IJETST- Volume||01||Issue||04||Pages 484-491||June||ISSN 2348-9480 2014

Open access Journal International journal of Emerging Trends in Science and Technology

Routing of Video/Audio Multimedia Live Stream over Wireless Mobile Network

Authors
[1]Ujwala C, [2]Sankar Dasiga
[1]M.Tech Student,Department of ECE, Nitte Meenakshi Institute of Technology
Bangalore, Karnataka, India.
[2]Professor,Department of ECE, Nitte Meenakshi Institute of Technology
Bangalore, Karnataka,India.
[1]ujwala.engg@gmail.com, [2]dasiga@hotmail.com

Abstract

The rapid adoption of smart phones has created a unique opportunity for mobile multimedia services for mobile users. Currently a majority of smart phones are equipped with both hardware that supports real-time video processing and ad-hoc wireless communication between peers and this allows real-time video streaming over multiple wireless hops between peer devices. Phones within communication range of each other automatically establish a wireless link creating a client mesh network (ad-hoc network of devices). Each phone in the client mesh network is able to produce/consume video and also acts as a relay to forward video to its next hop neighbors. Peer-to-peer video streaming from the cameras on smart phones to people nearby allows users to share what they see. Such streaming can be used in a variety of applications.

Index Terms: Multimedia processing,Video Streaming,H.264,Android,MultiHopRouting,WiFi.

I INTRODUCTION

The International Telecommunication Union's (ITU) statistics on mobile subscriptions indicates five billion mobile subscriptions for 2010 [1], with a 17% penetration of smartphones in 2009 [2]. The rapid adoption of smartphones has created a unique opportunity for mobile multimedia services for mobile users. Currently a majority of smartphones are equipped with both hardware that supports real-time video processing and Ad-hoc wireless communication between peers and this allows real-time video streaming over multiple wireless hops between peer devices. Phones within communication range of each other automatically establish a wireless link creating a client mesh network (ad-hoc network of devices). Each phone in the client mesh network is able to produce/consume video and also acts as a relay to forward video to its next hop neighbours. Peer-to-peer video streaming from the cameras on smartphones to people nearby allows users to share what they see. Such streaming can be used in a variety of applications, in particular in various social network applications including sharing unforgettable moments with friends that can be multiple wireless hops away, cooperative fieldwork (providing video sharing for teams distributed in a small area, e,g. teams of repairmen, and search and rescue teams in disaster areas), and support for health impaired persons including the elderly.

There exist solutions that provide video services for mobile devices. In [3], Cycon et. al. present a peer-to-peer videoconferencing application with modified H.264 video codec for mobile phones. However, the use of a customised codec and development library poses an issue on portability to other mobile devices for wider deployment. In addition, the paper does not present results to demonstrate the feasibility in an actual network deployment scenario. Qik [4] is one of the best known real-time video streaming services, which has support for a number of mobile devices. It adopts the client-server architecture; that is, mobile phones (with Qik as the client) stream live video to centralised processing servers using the available network infrastructure (such as cellular or WiFi infrastructure networks). Video sharing is then done over the Internet, placing great dependency on the network infrastructure. Peer-to-peer live video streaming is currently not supported by Qik.

The Figure1 shows a Peer to peer (P2P) technology mobile user, the ability to share real time video that they are watching or transmitting to other authorized users.

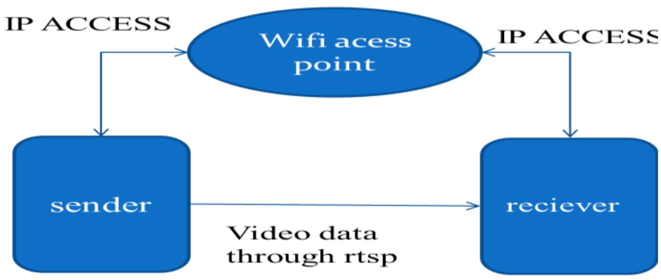


Fig 1: Peer-to-peer sharing

The two mobile phones need to be connected to wifi network not necessarily external wifi router for acquiring IP address. After connection established, real time protocol data can be transmitted from sender to receiver.

