The Journal of Social Sciences Studies and Research
Online ISSN: 2583-0457
Available Online at http://www.tjsssr.com Volume02|Issue 06 (November-December)|2022|Page:242-249

Original Research Paper

Smartphone Based Real Time Content Sharing in Opportunistic Vehicle Networks

Corresponding Author: Er. Manish Singh Rawal, Software Quality Assurance Engineer
Tekvortex Pvt. Ltd , Lalitpur
rawalsinghmanish@gmail.com

Article Received: 25-09-2022, Revised: 15-10-2022, Accepted: 04-11-2022

ABSTRACT:
Vehicle Network is a different network dedicated for communication between vehicles. They communicate with each by sharing the information they have while driving in the road. Everyone carries a smartphone with them and can be used as device to transfer data between the nearby vehicles in their range of communication by using the available network opportunistically. Opportunistic network can enhance the user experience by utilizing nearby wireless links with more bandwidth instead of just depending upon mobile data infrastructure and reduce the higher cost related to mobile data services. In this research, Wi-Fi direct has been used as the communication tool between the smartphones to transfer real time videos. These videos can be viewed by client vehicles to gather information about the different road conditions ahead of them and with this knowledge, prevent fewer accidents in the road and lessen the traffic in particular road. The system consists of smartphones working as server and client whom are connected to a hotspot and use Wi-Fi Direct to send real time videos at same time to each other. TCP has been used as the protocol because of its reliability factor. The smartphone is capable of displaying the both captured video and received video simultaneously at the same time. For doing that, two kinds of activity have been used. The Camera Activity to display the video taken from the local camera and fragment activity to display the received video. The MediaCodec API of the android has been used for encoding and decoding the video data captured from smartphone in H.264 format.

Keywords: Wi-Fi-Direct, TCP, API, Real time, H.264, Vehicle Network, Media Codec

INTRODUCTION:

Main purpose of this research is safety of people driving the vehicles and people on the road. Help drivers communicate about road conditions ahead. If drivers are aware, the probability of them also having the same accidents decreases as they are more aware. For example, there is snowing is roads ahead this information can be relayed to drivers on the same road but yet to come to a snowy road. This can help them drive slowly and more carefully. A key enabler of this revolution has been availability of communication protocols: from conventional Wi-Fi and 2g/3g/4g networks to new emerging, connection-less protocols such as Wi-Fi-direct and LTE-direct. Wi-Fi direct offers lower-level network abstraction .it is a standard that allows enabled devices to connect and communicate using Wi-Fi speed, even in absence of central access point. Wi-Fi direct works by creating groups of connected devices. when a group is created, a single device is elected to be group owner. This means it is responsible for creating a software access point to which all the other components may connect. Although it provides some high-level services such as discovery of nearby devices, the actual communication is still based on standard socket technology. Mobile Opportunistic communication, a mobile peer-to-peer video sharing application based on Wi-Fi, poses challenges in on-demand video streaming caused by limited wireless communication range, user mobility and variable user population density. This technology addresses these challenges by exploiting the opportunistic mix use of downlink and direct peer-to-peer communication to significantly improve achievable overall system throughput.

System Overview:

In this research one of the smartphone in vehicle acts as a server and other smartphones act as a client's receiving the live stream captured by the server smartphone and which is being delivered to other smartphone in the same network .We tested the application on Wi-Fi network in this case. When the media file is streamed to the client but is only played and not stored by the client, it is called real time streaming. As the media file is not stored on the client system, real time streaming is preferred over progressive download for large media files. It eliminates need for storage on the user system, like full video Internet radio and TV broadcast The media delivery [12] is an important part in Vehicle Communication because the vehicles behind the server vehicle can use it to look and conditions ahead of them. The live media stream acts like a third eye to vehicles if they want to know about road conditions ahead of them.

TCP :

In this research video is sent from client and server smartphone at same time using TCP. A TCP connection provides a full-duplex service. TCP has been useful in this research as the both the server and client can at the same time utilize the duplex property of TCP. The automatic resending of lost datagrams and reordering are one of the features of TCP, but TCP also features flow control and congestion control.

