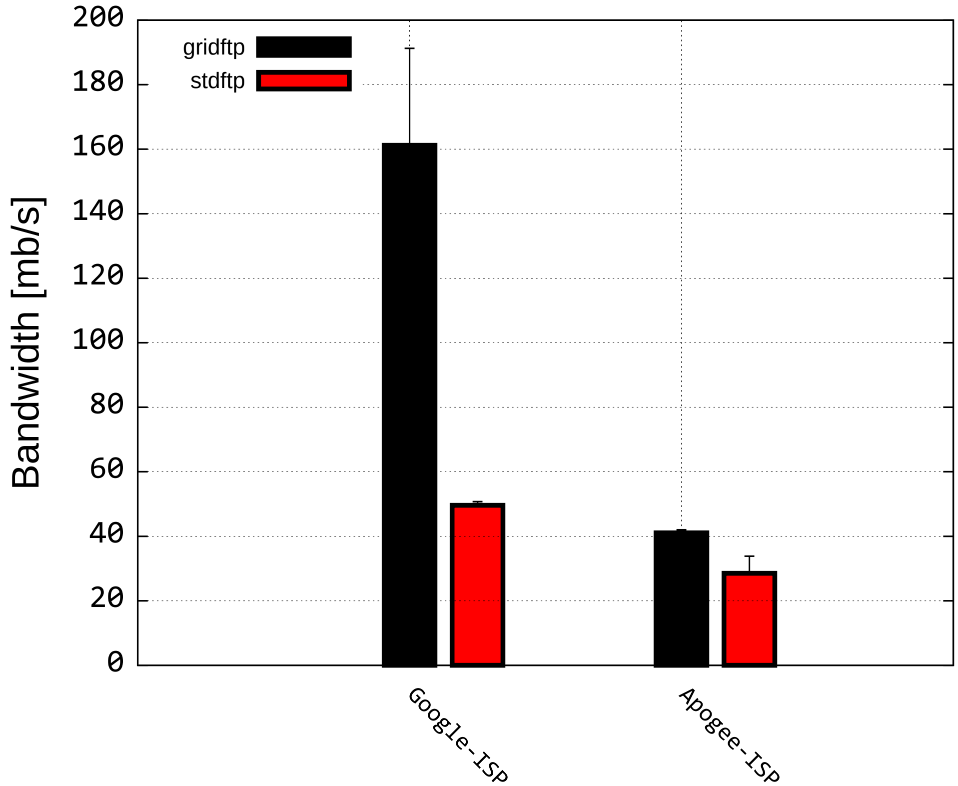

Figure 4.3. Bandwidth achieved with FTP and GridFTP

decrypt using AES128 algorithm , and also to generate and verify MD5 hash of video clips. The total media clip storage delay (D_{s}) is the summation of computing hash (d_{dgt}), encryption (d_{enc}), and cloud storage latency (d_{str}). i.e.

$$D_{\mathrm{s}} = d_{\mathrm{dgt}} + d_{\mathrm{enc}} + d_{\mathrm{str}}$$

Similarly, the retrieval delay (D_{r}) is summation of cloud retrieval latency (d_{ret}), decryption (d_{dec}), and hash verification latency (d_{dgt}).

$$D_{\mathrm{r}} = d_{\mathrm{ret}} + d_{\mathrm{dec}} + d_{\mathrm{dgt}}$$

These latency values are evaluated in 3 different ISPs, and to 4 different cloud destinations, as explained in section 3.4.

4.1.3 Local Recording

We monitor the recording performance in both wired and 802.11n LAN. An environment with poor 802.11n signal strength is considered for evaluation. tcp-dump is used to collect network captures of recording session. tshark [50] , awk, and shell scripts are used for extracting the RTP payloads and other metrics from the network capture files. Video frame jitter is computed using the RTP time-stamp and the arrival time of frames. If t_1, r_1 are the RTP time-stamp and arrival time-stamp of first frame, and, t_2, r_2 are the RTP time-stamp and arrival times-tamp of next frame, then the jitter is computed as $|(r2 - r1) - (t2 - t1)|$. Similarly, average frame-rate is computed by calculating average number frames per second, and the bandwidth consumed as number of bytes recorded divided by total duration.

Perceived video quality is subjective and is based on the viewers perception [22]. So, commonly used QOS (Quality of Service) parameters may not indicate the perceived video quality. The pure, quality verification mechanisms: PSNR (peak signal to noise ratio), and MSE (mean squared error,) require reference to original video, which is not possible obtain in our application. Other QOE (Quality of Experience) measures, such as, MOS (mean opinion score) are based on the mean rating of different viewers. We have used a similar approach by getting the opinion of clinical expert on the feasibility of using the recorded clips for diagnoses and treatment. We collect the CEOS (clinical expert opinion score) for video clips recorded in different home environments.

4.1.4 WebRTC Performance

The two-way video coaching performance is evaluated by continuously measuring RTT (round trip time), frame-rate, and bandwidth of WebRTC. Our evaluation span across three access networks, namely, wired LAN (802.3 switched Ethernet), 802.11n, and LTE. One end of the communication is always on the wired LA, with good upload and download bandwidth. While, the data is collected on the other end point in different access network. We collect the metrics through webrtc-internals, an extension in the chrome browser. The data is downloaded as JSON object and required fields are parsed through a script. Each metric is aggregated over a period of 1 second.

4.2 Bulk Transfer Measurements

Figure 4.2 shows the latency for transferring recorded video segments for 1 hour and 5 hour duration from home premises to InstaGENI rack. Figure includes measurements for both, standard FTP and GridFTP protocols, and on two ISPs, Google fiber, and Apogee. The video is recorded at 1280x800 resolution. Each video segment is of 5 seconds duration, and is approximately 2 MB in size. 1 hour recording contains 720 segments, and 5 hour recording contains 3600 segments. We carried out six iterations for each measurement and computed the mean and 95% confidence interval. The latency on Google fiber ISP is , 80 and 375 seconds for 1 hour and 5 hour video transfer with GridFTP. With standard FTP the values are 200 seconds and 1310 seconds, respectively. Similarly, on Apogee ISP, the latency values are 285 and 1350 seconds with GridFTP, and, 390 and 1950 seconds with standard FTP. So, on Google fiber ISP the latency are roughly 30% to 300% lower compared to apogee ISP. Also, we get 30% to 300% reduction in latency by using GridFTP instead of standard FTP. Generally, standard FTP has higher variation than GridFTP because many connection establishments and tear downs happen during the transfer, whereas GridFTP uses the pipelining with same connections to complete the transfer. The deviation from mean is higher on Apogee with standard FTP compared to Google fiber ISP, because, we think the number of users in Apogee is higher, which might contribute to the larger variation in best effort network.

Figure 4.3, shows the upload bandwidth achieved on Google fiber and Apogee, with standard and GrdiFTP protocols. GridFTP achieves 160 Mb/s on Google fiber and 42 Mb/s on Apogee ISP. Whereas, standard FTP achieves 40 Mb/s on Google fiber and 28 Mb/s on Apogee ISP. The bandwidth is 400% better on Google ISP both for GridFTP and 30% better for standard FTP.