II THE BUILDING BLOCKS:OVERVIEW

A) Overview of Android System

Android is an operating system for mobile devices that are developed by the Open Handset Alliance. User applications are mostly written in Java and run on Android's own Java virtual machine (named Dalvik).

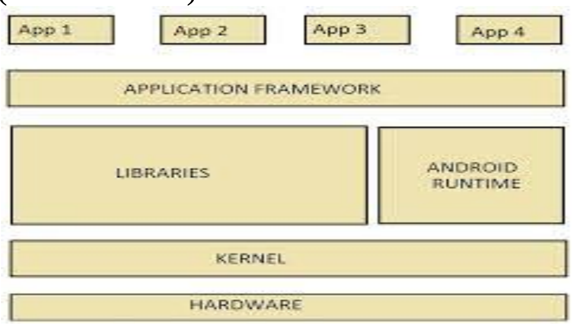


Fig 2: Android System Architecture

Figure2 shows the Android system architecture, which consists of the Linux kernel with device drivers and the Android runtime environment (along with a number of libraries) that support interactions between the Linux kernel and the high-level application framework. The application framework released in a bundle as the Android SDK [5] provides high-level Java interfaces for accessing the underlying resources, such as camera and WiFi. For example, our video streaming application makes use of the activity manager to detect and respond to events when triggered. The use of standard development toolkit encourages interoperability between components and maximises portability of the application.

In addition to the SDK, there is also a Native Development toolkit (NDK), which supports the use of native C or C++ codes in the applications. The NDK is an extension of the SDK to allow the development of lower-level source codes for more efficient data processing in the system. In our application, parts of the video encoding and decoding are implemented using the NDK to allow more efficient processing of video streams.

B) Codecs and Method of Video Encoding

Video encoding and decoding is an important aspect of any video streaming application. There are many ways by which a video can be encoded or decoded. We briefly describe two widely used video coding techniques which are implemented in our application.

• *Intraframe encoding* is the simplest form of encoding. It treats every frame as an individual image to encode. This method is resilient against lost frames due to each frame having enough information to create an entire image.

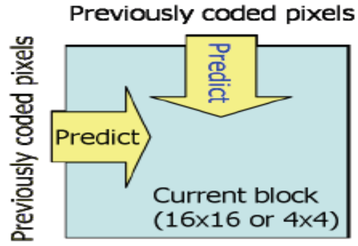


Fig 3: Intraframe Encoding.

Intra prediction uses 16x16 and 4x4 block sizes to predict the macro block from surrounding, previously-coded pixels within the same frame as shown in Figure3.

• *Interframe encoding* uses two types of frames, i.e., the key frames and predicted frames, for better compression ratio. The key frame contains complete information to create an image, whereas the predicted frames only contain the differences between frames thus previous frames are required for their successful decoding.

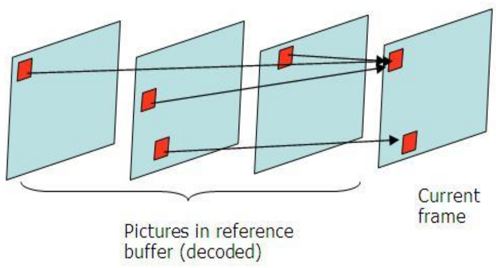


Fig 4: Interframe Encoding

Inter prediction uses a range of block sizes (from 16x16 down to 4x4) to predict pixels in the current frame from similar regions in previously-coded frames as shown in Figure4.