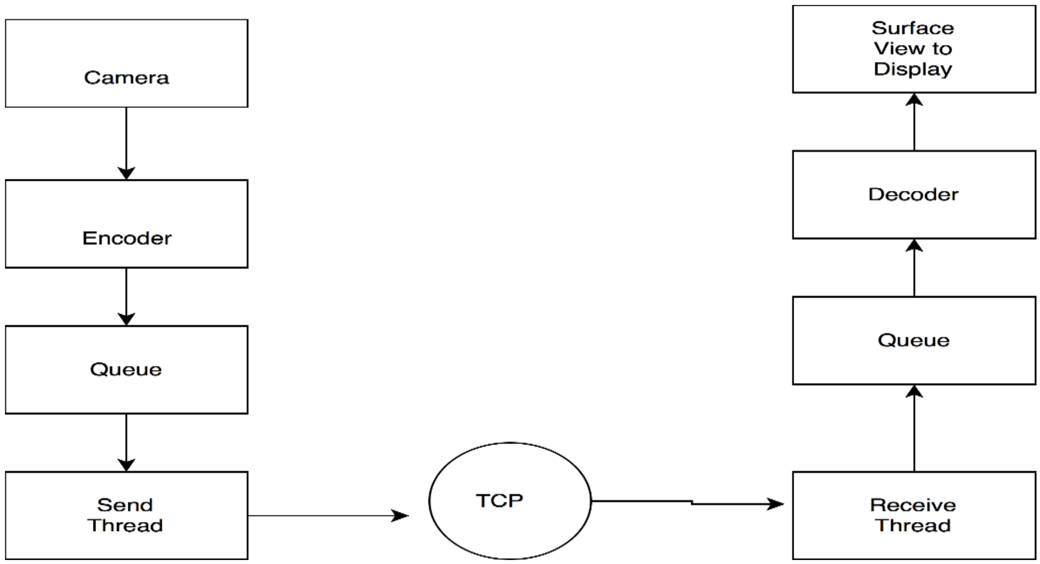

Figure :Use of TCP

Client and Server:

In the application there are buttons for client and server, after the user presses one of the buttons client/server, program goes to the Socket Wrapper method which takes client as an argument of type Boolean. This method contains three thread viz. init Socket();Send Thread(); and Receive Thread. This method helps to differentiate between Client and Server.