4.3 Storage and Retrievel Delay

In this section we measure the latency in storing and retrieving video clips from public and private cloud storage. We consider Amazon S3, Google standard and DRA (durable reduced availability) storage as public cloud providers, and, InstaGENI rack as private cloud storage provider. In all scenarios each test is repeated 6 times, and mean and 95% confidence interval are computed. 1 min, 5 min, and 10 min video clips are considered for storage and retrieval experiment. 1 min clip is approximately 20 MB, 5 min clip is approximately 100 MB, and 10 min clip is approximately 200 MB in size. Figures 4.4 to 4.6 show the latency for 1 min, 5min, and 10 min clips, respectively. Each figure shows the storage and retrieval delays on 3 different ISPs, to be specific, Google fiber, AT&T U-verse, and Apogee, and to 4 different storage destinations, namely, Google-storage, Google-

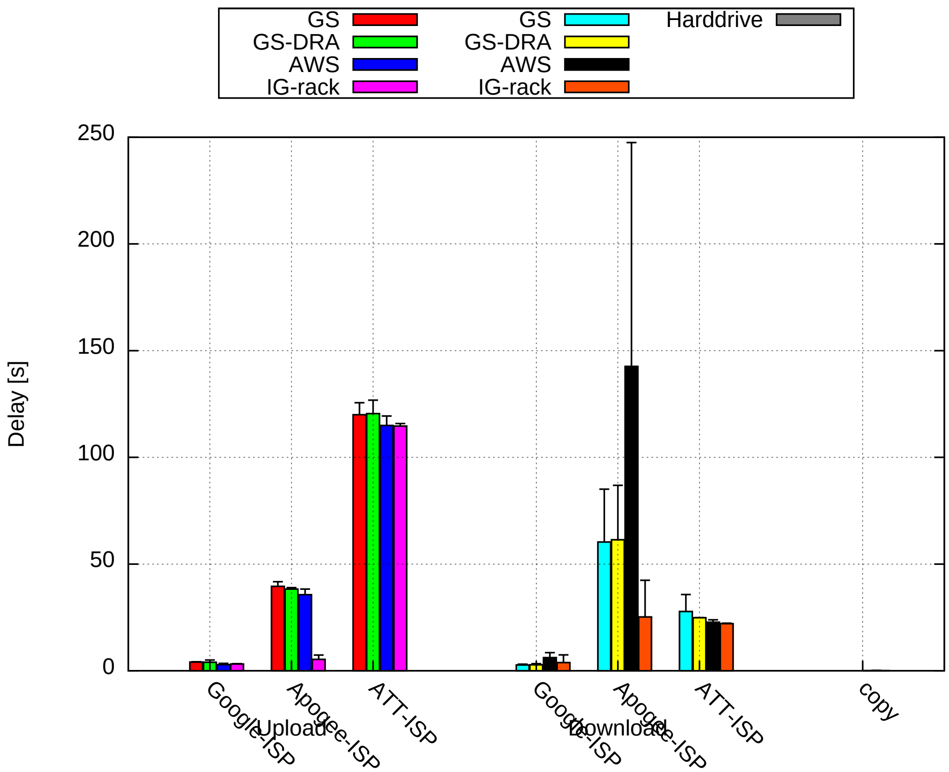

Figure 4.4. Storage and retrieval latency for 1 min video clip

dra-storage, amazon S3, and InstaGENI rack. We also show the hard drive read write latency for each clip for comparison. The acronyms, GS, GS-DRA, AWS, and IG rack correspond to Google storage, Google dra storage, Amazon S3 and InstaGENI rack, respectively.

The latency in copying file from one location to another in a hard drive is 0.069, 1.2, and 3.3 seconds for 20 MB, 100 MB and 200 MB file, respectively, with 5200 RPM hard drive on a 1.6 GHz processor. The storage and retrieval delays observed in all the scenarios are much higher than the basic disk read-write delays. This indicates that, network latency, and possibly, pre and post processing latency in cloud dominate the overall delay to public cloud storage.

Except for AT&T U-verse, storage and retrieval latency to InstaGENI rack are

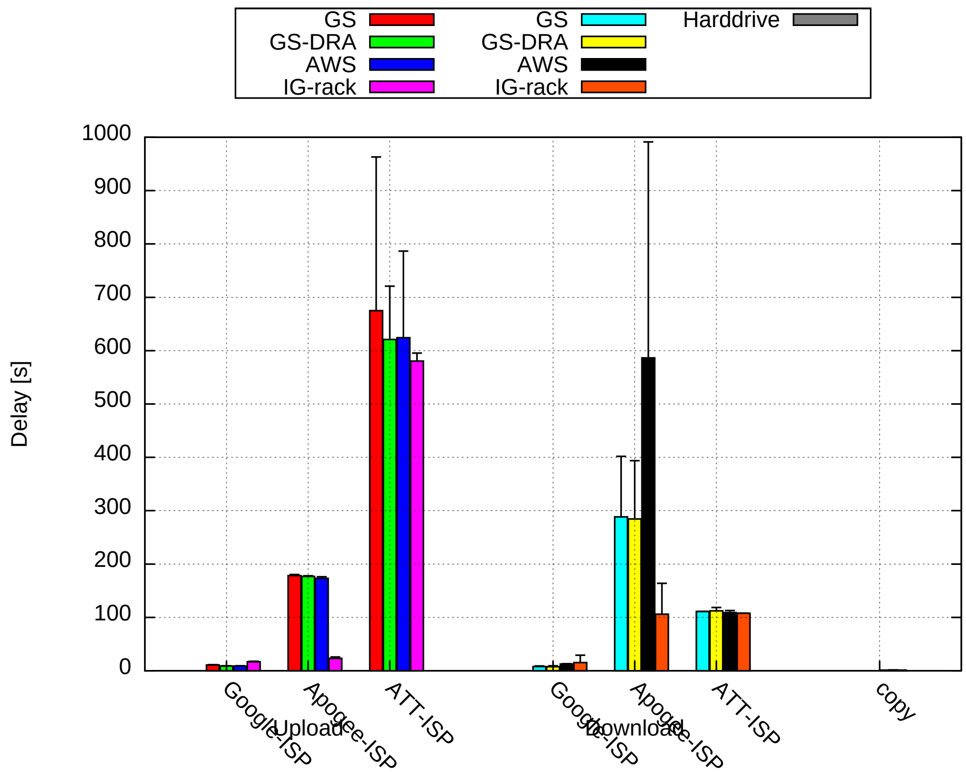

Figure 4.5. Storage and retrieval latency for 5 min video clip

less than the latency observed on Amazon S3 and Google storage. Because, probably there is less traffic between KanREN, on which IG rack is located, and other commercial ISPs. AT&T U-verse seems to have fine grained traffic shaping.