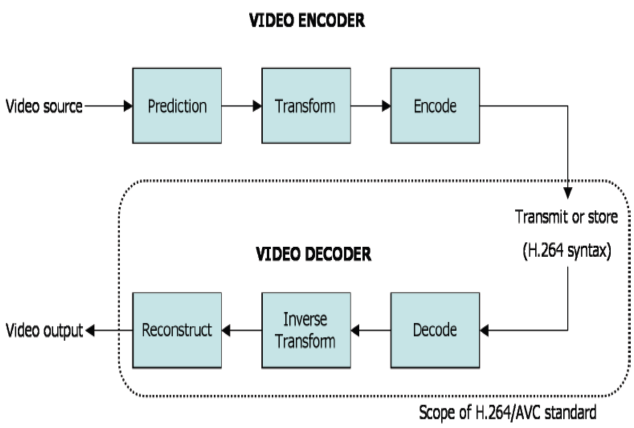


Fig 5: Video Streaming using H.264 encoding and decoding

H.264 is an industry standard for video compression, the process of converting digital video into a format that takes up less capacity when it is stored or transmitted.
Figure5 shows Video Streaming using H.264 encoding and decoding.An encoder converts video into a compressed format and a decoder converts compressed video back into an uncompressed format.
Recommendation H.264: Advanced Video Coding is a document published by the international standards bodies ITU-T (International Telecommunication Union) and ISO/IEC (International Organization for Standardization / International Electro technical Commission). It defines a format (syntax) for compressed video and a method for decoding this syntax to produce a displayable video sequence.

III DESIGN AND IMPLEMENTATION

A) System Architecture

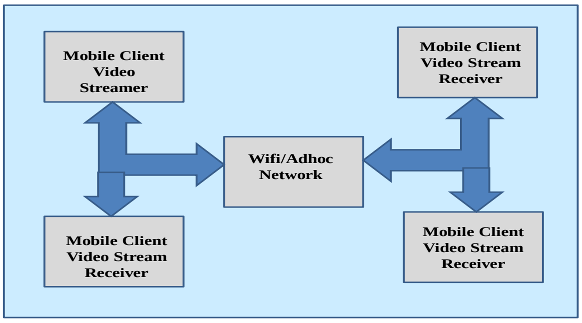


Fig 6: Architecture Of Project

The Mobile Application is broken down into two major subsystems, the mobile client we present a wireless multi-hop video streaming application for mobile phones with the Android operating system. This application allows sharing live information captured by mobile phone sensors (e.g., camera, microphone) with persons that might be multiple wireless hops away. The video streaming is based on peer-to-peer communication between mobile phones, i.e. without the use of video processing servers or network infrastructure. We show the feasibility of such peer to-peer video streaming application for Android phones in a variety of experiments that evaluate various video streaming Scenarios, including various video codec's and various generations of Android phones.

B) *Video En/Decoding*

Video frames need to be encoded before sending on to network. Encoding can significantly reduce the size of individual video frames, therefore minimising the bandwidth required for the streaming application. Figure7 shows the procedures of encoding on the sender and decoding on the receiver.

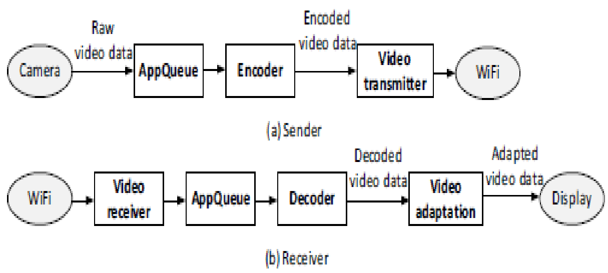


Fig 7: Decomposition description

As shown in Figure7, when starting the streaming application raw video data is retrieved from the camera and stored in the APPLICATION QUEUE (for buffering purpose). This raw data is then passed to the encoder, which encodes the data using the selected codec and encoding technique. The encoded video frame is then transmitted over-the-air by the WiFi module.
At the receiver, when an encoded video frame arrives it is buffered and sent to the decoder. Before being displayed on the screen, the video frames may require adaptation to the hardware specifications (e.g., screen resolution).

C) Routing

Real-time Transport Protocol (RTP) defines a standardized packet format for delivering audio and video over IP networks. RTP is used extensively in communication and entertainment systems that involve streaming medi a, such as telephony, video teleconference applications, television services and web-based push-to-talk features.

RTP is one of the technical foundations of Voice over IP and in this context is often used in conjunction with a signaling protocol which assists in setting up connections across the network. The main processes are identified as under:

- Adhoc net work /Wi-Fi setting need to be done with IP/Port.
- We need to establish the connection between android mobile user and Receiver User.
- When application initiate , User has to launch request and start streaming.
- It will verify camera is connected and then after process Raw Video.
- It Uses H.264 Encoder/Decoder for this process.
- Once the streaming start, video will process in encoding with the help of H.264encoder.
- Once Encoding is done and it will go in network with the help of RTSP/RTP.
- Receiver will receive the video and display on their device after the decoding.

