ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 3, March 2016

Design and Analysis Scalable and Interactive Video-On-Demand System

Patel Urmilaben S.[1], Sugnesh Hirpara[2]

PG Student, Dept. of Electronics & Communication, SOCET, Ahmedabad, Gujarat, India[1]

Asst. Professor, Dept. of Electronics & Communication, SOCET, Ahmeedabad, Gujarat, India[2]

ABSTRACT: This paper represents a design & analysis scalable and interactive video–on–demand system". It makes use of ARM11 processor & Linux operating system as platform & develops a real-time streaming server through implementation of RTSP protocol. The development of web page to clients can easily connect the web page for video-on-demand system. The real time video streaming is encoded in H.264 video coding standards. It makes uses of ARM 11 embedded processor because low power consumption and high performance. We describe a video delivery system using local /internet Wi-Fi connection as connection mechanism for on demand video playback and both real time streaming. The pre-recorded video stored to the server side and user requirement to send the video of the clients and client watch the video live.

KEYWORDS: Media management system, RTSP, local/internet WI-FI, Streaming media technology, Video-on-demand system.

I. INTRODUCTION

Single server to multiple clients delivers of the streaming. Three options available for delivered of the streaming like USB insert, wireless connection and LAN. Network data transfer from one server to multiple clients to adjust is same network. Same network means one topology. One topology inside each and every identified have to one server to the multiple clients to be sate of same network and then data transmit from server to the multiple clients.

Video signal is separated into picture is called frames. Each image is transmitted over the transport interface from top to bottom. So, the image starts with left corner pixel and ends with the right corner pixel. The video signal convert with image and calculating, the speed is called a frame rate [2]. This enables each frame to be crowned to the same size enabling each slice to be transmitted immediately in a single UDP or TCP packet and on arrival immediately decoded.

The streaming media protocols which include SIP, RTP and RTSP provide the service of connecting the client and the server as well as the multimedia data transmission between them.SIP is a client/server protocol using the text mode which is intended to facilitate the development of new IP-based services. The H.264 video coding technology is used for encoding the video images of the real time video streaming. The H.264 standard is widely used in various fields for its high efficiency of the compress technology of coding and better adaptability on network [3].

The real time multimedia can be streamed to wireless handheld devices, for e.g., it is used to distribute the digitally captured On-Campus live events such as Businesses, Education, Media companies , Homes , Company , Conference , lectures , rules and regulations of university to new attendees and other events of general interest in real-time. Additional feature of video-on-demand also provides by the server, where the users can video start when he wants, back, go forward, and make pauses in the video at his desire. Since the ability to access information at one's convenience is a desired feature of any multimedia distribution system, video-on-demand is of course the best in video streaming and is a dream of every user [1].

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 3, March 2016

II. RELATED WORK

The implementation of a video network streaming system implementation on a raspberry pi board. The live streaming video content from the single server to multiple clients. The stored video delivered to the single server through to the multiple clients, using RTSP protocol. The test results prove that it completely satisfies the embedded system user's demand and performs well [1]. The raspberry pi of capabilities over arduino and advantage of the pibot more than the predictable observation system. Also by mean it the ability to detect and recognize faces it be able to finished to on the alert us about every unknown person and obtain a snap of it and email us the same. Live video streaming application survey on arm 11 board [3]. The surveillance system uses H.264 video coding standard to encode the video data in order to lower the bit rate to improve the network adaptability. The surveillance system there excellent video quality and give of the network condition [4].

III. DEFINITION OF RELATED CONCEPTS

A. ARM11 processor (Raspberry pi)

The Raspberry Pi is based on the Broadcom BCM2835 system on a chip (SoC), which includes an ARM1176JZF-S 700 MHz processor, Video Core IV GPU, and was originally shipped with 256 megabytes of RAM, later upgraded (Model B & Model B+) to 512 MB. The system has Micro SD (Model B+) sockets for boot media and persistent storage. The Raspberry Pi chip, operating at 700 MHz by default, will not become hot enough to need a heat sink or special cooling The video controller is capable of the following video resolutions: 640 × 480 VGA; 1280×720 720p HDTV[1].