For 1 minute clip, the upload and download latency on Google fiber are of the order of 5 seconds. They are 6 to 8 times higher in the Apogge ISP, and about 30 times on AT&T U-verse. Similarly, for 5 minute clip the latency on Google fiber are of order of 10 to 15 seconds and is 15 to 20 times higher on Apogee ISP, and 30 to 40 times higher on AT&T U-verse. For 10 minute clip the latency on Google fiber are of the order 15 to 20 seconds, and they are 15 to 20 times higher on Apogee ISP, and 70 to 80 times higher on AT&T U-verse. From this we can conclude that there are some fixed latency for pre/post processing and these

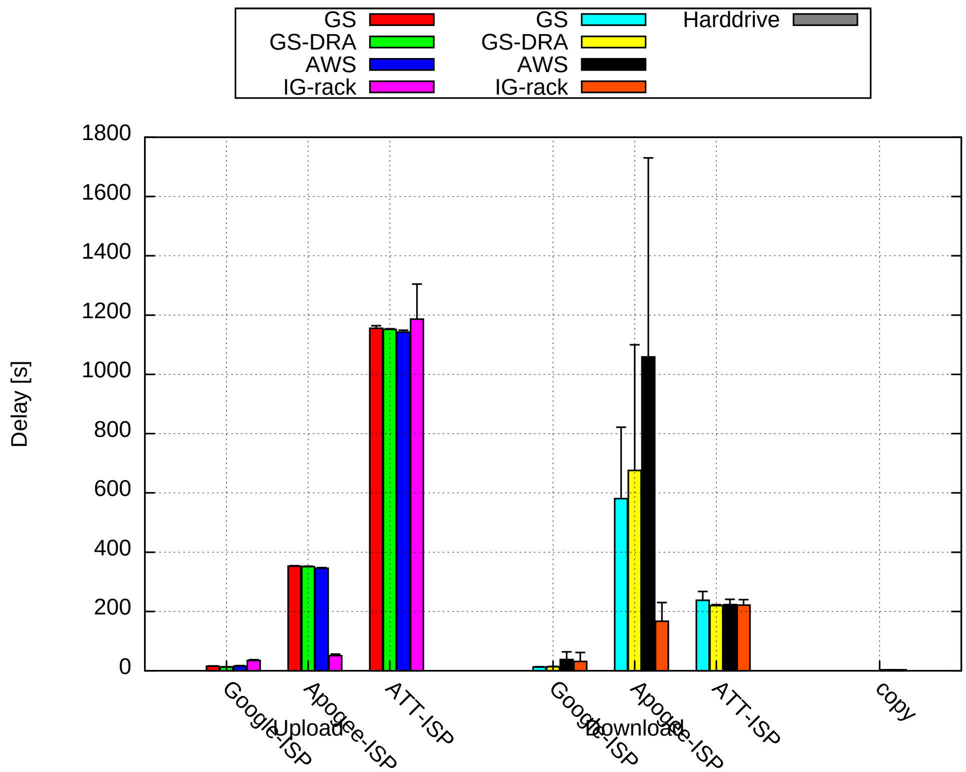

Figure 4.6. Storage and retrieval latency for 10 min video clip

values dominate the overall latency for smaller clips. As the object size increases the amortized fixed latency becomes negligible and network latency becomes the dominant component. So, for bigger objects (> 100 MB) Google fiber is 15 to 20 times faster than Apogee ISP.

The variations in latency on Google fiber ISP are very small. On Apogee, both upload and download latency experience large variations. We hypothesize that, Google fiber is still in nascent stage and is more over-provisioned compared to Apogee ISP. Also, Apogee being a campus residence Internet service provider, does not do fine grained traffic policing as is done in purely residential services. On Google fiber and Apogee, the latency to Amazon S3 is higher compared to other cloud destinations. It also exhibits the highest variation. This is probably be-

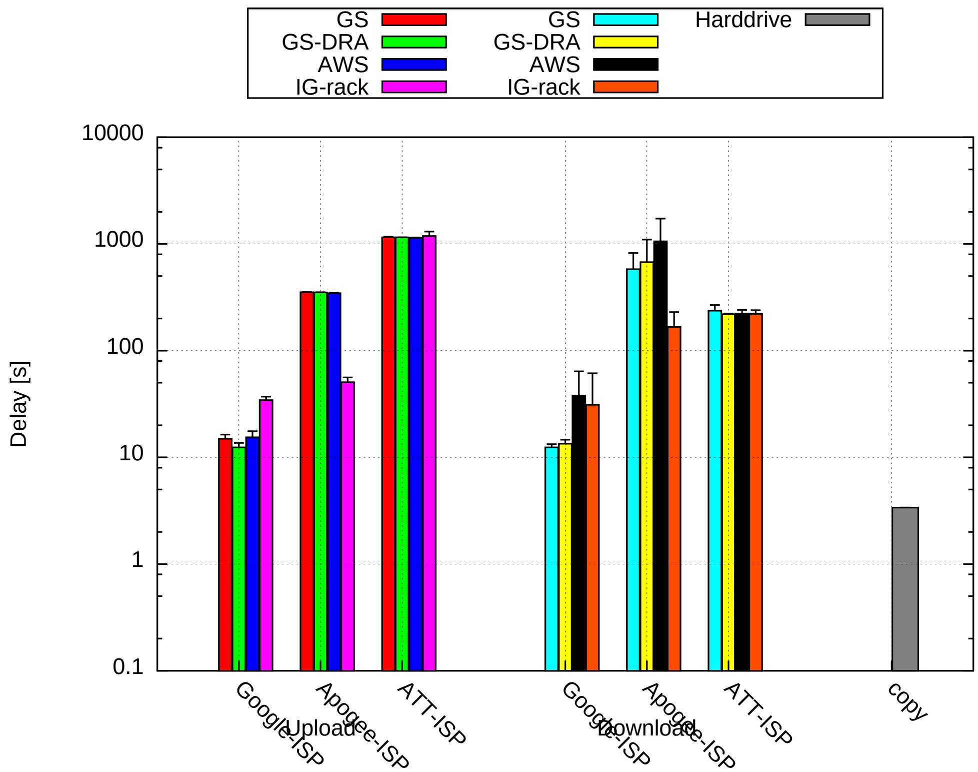

Figure 4.7. Storage and retrieval latency on log scale

cause, Amazon S3 being most widely used cloud service, experiences higher loads leading to higher latency and variations. AT&T U-verse having a fine grained traffic shaping, shows , in general, lower delay variations to all cloud destinations. Generally the upload latency variation is smaller compared to download latency variation. Again, we believe that, on all ISPs, the number of upload flows is much smaller than the number of download flows. On best effort network, more number of flows introduces larger variations.

Figure 4.7 shows the storage and retrieval delays for 10 minute clip in the log plot. It can be observed that, the Google fiber is order of magnitude higher than plain copy. Similarly, Apogee is order of magnitude slower than goolge fiber, and AT&T U-verse is order of magnitude slower than Apogee.