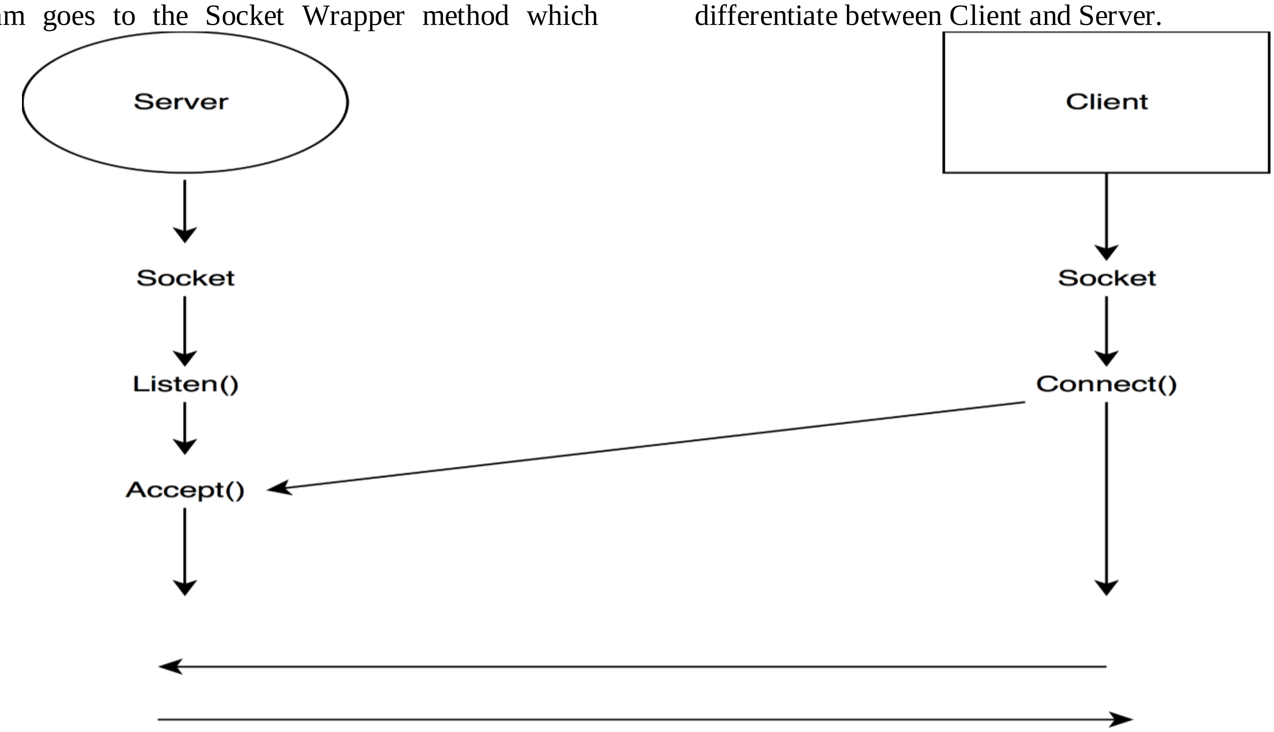

Figure : Client and Server data exchange process

Then the socket for client and server is created and define the IP and port of the server to Client Smartphone. For testing in Wi-Fi Direct, port address of 18888 used. It is necessary to define the ports of client and server when the object of socket is created for client and server. When the button called RECV and SEND is pressed, the thread defined in the Socket Wrapper can be used. When one of the thread i.e. Send Thread or Receive Thread starts, there might be something wrong while sending or receiving the data so the try and catch block is made inside the thread. As we know the video data is encoded before sending for each side. The video data will be stored in the queue named H264queue inside each thread of Send Thread and Receive Thread.

Camera Activity:
Camera Activity is the main activity which consists of two fragments viz. Camera2Basic Fragment where we display the local video captured by the smartphone device and in another part we display the video captured from other device. We display our videos in the Camera Activity. But before displaying the video we need to setup the output format of the video we are going to display. Also we have created some buttons like RECV, PLAY, RECORDH264, SEND, SERVER and CLIENT.

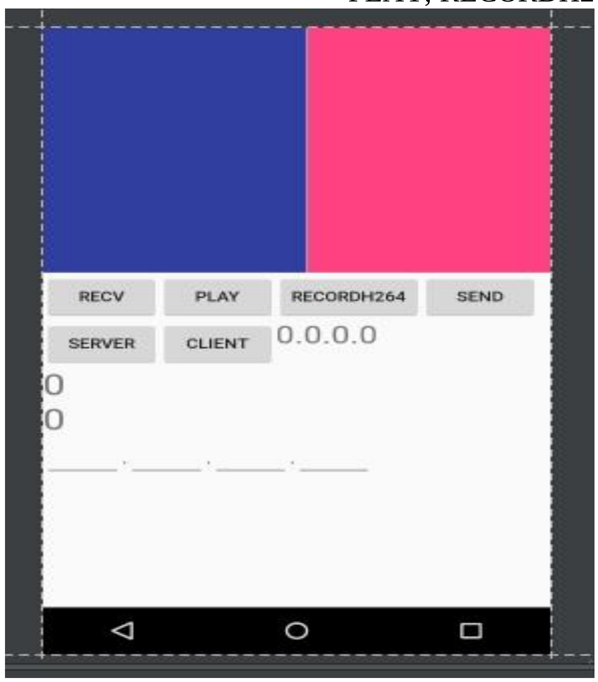

Figure: Camera User Interface

RECEIVE- It will receive data from other smartphone camera to display on the activity.
PLAY- It will play the video content while recording the video.
RECORD- It will capture data from camera and encode and store it temporary.
Here we have used texture view to display the video captured from the local camera device and surface views to display the image taken from client device. With the help of two views we can display both the videos simultaneously. The way to start this application working is by pressing the server and client button by one each of the smartphone user. The IP of both the smartphone is also being displayed on the screen. The IP of the local host can be obtained using the method named get Local Host IP(). This method uses object of Network Interface Class of Java to get all network interface and get every IP binding to each network interface. Then store all the IP addresses to a list variable named inet of type Inet Address. An instance of Inet Address consists of IP Address and its corresponding host name. Then this method returns a desired local host address.