B. Streaming protocols

a. RTSP

The Real Time Streaming Protocol (RTSP) is used to control the playback of an RTP stream. An RTSP Server allows a viewer to receive the RTP packets sent from the device. A viewer (e.g. VLC) cannot directly open an RTP stream, therefore an RTSP server is required to provide additional stream information and control the starting and stopping of the streaming packets. It incorporates the following commands: options describe, announce, setup, play, pause and teardown. The Session Description Protocol, or SDP, based on RFC 4566 is used in conjunction with the describe command. However, the operating system or browser will typically need to have the decoder installed for the type of video being transmitted. H.264 is currently the most common streams for videos using RTP.

The Real-Time Streaming Protocol (RTSP) establishes and controls either a single or several time-synchronized streams of continuous media such as audio and video. It does not typically deliver the continuous streams itself, although interleaving of the continuous media stream with the control stream is possible. It is an application layer protocol used to control streams of data. RTSP uses RTP as transport protocol. RTSP has the following properties: Extendable, easy to parse, secure , transport-independent, multi-server capable, control of recording devices, Separation of stream control and conference initiation, Suitable for professional applications, Proxy and firewall friendly, HTTP-friendly etc [1].

- An example of such a client request using OPTIONS is as follows (assuming port number of 554):
  command for connecting to server from windows or Linux
  rtsp://192.168.1.101:8554/pi_encode.h264 note: [replace your IP with given.]

b. RTP

RTP is Real Time Transport protocol also known as RTTP (Real time transport protocol). RTP is designed for end-to-end, real-time, transfer of streaming media. It provides services like time reconstruction, loss detection, and content identification. RTP supports one way communication i.e., from server to client [1].

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 3, March 2016

c. **SDP**

The Session Description Protocol (SDP). SDP is intended for describing multimedia sessions for the purposes of session announcement, session invitation, and other forms of multimedia session initiation. When initiating multimedia teleconferences, streaming video, or other sessions, there is a requirement to convey media details, transport addresses, and other session description metadata to the participants. SDP provides a standard representation for such information, irrespective of how that information is transported. SDP is purely a format for session description it does not incorporate a transport protocol, and it is intended to use different transport protocols as appropriate, including the Session Announcement Protocol Session Initiation Protocol, Real Time Streaming Protocol. SDP is intended to be general purpose so that it can be used in a wide range of network environments and applications. However, it is not intended to support negotiation of session content or media encodings: this is viewed as outside the scope of session description [1].

IV. HARDWARE DESIGN

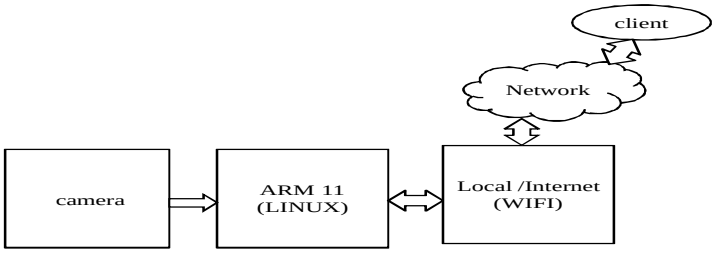

Figure1: Block diagram of the hardware system design

Video streaming system consists of five blocks: Camera, ARM 11(Linux), local/internet (WIFI), network, and client
Camera: It is used to capture videos to the server side. There is a plug with in ARM 11 used as extension to connect the camera. This camera continuously monitors the class room, and sends the video of the client.
ARM 11 (raspberry pi): Real time video streaming client it is used for compression and processing of videos. Streaming Media server allows you to send media to clients across the local/internet WIFI through the streaming media protocol. It enables the pre-recorded of video content from a single server to a large number of clients. The ARM 11 processor & Linux operating system as a platform & develops a server through implement of RTSP protocol.
Local/internet (WI-FI): most portable computers and smart phones have some type of inbuilt WI-FI network interface cards. Such development of WI-FI networks allows the user to convert the video distribution system to wireless network [1]. A local area network is a computer network that interconnects computers within a limited area such as a school, laboratory, or office building.