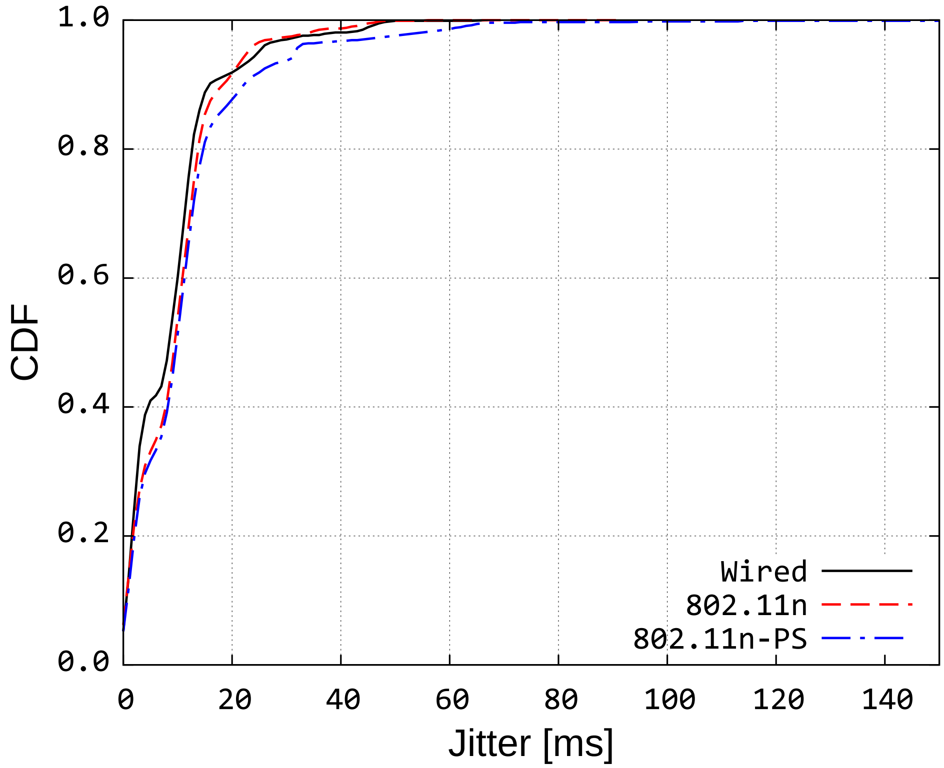

Figure 4.8. Video frame jitter for live recording

Table 4.1. Encryption, decryption, and digest time for different video clip sizes

	1 min	**5 min**	**10 min**
Encryption [ms]	324.34	1899.53	3850.00
Decryption [ms]	341.72	1750.23	3722.30
MD5 digest [ms]	52.79	225.36	446.65

Table 4.1 shows the additional latency during encryption, decryption, and digest computation for 1, 5, and 10 minutes clips. OpenSSL AES-128 cbc algorithm is used for encryption, whereas MD5 is used for cryptographic digest (hash) calculation.

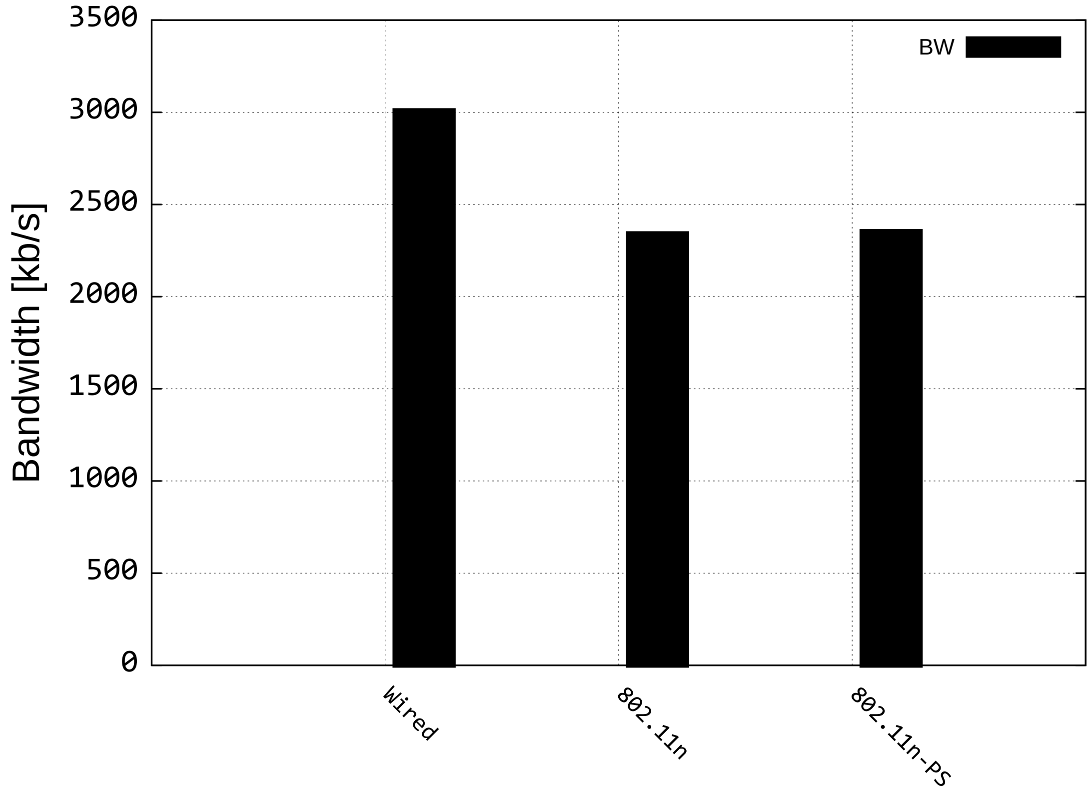

Figure 4.9. Average bandwidth of live recording

Table 4.2. CEOS table

Access link type	CEOS	Remarks
802.11n-PS	Acceptable	Excellent clarity, focus and lighting. Audio, video sync was bit off
802.11n	Acceptable	Quality was adequate.
Wired	Acceptable	Focus and lighting were not optimal. Audio, video sync was good

4.4 Video Recording Performance

In this section we measure the continuous video recording performance in LAN. The measurements are collected for 10 minutes duration. The camera is configured for 30 frames/s and 1280X800 video resolution. A PS suffix in the abbreviated notation used in plots indicate the poor signal. Figures 4.8 to 4.10 show the video