CODING:
Encoding:

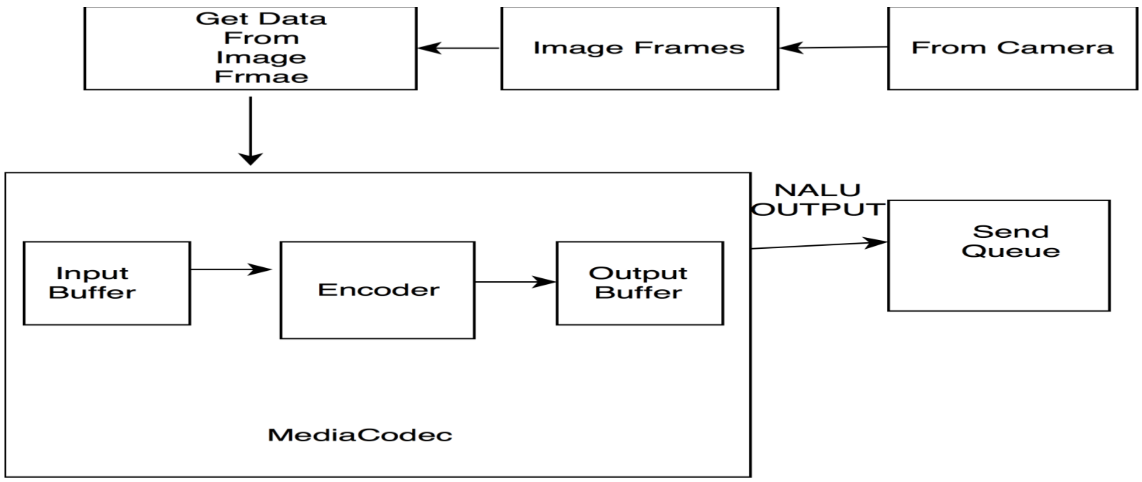

Figure : Encoding Data from Camera

ALGORITH FOR ENCODING

Get Data of Image From get Data Image method.
Store in Byte Buffer if there is no data in input buffer.
Set Image Width and Height Using Media Codec API
ENCODE Using MediaCodec API by setting KEY_BIT_RATE and KEY_FRAME_RATE.
Set Output as H264
Retrieve and Remove head of byte buffer.

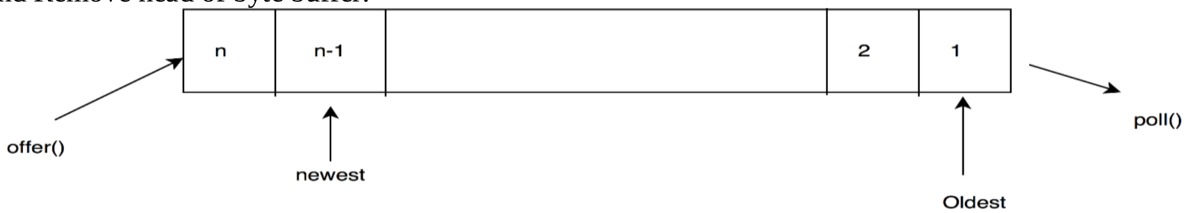

Figure :Using Queue to Encode

Offer (): inserts specified element into this queue if it is possible to do so immediately without violating capacity restrictions. Peek (): Retrieves, but does not remove, the head of this queue, or returns null if this queue is empty.
Poll(): retrieves and removes head of this queue, or returns null if this queue is empty.
Remove(); retrieves and removes head of this queue.