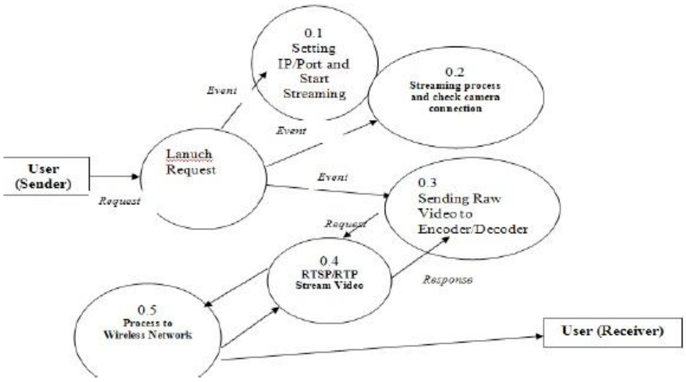


Fig 8:RTSP/RTP PROCESS

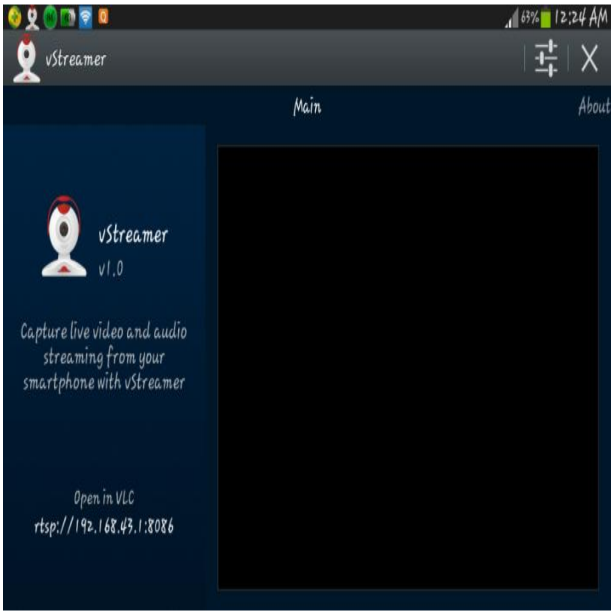
To stream a video across multiple hops, a multi-hop path between the sender and receiver has to be found. A routing protocol is therefore required to establish and manage connectivity within the mobile ad hoc network. A routing protocol may form various logical network topologies depending on the selected metrics (such as link quality) and QoS requirements.
In our prototype implementation, the peer-to-peer video streaming application relies on a network layer routing protocol (such as OLSR [6]) to carry out the routing tasks. information that the routing protocol can provide to the application layer (e.g. link quality) can be used by our video streaming application to adapt the video to the current context of the network, similar to our work presented in [7]. However such adaptation is out of scope for this paper as we concentrate here on the feasibility of using various generations of Android phones for multi-hop infrastructureless video streaming.

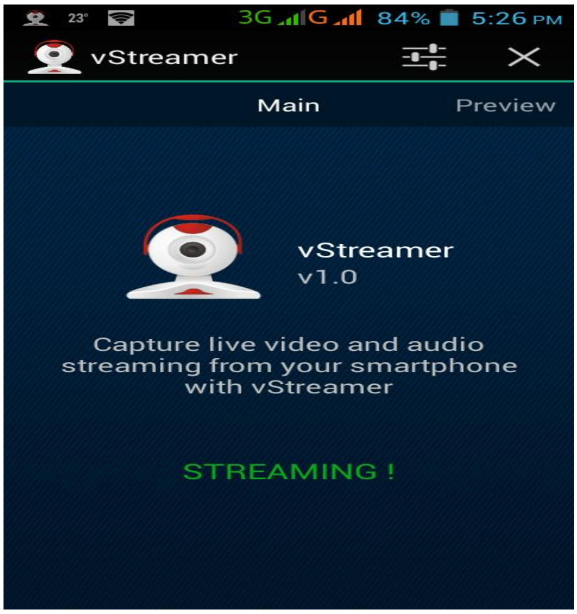
IV EXPERIMENTAL RESULT

The live video service is provided using two parameters in general

1.IP address.
2.Port number

Open the video streamer application and once the application is opened,in the main we can see the IP address and Port number along with the Protocol.