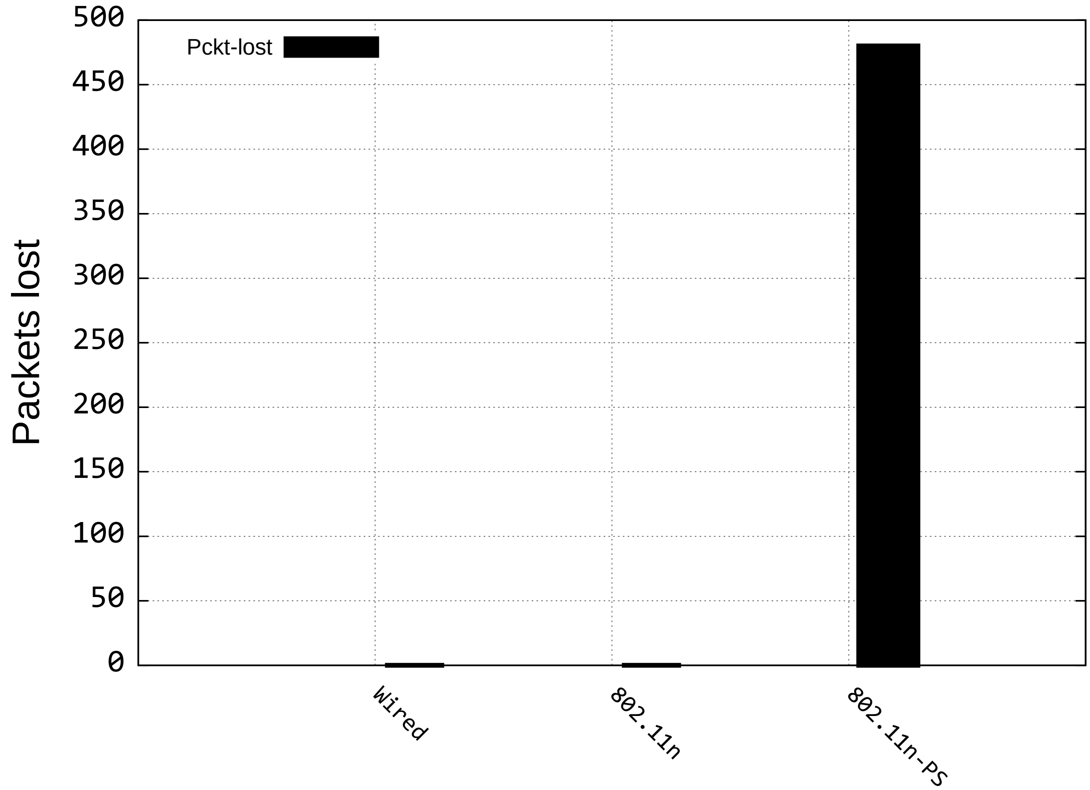

Figure 4.10. Packets lost during live recording

frame jitter, average bandwidth achieved, and the packets lost.

In wired and 802.11n connection, the jitter is less than 50 ms for all frames. In reduced signal strength scenario, the maximum delay variation stretches to 70 ms. On wired links the achieved bandwidth is 3000 kb/s. Whereas, on 802.11n links it is around 2400 kb/s. There were no packets lost with wired link, but around 500 packet drops are observed poor signal 802.11n connection.

As explained in the background section the perceived video quality is based on many factors including the perception of viewer. In our application, to subjectively measure the quality of recorded clips in different LAN environments, we have included the feedback from clinical experts, called CEOS (clinical expert opinion score). It is a rating of Good, Acceptable, and Poor. It is based on the expert

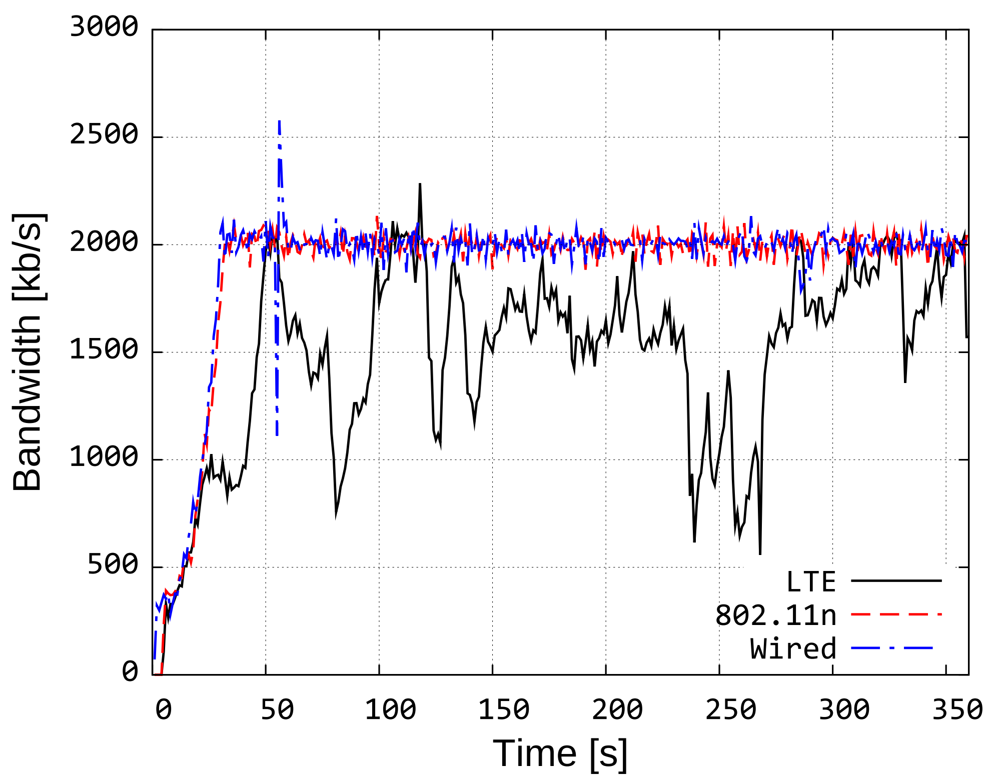

Figure 4.11. WebRTC sent video bandwidth variation

opinion as to whether the video is usable for diagnosis and treatment. Table 4.2 shows presents the COES rating for in different LAN environments.

Table 4.3. WebRTC average metrics

	Sent BW [kb/s]	**Rec BW [kb/s]**	**RTT [ms]**	**FR [frames/s]**
LTE	1479.095	1372.628	162.669	29.357
802.11n	1891.235	1981.578	35.273	29.401
Wired	1900.506	1959.620	8.638	29.741

Table 4.4. WebRTC STD metrics

	Sent BW [kb/s]	**Rec BW [kb/s]**	**RTT [ms]**	**FR [frames/s]**
LTE	322.268	472.615	60.26	1.706
802.11n	42.981	108.870	2.432	0.467
Wired	47.921	30.088	4.778	0.420