DECODING:

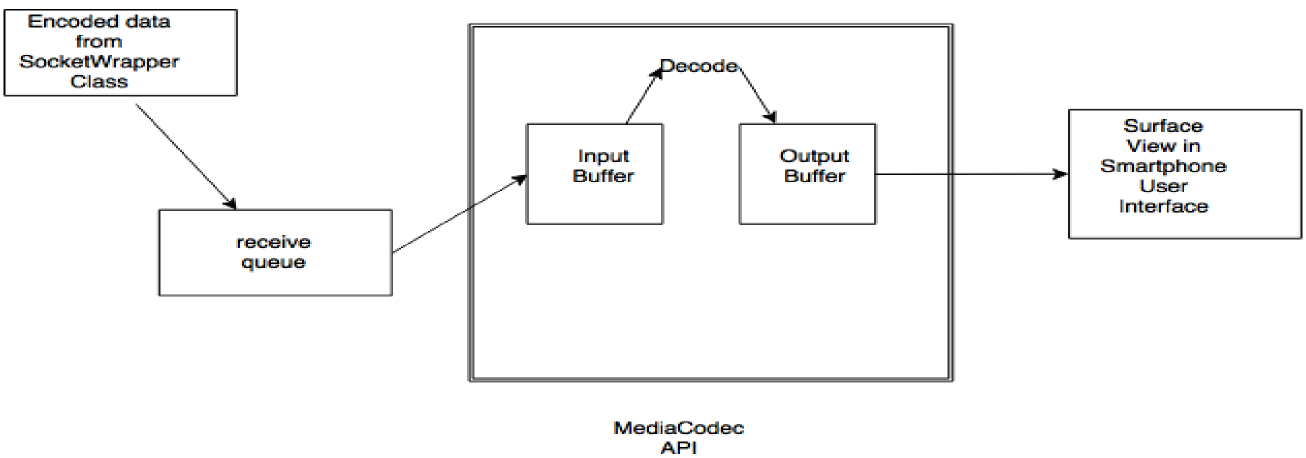

Figure: Decoding the data from Socket Wrapper

ALGORITHM FOR DECODING:

Receive encoded data in queue
Send the data to input buffer of Media Codec.
Set the variable for Surface view to display.
Set variable to access Media Codec input and output buffer.
Configure the desired Video Parameters.
Decode the H264 thread using Media Codec.

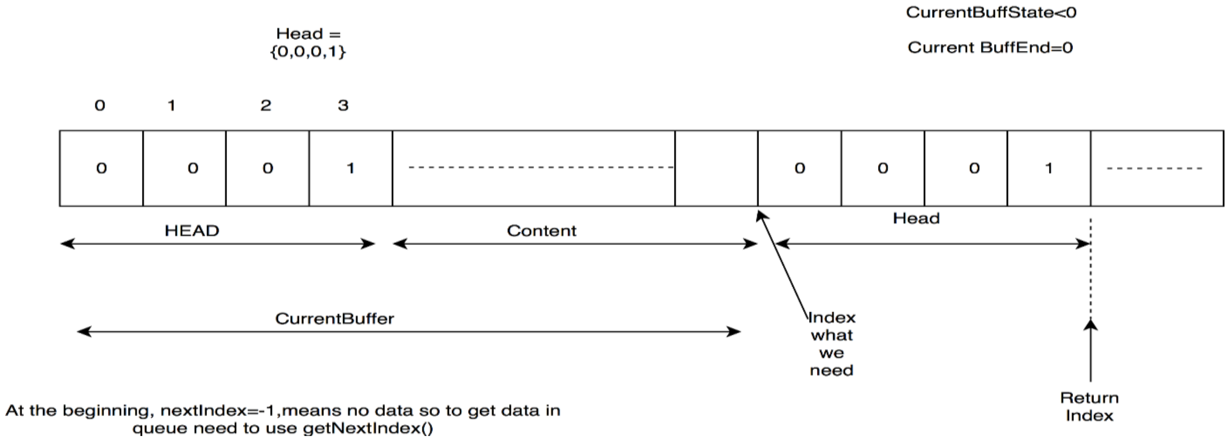

Figure: NAL Unit Decoding Process

Encoded data is in the form of this format, in NAL Unit which consists of head and content. At the beginning the current state of buffer is "Current Buff State < 0 and Current Buffer End = 0". Current Buffer consists of Head and Content of the NAL unit. In the beginning, next Index= "-1" which means there is no data so to get the data in queue, need to use get Next Index() function. There is a mechanism to ensure that the head {0,0,0,1} is not present in the content.