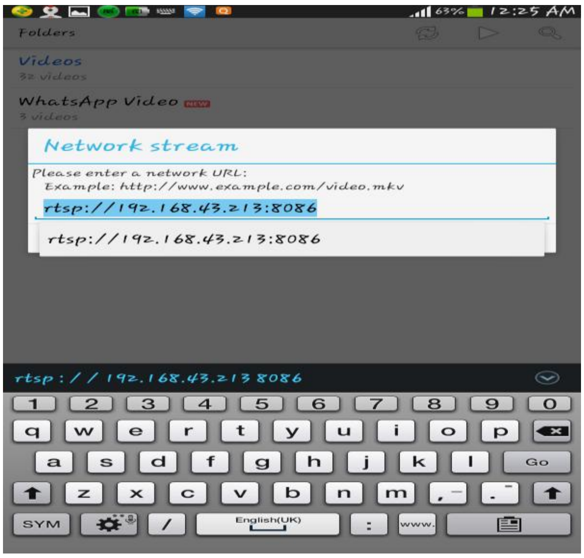




In order to play a video open the MX player and select an option called network stream and provide the available IP address with Port number along with the Protocol and the format is

rtsp://192.168.1.43:8086

Protocol IP Address Port Number



When once the available IP adress and Port number along with the Protocol is provided it starts streaming and thus live video streaming can be enjoyed.

V TEST ANALYSIS

To demonstrate the feasibility of the video streaming application on Android phones. We selected mobile phones represent three generations of mobile devices on the consumer market. As shown in Figure8, these include Micromax A74,Samsung Galaxy tab3 and LG cookie Smart T375.

(a) 

 (b)

 (c)

(a)Micromax A74
(b)Samsung Galaxy Tab3
(c)LG Optimus Smart T375

Fig 9: Experiment devices

Table 1: Device Specification

	Micromax A 74	Samsung GALAXY TAB3	LG OPTIMUS
Processor	1.3 GHz	1.2 GHz	1GHz
Memory	512 MB RAM	1 GB RAM	512 MB RAM
Connectivity	802.11 b/g/n	802.11 a/b/g/n	802.11 a/b/g/n
OS version	Android v4.2.2 (Jelly Bean)	Android 4.1 (Jelly Bean)	Android v4.1 (Jelly Bean)

As shown in Table 1, each generation of mobile phone is equipped with significantly different hardware resources. Based on understanding the differences in the hardware specifications we decided on the way in which the experimental test bed should be set up.

We deploy the application on each of these phones. Figure10 shows a screen shot of the live view of the application and its menu bar.