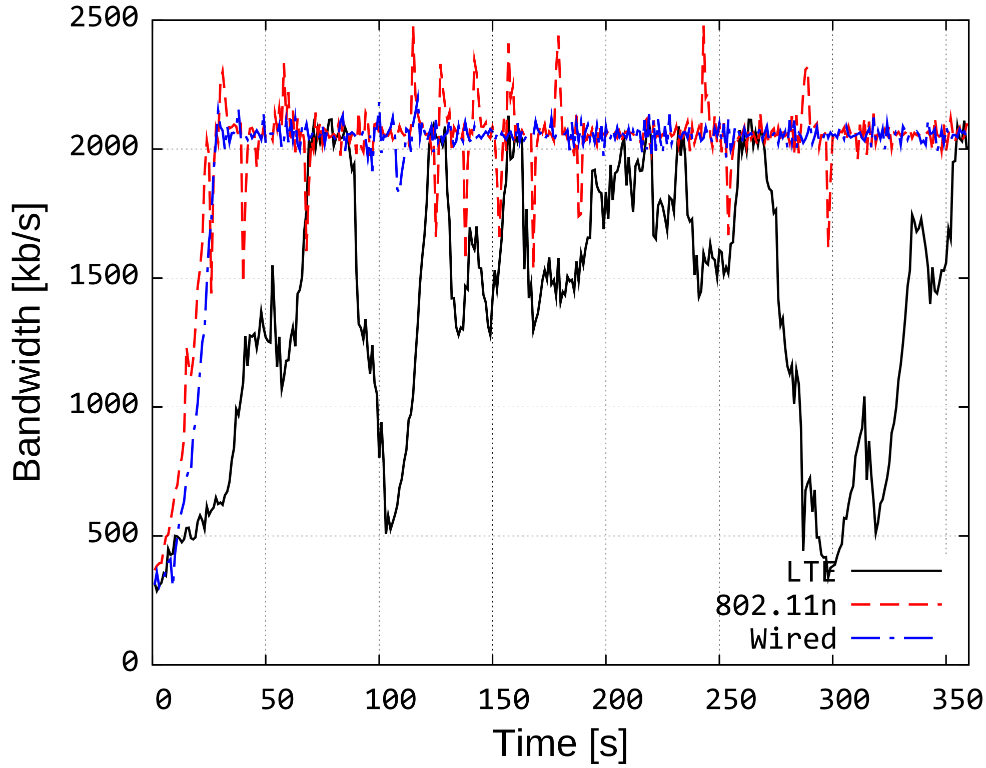

Figure 4.12. WebRTC received video bandwidth variation

4.5 Two-way Video Chat Performance

In this section we measure the performance of two-way video chat using WebRTC. ICE/STUN/TURN triplet service is used for connection establishment and NAT traversal. WebRTC utilizes RTP/UDP as the main transport protocol for the media, and is governed by GCC (Google congestion control) algorithm. DTLS-SRTP secured protocols are used to achieve the 3 facets of security, namely, authentication, confidentiality, and integrity. By default, WebRTC uses VP8 as video codec, and Opus as audio codec. Performance is measured on 3 access networks as explained in section 4.5. Figures 4.11 to 4.14 show the variation of sent bandwidth, received bandwidth, RTT, and frame-rate. The experiment is con-

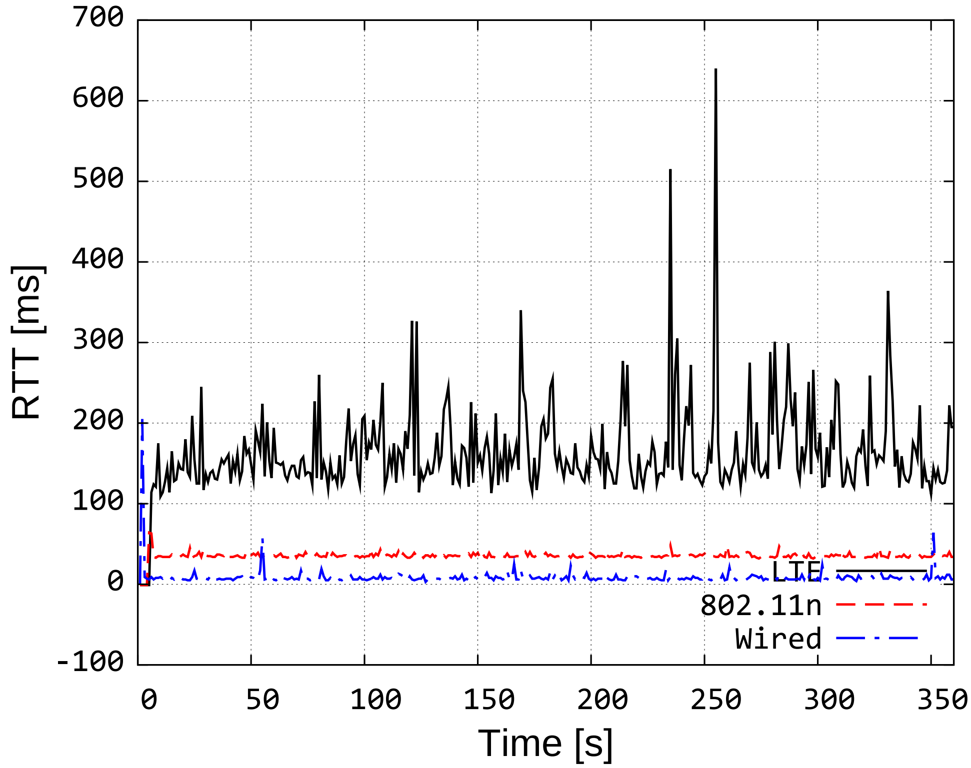

Figure 4.13. WebRTC RTT variation

ducted for 5 min. WebRTC starts with 300 kb/s bandwidth and reaches steady state at 2 Mb/s within 1 min from the start. The table 4.3 shows the mean values of bandwidth, RTT, and frame-rate for the 3 access networks.

Table 4.3 and 4.4 shows the mean and standard deviation of WebRTC metrics on different access links. The sent bandwidth, received bandwidth, RTT, and frame-rate are stable at 1900 kb/s, 1959 kb/s, 8 ms, and 29.75 frames/s for wired connection. In 802.11n network, the values for same metric are, 1890 kb/s, 1981 kb/s, 35 ms, and 29 frames/s, respectively. Though RTT increases by 4 times in 802.11n connection, the bandwidth and frame-rate are quite stable. Because, the RTT and its variation are still small *wrt* real-time requirement of 100 ms. On LTE access link, RTT increases by 20 times compared to wired link, leading to