Performance Evaluation:

Application works by using the local network, as Vehicle Network has its own network to communicate between each other. The application works by first choosing a server and clients. One of the vehicle ahead of the other vehicle is chosen to be the server and other vehicle behind the vehicle can be the client. The vehicles must be in the same network. The vehicles discover each other by using the Wi-Fi Hotspot, by finding the vehicle within its reach. After finding the nearby vehicles, we begin the video transmission. This Application can display the video at the same time of transmission from another side and the video being captured by the local smartphone with delay until 20 meters' distance and without any delay until 10 meters just using Wi-Fi Direct without any RSU.

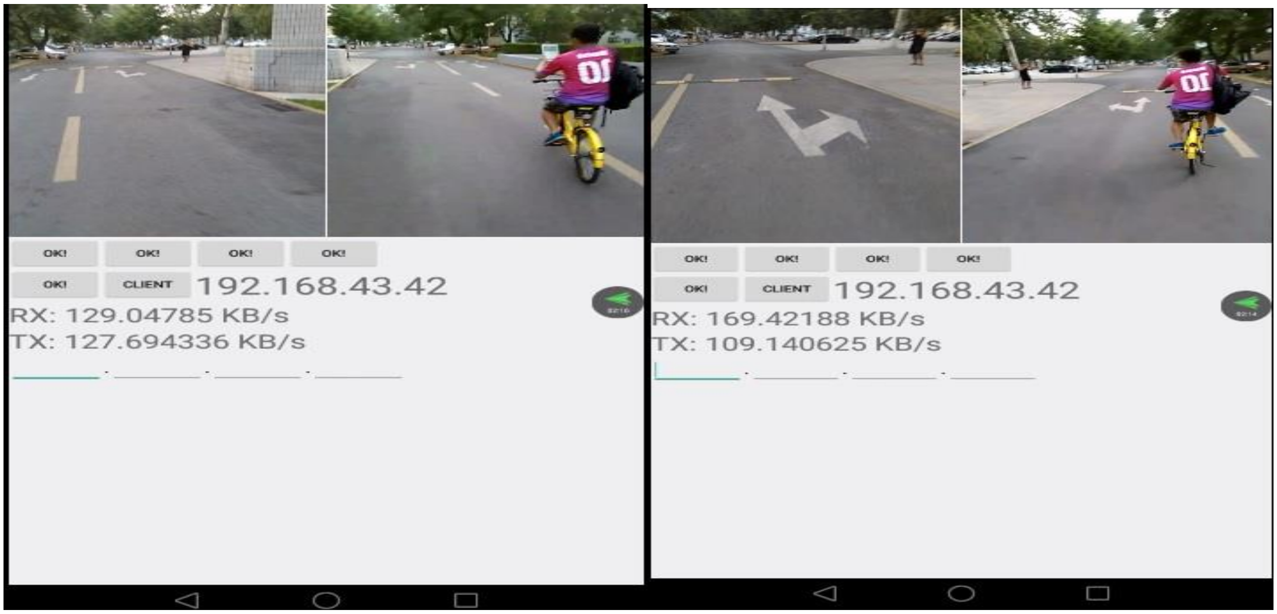

Figure: Image of Video Seen By Client Side from Behind

The person in yellow cycle is the server vehicle, the video taken by him appears on the left side of the smartphone display screen showing road ahead to client and on the right side of the smartphone display screen is the video taken by client behind the server. When the resolution and bit rate of the video stream being transmitted to other devices was changed, quality of video obtained in device also changed. If we increase the resolution and bit rate of the video stream transmission, a better video at the client side was observed. The video transmitted at higher resolution and bit rate looks clearer than video transmitted at lower resolution and bit rate. But when the resolution and bit rate was increased more than 320*240 at 2000000 bps, some delay was experienced while transmitting the live video. This application is also able to display the transmission speed and receiving speed of the mobile devices at the same time using Traffic Stats class in Android. It gives the whole speed of the smartphone system so to measure the receiving and transmitting speed of the video, only this application is opened and all the other applications are closed because even the background application can also contribute to this speed that we want to monitor. There is a thread running in this code every 1000 millisecond which is used as a counter to count the number of bytes coming and going out of the whole device to monitor the speed. Using the Traffic Stats and m Start RX variable which is initialized to zero globally, are able to store the number of bytes in rxBytes variable where the number of bytes during receiving is stored. If the number of bytes is greater than 1024 bytes its converted into KB/s. If it's more than 1024 bytes and less than 1048576, it is converted to MB/s. The same process is applied while getting the transfer speed of video.