IV. FLOW CHART OF REAL TIME VIDEO STREAMING

Programming was done strictly within the C language. The system selected Linux operating system as software platform, use embedded Linux 2.6 kernel [5]. The software controls the operation of the entire system, in structuring the camera, compression and processing of video, and transmitting it to clients using transport level and application protocols. There are mainly three function modules, that is, Video capture module, Video Compression module, and Video Streaming module. The flow chart system of Real time video streaming shown in Figure 2.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 3, March 2016

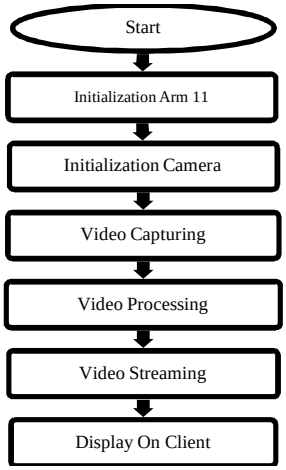

Figure 2: Flow chart of real time video streaming

A. Video capture module

Video Capture Based Video4Linux (referred to as "V4L") is a Linux kernel on the video device driver, which is for video equipment, application programming interface functions to provide a system. V4L pi camera using the programming on the need to use Linux system. Programming was done strictly within the C language [1].

B. Video Processing

a. Processes of the real time video streaming

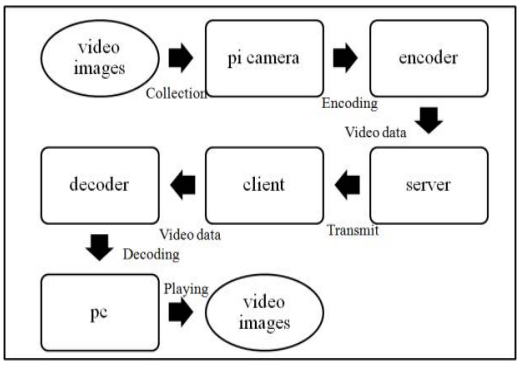

Figure 3: processes of the real time video streaming

The server side is composed of the pi camera, encoder and the server. The client connects with the server and receives the video data by local/internet WIFI. The encoder will encode the video images collected by the pi camera with the H.264 video coding standards. The multimedia data encoded by the encoder will be transmitted to the server by local/internet WIFI and the client of the real time video streaming will receive the multimedia data of the client [4].

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 3, March 2016

b. **Processes of the communication between client and server**

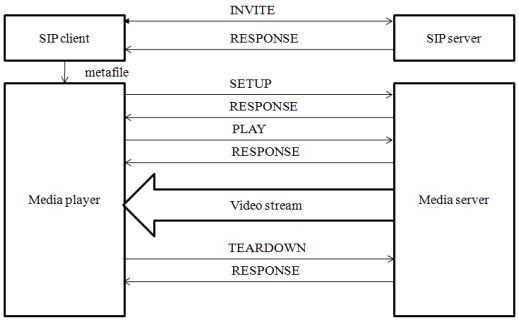

Figure 4: processes of the communication between client and server

The communication between the client and the server is mainly composed of two parts, the first part is the connection established between the server and the client through the SIP protocol. Client sends the INVITE message to request video files from the server side. Then the SIP server wills response to the client and sends the metafile to the SIP client. The metafile contains the address of the multimedia data which is needed for the media server to transmit data [4].

In the second part of the system, the multimedia data is transmitted. The SIP client transfers the metafile to the media player to request the video data. The RTSP client of the media player will send SETUP message to the RTSP server of the media server to establish a connection. After receiving the response from the media server, the RTSP client will request the video data by sending PLAY message relying on the address of the video data in the metafile. The media server uses RTP/RTCP protocol to send the video stream to the media player. After the media player receives the video data, it will decode the data and play it on the client side. The RTSP client can send the TEARDOWN message to disconnect with the media server [4].

C. Video Streaming Server

Pi camera is connected to the ARM11 board. Make Mincom-s settings in the terminal window, during the settings we run the application related shell script in terminal which will execute application in board resulting video streaming on web browser using RTSP protocol, entering a static IP address by user in any wireless device which is in local network can view the remote location. The server periodically obtain videos from camera through the private network, such videos are transmitted from camera to the server [1].

V. EXPERIMENTAL RESULTS AND ANALYSIS

A. EXPERIMENTAL RESULTS

Setting up and about the camera module as well as then it connect with the ARM-11 board. The connecter for the camera is the one close to the Ethernet as well as HDMI ports; this is able to be avoided by touching an exposed earth connection.

The **Enable Camera** selection, after that verify the selection by choose **enable** within the dialog that appear. At this time exit the tool with select the **Finish** option as of the main menu. In the dialog that appears ask to reboot choose the **Yes** selection therefore the Pi reboots.