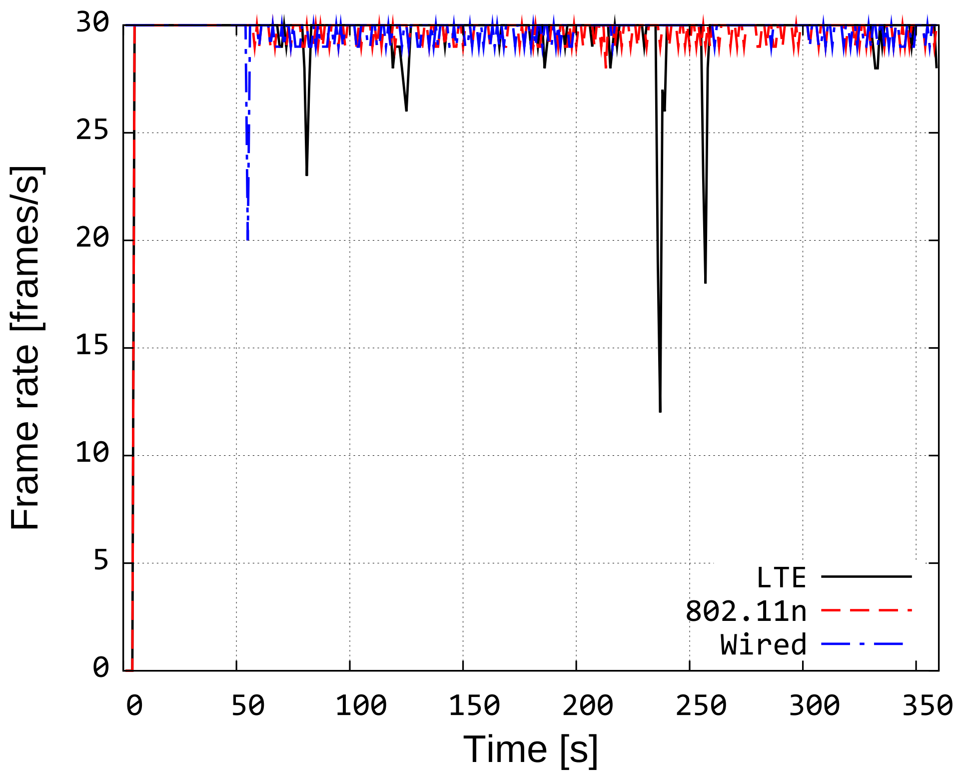

Figure 4.14. WebRTC framerate variation

the sent and received bandwidths dropping by more than 30%. Also, the standard deviation is approximately 30% of the mean value for both bandwidth and RTT, which is highest among all the access links. The frame-rate is maintained as characteristics of Google congestion control and adaptive video encoding. But, with RTT being much higher than the real-time recommended 100 ms threshold, and large variations, there is perceivable degradation in video quality. This can be attributed to commonly known buffer bloat problem on cellular networks.

4.6 Summary

On an average, the video monitoring consumes 3 Mb/s bandwidth (continuously) in home network, with minimum of 1 Mb/s, and maximum of 5 Mb/s. All the 3 environments produce acceptable quality video clips. A 10 min video clip storage and retrieval to cloud storage has approximately 20 seconds of latency in Google fiber. It is approximately 15 to 20 times better in Google ISP than compared to Apogee ISP, and about 80 times better compared to AT&T U-verse. Bulk transfer of 5 hours of recorded video(roughly 6.5 GB), takes on an average of 6 minutes, in Google ISP. Theoretically, it would have taken several hours for this data transfer in AT&T U-verse. In general, GridFTP boosts the performance by 30% over the standard FTP, and upto 300% in high speed paths. The two-way video coaching using WebRTC achieved approximately 2 Mb/s of bandwidth with 30 frames/s frame-rate and 8 ms average RTT on the wired access link. The same values for 802.11n and LTE access links were, 2 Mb/s, 30 frames/s, 30 ms, and 1500 kb/s, 30 frames/s, 160 ms, respectively.

Chapter 5

Conclusions and Future Work

This chapter presents the conclusions and future work.

5.1 Conclusions

We have developed an advanced video based monitoring system to aid the caregivers of dementia patients. Verified a prototype of the system by recruiting healthy subjects, and conducting field trials in Google fiber space. After the first trial at Google fiber homes, we received feedback on usability issues of record and capture application. Also, recorded contents were not being played back on nexus tablet. The record and capture application were improvised by adding restart and stop provision. The media container was changed to MP4 with H.264 video and AAC audio codeces. This resolved the compatibility issues on nexus tablet.
We measured the performance of different components in our system. On an average, the video monitoring consumes 3 Mb/s bandwidth (continuously) in home network, with minimum of 1 Mb/s, and maximum of 5 Mb/s. With multi camera monitoring, the required bandwidth in local LAN might create scalability issues,

as the achievable bandwidth with 802.11n is much less than the maximum possible bandwidth of 600 Mb/s. Although enough bandwidth is available on the backbone network in Google fiber to support multi camera recording and playback, the first link in 802.11n medium will be the bandwidth bottleneck. A 10 min video clip storage and retrieval to cloud storage has approximately 20 seconds of latency in Google fiber. It is approximately 15 to 20 times better in Google ISP than compared to Apogee ISP, and 80 times better compared AT&T U-verse. Bulk transfer of 5 hours of recorded video(roughly 6.5 GB) takes on an average of 10 minutes, in Google ISP. Theoretically, it would have taken several hours for this data transfer in AT&T U-verse. In general, GridFTP boosts the performance by 30% over the standard FTP. The two-way video coaching using WebRTC achieved approximately 2 Mb/s of bandwidth with 30 frames/s frame-rate and 8 ms average RTT on the wired access link. The same values for 802.11n and LTE access links were, 2 Mb/s, 30 frames/s, 30 ms, and 1500 kb/s, 30 frames/s, 160 ms, respectively.

The quality of video clips is not affected by the conditions in local area network. In 1 Gb/s Google fiber ISP, we could practically achieve 100 Mb/s to popularly used cloud storage providers. The large bandwidth in WAN helps in moving around large quantities of data for real-time diagnosis and treatment. Fine tuning the standard protocols for high bandwidth networks can reduce the latency by 30%. WebRTC performance is quite stable on low BDP (bandwidth delay product) paths. On paths with high delay and large buffers, the performance is wavy.

5.2 Future work

In future we would like to evaluate the performance of using multiple cameras in the home, and real-time multiple video playback. We would like to evaluate the feasibility of doing multiple recording in home environment with 802.11n and 802.11ac networks. As an alternate to COTS IP camera, we would like to evaluate using Dropcam for continuous recording. As Dropcam has optimized the video encoding for cloud streaming by achieving maximum bit rate of 512 kb/s, it could be an ideal solution for addressing scalability problem in multi camera monitoring system. However, security and privacy concerns have to addressed in Dropcam's usage, whereas in the current local recording based solution, by design, privacy and security concerns are minimized. In the current evaluation 5 sec media segments are used. In future we want to evaluate the effects of varying media segment sizes on storage retrieval, and bulk data transfer latency. Currently, caregivers need to identify the behavioral changes and initiate the video capture. In future a video/audio based behavior detection algorithms can be integrated, so that, video capture can be automated. So far, InstaGENI rack is been used as a standalone private cloud entity. The mini-PC used in home environment can made accessible in InstaGENI data plane through some sort of layer 2 tunneling mechanism. This will give us freedom to exploit the GENI testbed to execute compute and memory intensive algorithms on the collected data. This could also be used for anonymization of video clips to protect the privacy of patients. Multiparty video conferencing is still at nascent stage. Google fiber environment can be used to verify the practical usability of mesh or hub based multiparty video conferencing.

References

[1] National Alliance for Caregiving and AARP. (2009). Caregiving in the U.S. 2009: National Alliance for Caregiving in collaboration with AARP. 2009.

[2] Wimo A. and Prince M. World Alzheimer Report 2010: Alzheimer's Disease International. 2010.

[3] Hooker K., Bowman S. R., Coehlo D. P., Lim S. R., Kaye J., Guariglia R., and Li F. Behavioral Change in Persons With Dementia. *The Journals of Gerontology Series B: Psychological Sciences and Social Sciences*, 57(5):453 –460, 2002.