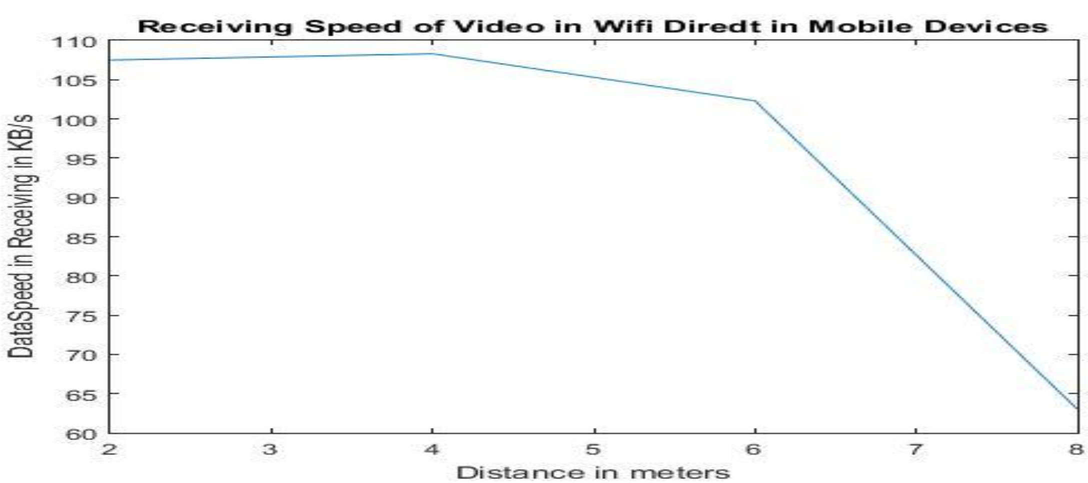

Figure 1: Receiving Speed Vs Distance

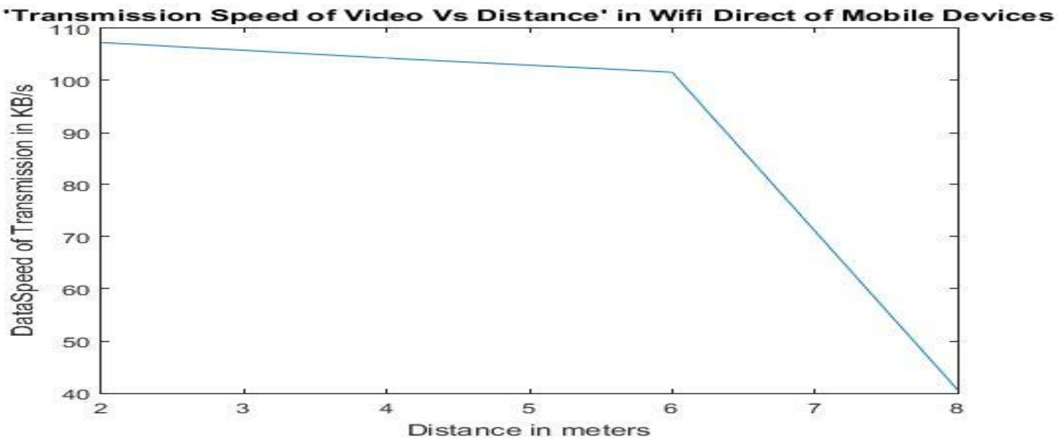

Figure 2: Transmission Speed Vs Distance

The devices are capable of transmitting the captured video and receiving the video from another mobile device at the same time. So each device is transmitting and receiving video at the same time. The data was taken in open environment where there is less chance of interferences from other Wi-Fi Access Points nearby. In this application also can increase the number of clients if the available bandwidth in the network is more and the more number of clients are nearby. The further the clients away from the Hotspots Access Point, the less the number of clients that can receive the broadcasted video because of the packet losses and interference from other Wi-Fi Networks. The performance can be further improved in scenario where there are other smartphones which are broadcasting their SSID at higher strength. Connecting to them will benefit because the chances of packets loss is less when the signal strength is better.