Figure 5: pi camera capture the image

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 3, March 2016

Later than the Pi reboots log reverse within also experiment so as to the camera works by using the rasp still command. Run the following command:

raspistill -t 5000 -o image.png

- Using **raspistill** command will produce a capture every 5 seconds.
- Check the following command to actual camera working that show the image in pc

We can also use the following command to save a still image as a **png** file. Here, the –t parameter controls the amount of time in milliseconds between the camera preview starting and the image being captured.

The input raw video file is captured using a pi camera interfaced to ARM 11. The video file size 31MB and frame per second is 30fps.The live streaming video of the real time streaming (based on H.264, SIP, RTP and RTSP) on the client side. The test is conducted in the class room. The pi camera and the H.264 encoder connect with the server of the network (connectivity hotspot software).

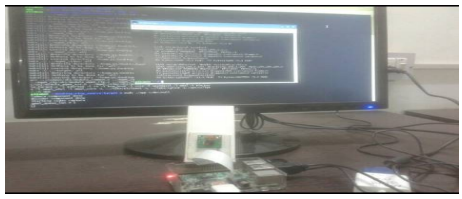

Figure 6: server continuously capture the video

The server of the real time video streaming transmit the video data received from the pi camera and the H.264 encoder to the client of the system by network. The client of the real time streaming show the video decoded on the laptop.

- command for connecting to server from windows or Linux
- **rtsp://192.168.1.101:8554/pi_encode.h264**
- Note: [replace your IP with given.]

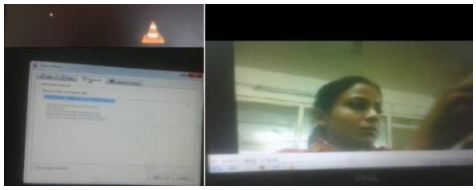

Figure 7: the real time streaming from server to the clients

B. Analysis of the experimental results

The experiment results show that the real time streaming (video-on-demand) provides good video quality. The frame rate of the video is 30 fps and the network bandwidth of the LAN is 2M.The bit rate of the video 30 Mbps.

As the video data is encoded with H.264 video coding standard, the bit rate of the data 30Mbps.The streaming media protocols including SDP,RTP, RTSP and make transmission of the video and make the functions of pausing playing, forward the video latency (5 second), and less convenient.

VI. Conclusion And Future Work

This paper describes the design and analysis scalable and interactive video-on-demand system. H.264 video coding standard to encode the video data in order to the bit rate improve the network adaptability. The system realizes the connection and authentication between client and the server with the SDP protocol and also realizes the real time transmission, playing and playback of the video with RTP and RTSP protocol. The pre-recorded videos

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 3, March 2016

watch the client using RTSP protocol. The RTSP protocol implement of the raspberry pi B+ model then realize the client watch the video using RTSP protocol then latency generated minimum 5second.

This web application to develop and create environment to client can easily connect the web page and easily watch the video for video-on-demand system. The new library design for using (RTSP, RTP, RTCP) protocol. This library use to client easily watches the video for client requirement to server side.

REFERENCES

1. Rupalwala, S.S. , "ARM 11based real-time video streaming server using RTSP protocol", Electrical, Electronics, Signals, Communication and Optimization (EESCO), 2015 International Conference on, IEEE 2015, 24-25 Jan., ISBN-978-1-4799-7676-8 , pp 1-5
2. T.Bernatin , G.Sundari "Video compression based on Hybrid transform and quantization with Huffman coding for video codec" International conference on control, instrumentation, communication and computational technologies (ICCICCT), pp 476-480,July - 2014,IEEE.
3. Ikhankar, R.; Kuthe, V.; Ulabhaje, S.; Balpande, S.; Dhadwe, M. ,"Pibot:The Raspberry Pi Controlled Multi-Environment Robot For Surveillance & Live Streaming", International Conference on Industrial Instrumentation and Control (ICIC), IEEE 2015, May 28-30, Pages 1402-1405
4. Dian Chu; Chun-hua Jiang; Zong-bo Hao; Wei Jiang "The Design and Implementation Of Video Surveillance System Based on H.264, SIP, RTP/RTCP and RTSP", Computational Intelligence and Design (ISCID), 2013 Sixth International Symposium on, IEEE 2013, Volume-2, 28-29 Oct., pp 39-43
5. Kavitha mamindla, Dr V. Padmaja, "Embedded real time video monitoring system using ARM", IOSR July 2013.