[4] Talley R. C. and Crews J. E. Framing the public health of caregiving. *American Journal of Public Health*, 97(2):224 –228, 2007.

[5] FFmpeg. `https://www.ffmpeg.org/`.

[6] WebRTC. `http://www.webrtc.org/`.

[7] Nicholas Bastin, Andy Bavier, Jessica Blaine, Jim Chen, Narayan Krishnan, Joe Mambretti, Rick McGeer, Rob Ricci, and Nicki Watts. The InstaGENI initiative: An architecture for distributed systems and advanced programmable networks. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, 61:24 –38, March 2014.

[8] OpenSSL. `https://www.openssl.org/`.

[9] Google Fiber. `https://fiber.google.com/cities/kansascity/`.

[10] GENI: Global Environment for Network Innovations. `http://www.geni.net/`.

[11] KUMC NetworkInnovation. `https://informatics.kumc.edu/work/wiki/NetworkInnovation`.

[12] Reischl U. and Oberleitner R. Development of a telemedicine platform for the management of children with autism. *German Journal for Young Researchers*, 1(1), 2009.

[13] Oberleitner R., Elison-Bowers P., Reischi U., and Ball J. Optimizing the personal health record with special video capture for the treatment of autism. *Journal of Developmental and Physical Disabilities*, 2007.

[14] Skubic M, Alexander G, Popescu M, Rantz M, and Keller J. A smart home application to eldercare: current status and lessons learned. *Technol Health Care*, 17(3):183 –201, 2009.

[15] Tsai J and Rosenheck RA. Use of the internet and an online personal health record system by US veterans: comparison of Veterans Affairs mental health service users and other veterans nationally. *J Am Med Inform Assoc*, July 2012.

[16] Neal D. and Rahman S.M. Video surveillance in the cloud-computing? *Electrical & Computer Engineering (ICECE), 2012 7th International Conference on*, pages 58 –61, December 2012.

[17] Rodriguez-Silva D.A., Adkinson-Orellana L., Gonẓlez-Castano F.J., Armino-Franco I., and Gonẓlez-Martinez D. Video Surveillance Based on Cloud Storage. *Cloud Computing (CLOUD), 2012 IEEE 5th International Conference on*, pages 991 –992, June 2012.

[18] Xiao Su, Schlinker B., Yi Shang, and Zhenzhen Ye. Optimal media storage in federated cloud environments. *Communications (ICC), 2012 IEEE International Conference on*, pages 5518 –5528, June 2012.

[19] Kannan S., Gavrilovska A., and Schwan K. Cloud4Home – Enhancing Data Services with @Home Clouds. *Distributed Computing Systems (ICDCS), 2011 31st International Conference on*, pages 539 –548, June 2011.

[20] Rajkumar Kettimuthu, Robert Schuler, David Keator, Martin Feller, Dingying Wei, Michael Link, John Bresnahan, Lee Liming, Joseph Ames, Ann Chervenak, Ian Foster, and Carl Kesselman. A Data Management Framework for Distributed Biomedical Research Environments. *E-SCIENCEW '10 Proceedings of the 2010 Sixth IEEE International Conference on e-Science Workshops*, pages 72 –79, 2010.

[21] Chang G., Shams K., Callas J., and Kern A. A novel approach to automated, secure, reliable, & distributed backup of MER tactical data on clouds. *Aerospace Conference, 2012 IEEE*, pages 1 –7, March 2012.

[22] Winkler S. and Mohandas P. The Evolution of Video Quality Measurement: From PSNR to Hybrid Metrics. *Broadcasting, IEEE Transactions on*, 54(3):660 –668, 2008.

[23] Srikanth Sundaresan, Walter de Donato, Nick Feamster, Renata Teixeira, Sam Crawford, and Antonio Pescapę. Broadband internet performance: a view from the gateway. *SIGCOMM '11 Proceedings of the ACM SIGCOMM 2011 conference*, pages 134 – 145, 2011.

[24] Test Methodology White Paper. `http://www.samknows.com/broadband/uploads/Methodology_White_Paper_20110701.pdf`.

[25] Luca De Cicco, Gaetano Carlucci, and Saverio Mascolo. Experimental Investigation of the Google Congestion Control for Real-Time Flows. *FhMN '13 Proceedings of the 2013 ACM SIGCOMM workshop on Future human-centric multimedia networking*, pages 21 –26, March 2013.

[26] GridFTP. `http://toolkit.globus.org/toolkit/docs/latest-stable/gridftp/`.

[27] Google Cloud Storage: Store Your Data in Google's Cloud. `https://developers.google.com/storage/`.

[28] Amazon S3. `http://aws.amazon.com/s3/`.

[29] Durable Reduced Availability Storage. `https://developers.google.com/storage/docs/durable-reduced-availability`.

[30] F. Audet, Ed., and C. Jennings. Network Address Translation (NAT) Behavioral Requirements for Unicast UDP. RFC 4787 (Best Current Practice), January 2007.

[31] J. Postel. Transmission Control Protocol. RFC 793 (Standard), 1981. Updated by RFCs 1122, 3168.

[32] P. Srisuresh, B. Ford, and D. Kegel. State of Peer-to-Peer (P2P) Communication across Network Address Translators (NATs). RFC 5128 (Informational), 2008.

[33] Dropcam. `https://www.dropcam.com/`.

[34] J. Postel and J. Reynolds. FILE TRANSFER PROTOCOL. RFC 959 (Standard), 1985.

[35] D. Mitton and M. Beadles. Network Access Server Requirements Next Generation (NASREQNG) NAS Model. RFC 2881 (Informational), 2000.

[36] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Berners-Lee. Hypertext Transfer Protocol – HTTP/1.1. RFC 2616 (Standard), 1999.

[37] H. Schulzrinne, A. Rao, and R. Lanphier. Real Time Streaming Protocol (RTSP). RFC 2326 (Standard), 1998.

[38] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RTP: A Transport Protocol for Real-Time Applications. RFC 3550 (Standard), 2003.

[39] J. Postel. User Datagram Protocol. RFC 768 (Standard), 1980.

[40] Python Pyramid. `http://www.pylonsproject.org/`.

[41] AVC/H.264. `http://en.wikipedia.org/wiki/H.264/MPEG-4_AVC`.

[42] G.711. `http://en.wikipedia.org/wiki/G.711`.

[43] GpENI: Great Plains Environment for Network Innovation. `https://wiki.ittc.ku.edu/gpeni/Main_Page`.

[44] Telehousecalls. https://www.telehousecalls.org/.

[45] Globus. https://www.globus.org/.

[46] gsutil Tool. https://developers.google.com/storage/docs/gsutil.

[47] boto: A Python interface to Amazon Web Services. https://boto.readthedocs.org/en/latest/.

[48] GNU Wget. http://www.gnu.org/software/wget/.

[49] Apogee. http://www.apogee.us/.

[50] tshark. https://www.wireshark.org/docs/man-pages/tshark.html.