CONCLUSION:

Video is really useful to monitor the road situation for drivers as it provides a real view of the road situation a driver is about to face. Video is important information to judge or forecast the situation present which a driver can see while driving. Driver can look at the video while driving, just like he/she is looking at the mirror to look behind to know about the vehicles coming from behind. The video in the smartphone can provide even better information of road condition and lessen the number of accidents or reduce the number of traffic going to a particular way. After looking at the video the driver can decide before to choose the way where there is less traffic and the way which is safer. For example, if the road ahead has a landslide or the road further ahead is under construction, the driver can choose an alternate way and reduce the traffic jams. Even the Pedestrians and bicycle riders carry smartphone with them, they can also benefit and gain information of road situations. Due to limited bandwidth of channel, there is need for some methods for monitoring volume of data sent to network. The performance of routing protocol in Vehicle Network rest heavily on mobility model, driving situation, vehicular density. So, having a general routing solution for all Vehicle Network application environments or standard for routing protocols in VANET is necessary. These days more and more smart on-board applications may store lots of private information and vehicular route data which can reveal individual's activities. These dangers have to be overcome before communication architecture in VANETs is installed. In my future work I would like to use multi-casting technique to send more videos captured by more smartphones by using UDP protocol instead of TCP protocol. I will try to develop or use better encoding technique than H.264 which will encode the images better and compresses the size of data being sent to each other by smartphone. The new better encoder helps in decreasing the size of encoded data being sent to each other by compressing the video data more and more and make the size of data small. Also use better Wireless Networks like LTE or 4G which can provide better speed for the application instead just using Wi-Fi Direct or Wi-Fi Access points which is currently being used in this application.

REFERENCES:

[1] Qiujia Ji ,Hewei Yu , Haichao Chen . A Smart Android Based Remote Monitoring System. // Technological Advances in Electrical, Electronics and Computer Engineering (TAEECE), July 2015

[2] A Aloman ,A.I. Ispas, P. Ciotirnae. Performance Evaluation of video streaming using MPEG DASH, RTSP, and RTMP in mobile networks.// IFIP Wireless and Mobile Networking Conference , Feb 2016

[3] Teddy Mantoro, Media Anugerah Ayu and Dwiki Jatikusumo .Live Video Streaming for Mobile Devices: An application on Android Platform // Uncertainty Reasoning and Knowledge Engineering (URKE), October 2012
[4] Sandor Molnar, Peter Megyesi, Szilard Solymos, Zsolt Kramer et al. Flexible Media Transport Framework For Android // Multimedia & Expo Workshops (ICMEW), September 2016
[5] Ashkan Nikravesh, David Ke Hong, Qi Alfred Chen et al. QoE Interference Without Application Control // Proceedings of the 2016 workshop on QoE-based Analysis and Management of Data Communication Networks, August 2016

[6] Van Quyen Do, Nyugen Binh, Sun-Tae Chung.Design and Implementation of an Embedded Multimedia Live Streaming Decoder System. // Advanced Technologies for Communications (ATC), February 2015

[7] Binwen Fan, Guoyue Sun et al. Image Acquisition and Transmission in the Video Telephone Based on Android.// Fuzzy Systems and Knowledge Discovery (FSKD), January 2016

[8] Hyun Lee,Jae-Yong Yoo et al. MOVi+ : Opportunity Extension for Mobile Peer-to-Peer Video on Demand.// Consumer Communications and Networking Conference, March 2013

[9] Chen Zhang, Yuanzhu Chen et al. TCP adaptation with network coding and opportunistic data forwarding in multi-hop wireless networks. // PeerJ Computer Science 2, October 2016

[10] Noha M. Sadek, Hassan H. Halawa et al. A Robust Multi-RAT VANET/LTE for Mixed Control & Entertainment Traffic.// Journal of Transportation Technologies, May 2015

[11] Soumyalatha N, Rakesh Kumar Ambhati et al. Performance Evaluation of IP Wireless Networks Using Two Way Active Measurement Protocol.// Advances in Computing, Communication and Informatics (ICACCI) , October 2013

[12] Rajesh Bhadada, Birendra Rai.Performance Evaluation of Multicast Video Streaming via P2P Networking. // Special Issue of International Journal of Computer Applications, Dec 2011