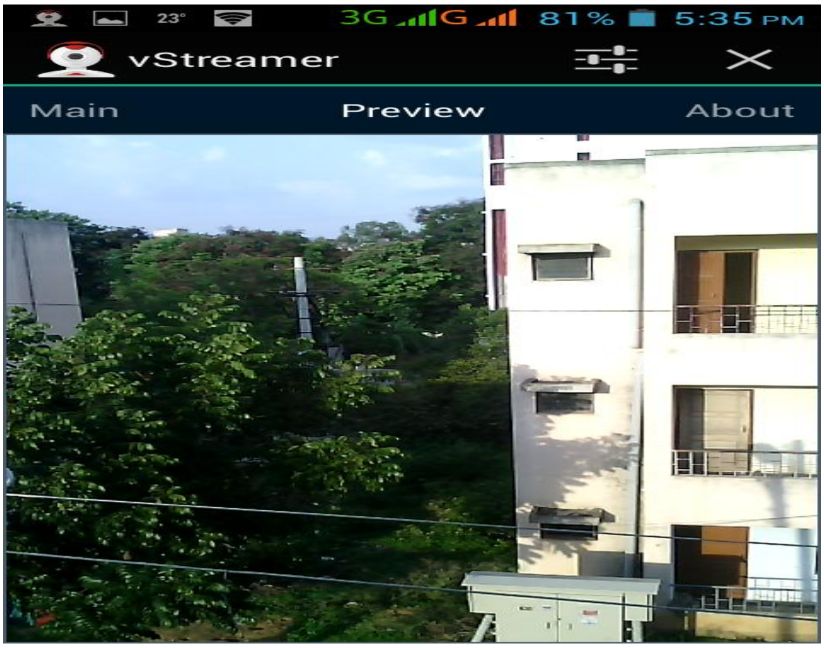


Fig 10: Screenshot of the Application

A)Performance

TABLE 2: performance of streaming in 10 meter distance

Device name	First view delay(sec)	Sound lag(sec)
(a)1.3 GHz dual-core processor	6.57	2.32
(b) 1.2 GHz dual-core processor	5.32	1.54
(c)1GHz	4.54	2.0

Table 2 shows the first view delay and sound lag when the video streamer application runs on different processors with a bit rate of 700kbps ,frame rate of 20fps and resolution of 640x480 within a distance of 10 meters.

TABLE 3: performance of streaming in 20 meter distance

Device name	First view delay(sec)	Sound lag(sec)
(a)1.3 GHz dual-core processor	7.85	3.37
(b) 1.2 GHz dual-core processor	6.4	2.69
(c)1GHz	6.66	2.80

Table 3 shows the first view delay and sound lag when the video streamer application runs on different processors with a bit rate of 700kbps ,frame rate of 20fps and resolution of 640x480 within a distance of 20 meters.

VI CONCLUSIONS

In this paper, we presented a wireless multi-hop video streaming application for Android mobile phones. This application allows users to capture live video feeds using the mobile phone camera and to share these feeds with people who might be multiple wireless hops away. The video feeds are shared using wireless client mesh network (ad hoc network) established between mobile phones. Thus the video streaming does not rely on a traditional network infrastructure (such as the cellular), therefore it is a free-of-charge communication. Such a multi-hop video streaming can be used in a variety of application domains including social networking. We presented an evaluation of the prototype application and demonstrated the feasibility of the multi-hop video on three generations of Android phones (with different resource capabilities).

REFERENCES

1. ITU. Statistics on global mobile subscriptions. http://www.itu.int/newsroom/press_releases/2010/06.html.
2. C. Quick. (2009) With smartphone adoption on the rise, opportunity for marketers is calling. http://blog.nielsen.com/nielsenwire/onlinemobile/with-smartphoneadoption-on-the-rise-opportunity-for-marketers-is-calling/.
3. H. L. Cycon, T. C. Schmidt, G. Hege, M. Wahlisch, D. Marpe, and M. Palkow, "Peer-to-peer videoconferencing with h.264 software codec for mobiles," in *Proc. Int. Symp. a World of Wireless, Mobile and Multimedia Networks WoWMoM 2008*, 2008, pp. 1–6.
4. Qik. www.qik.com.
5. Android development sdk. http://developer.android.com.
6. Optimized link state routing protocol (olsr).
   a. http://www.ietf.org/rfc/rfc3626.txt.
7. P. Hu, W. L. Tan, R. Wishart, M. Portmann, and J. Indulska, "Meshvision:an adaptive wireless mesh network video surveillance system," *Multimedia Systems*, vol. 16, pp. 243–254, 2010, 10.1007/s00530-010-0191-z.
8. Q. Huynh-Thu and M. Ghanbari, "Temporal aspect of perceived quality in mobile video broadcasting," *Broadcasting, IEEE Transactions on DOI-10.1109/TBC.2008.2001246*, vol. 54, no. 3, pp. 641–651, 2008.787

BIOGRAPHIES

UJWALA C
M.Tech, Digital communication and Networking, NMIT, Bangalore.
BE,Electronics and Communication, SIET, Tumkur
Email: ujwala.engg@gmail.com

Prof. SANKAR DASIGA
M.Tech, Electronic Instrumentation
Department.of ECE
NMIT,BANGALORE.

Experience-Program Manager
NXP Semiconductors India Pvt Ltd.
Manager - Software Development
Cisco Systems India Pvt Ltd
Technical Manager
Wipro Infotech Ltd
Project Leader
Philips Software Centre
Manager / Consultant
Siemens Information Systems Ltd

Email:dasiga@hotmail.com