Compressed Data Transfer from Mobile-To- Mobile Using Wi-Fi

Authors
Nikita Darekar[1], Disha Mishra[2], Swapnil Sonawane[3]
[1]133/20, Alibhai Charity Bldg, G.k Road, Dadar East, Mumbai-400014, Maharashtra
Email: *nikita.darekar@vit.edu.in*
[2]Main Road, Behind Kamptee Vachnalay, Kamptee- 441001, Nagpur
Email: *disha.mishra@vit.edu.in*
[3]Vidyalankar Campus, Vidyalankar College Marg, Wadala (East), Mumbai- 400037, Maharashtra
Email: *swapnil.sonawane@vit.edu.in*

ABSTRACT
This paper proposes a project with an aim to achieve Compressed File Sharing, between two Users with fast transfer of data. The data includes documents, image, audio and video. The connectivity is done using Wi-Fi direct libraries in an Android Studio, using Peer-to-Peer (P2P).

INTRODUCTION

In the recent years, Peer-to-Peer (P2P) communications and its applications have become conventional architecture in the wired network environment. In P2P systems, each node can act as a client and as a server at the same time and shares with others its own data. Wi-Fi direct technology is used for this connection. Today, many such applications for file transfer are available. But this project provides an extra feature of compression. That means the data will be transferred from one user and will be received by the other user in the compressed form. For the compression, the existing algorithms are used. Based on our research and literature survey, JPEG and FFMPEG algorithms have been implemented for compression of files, images and videos respectively.

With the increasing of network bandwidth, and the development of video compression technology, multimedia is becoming more and more popular on the Internet. Video and audio are gradually replacing the traditional information transmission method such as text and picture. In the era of the Internet, multimedia data varies from transmission protocols, media container formats, and the video / audio coding standards.

This paper is organized as follows. Section II provides problem definition for the proposed system, Section III provides literature survey relating file sharing. Section IV provides a detailed technical analysis and implementation of Wi-Fi Direct, JPEG compression technique, FFMPEG compression technique. Section V focuses on the proposed architecture or overall flow, Section VI highlights the expected outcome. Finally, Section VII concludes the paper.

Problem Definition

To develop a system for providing peer-to-peer content sharing on Wi-Fi network for Smart Phones. Peer-to-peer model faces unique challenges in the mobile context - limitations on processing power, on-board device memory, wireless data bandwidth, available battery energy. To employ peer-to-peer sharing among smart phones to consume free peer-to-peer wireless links.

The current drawbacks of this android application is that the quality of the files is compromised, currently works only on Android platform and Wi-Fi connection is a must.

Literature Survey

Nowadays many applications like SHAREit and Xender are trending in the world which are basically file transfer tool. It transfers all kinds of files using Wi-Fi hotspot without the usage of internet and USB. Also, there are applications like Photo compress and Video compressor which compresses audio and video files respectively. We aimed at proposing a system which integrates the above two technique of file sharing and file compression in a single application which is an advantage over using many applications for the same use.

Technical Analysis & Implementation

It is an approach that is used to basically compress the data before transferring. The application will be more useful where the bulk data transfer is needed with fast speed irrespective of its quality. Hence the compression module must be technically reliable.

Wi-Fi Direct

In a typical Wi-Fi Direct environment, clients discover and associate to WLANs, which are created and announced by Access Points (APs). In this way, a device unambiguously behaves either as an AP or as a client, each of these roles involving a different set of functionality. A major novelty of Wi-Fi Direct is that these roles are specified as dynamic, and hence a Wi-Fi Direct device has to implement both the role of a client and the role of an AP (sometimes referred to as Soft- AP). These roles are therefore logical roles that could even be executed simultaneously by the same device, for instance by using different frequencies.
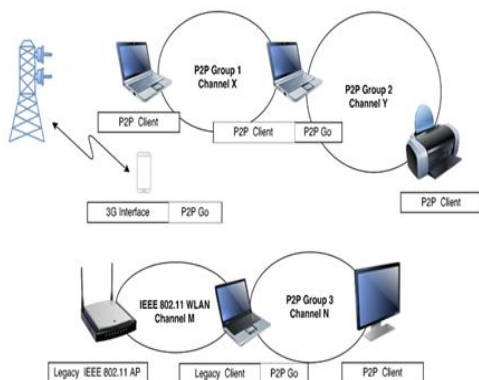

Fig 1: Structure of Wi-Fi Direct Network

JPEG compression

JPEG compression coding is an international standard which has high proportion of compression and only a little distortion so it has been widely used in our daily life. JPEG Compression is very useful for images that will be viewed online, where the size of the image makes a big difference in how long it takes a page to load. Judicious use of JPEG Compression can also be useful when printing a publication that may have hundreds or thousands of images, such as a high school yearbook. In this case, compressing the images slightly will save a large amount of space while sacrificing little-to-no quality.

JPEG is an algorithm designed to compress images with 24 bits depth or greyscale images. It is a lossy compression algorithm. The limited resolution of most computer monitors means that slight imperfections in the quality of the image will not be as noticeable. The standard diagram for JPEG compression coding process is
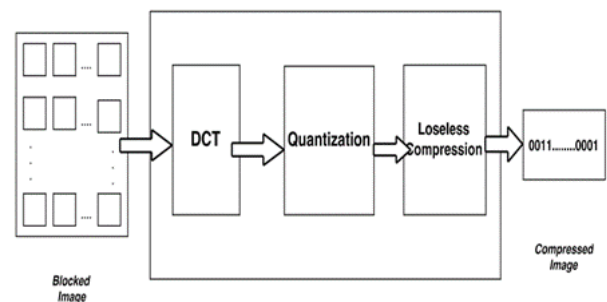
Phases of JPEG

Fig 2: Phases of JPEG Compression

FFMPEG Compression

Fast Forward Moving Picture Experts Group (FFMPEG) used for compression and conversion of audio and video files. In order to adapt to the vastly different transmission protocols, media container formats, as well as video / audio coding standards, the system uses FFMPEG as its kernel.

FFMPEG is an open source, very powerful multimedia framework that supports multiple transmission protocols, media container formats, as well as video / audio coding standards and it provides a unified data structure to store the

information extracted from multimedia data, thus effectively solves the difficulty in analysis of wide range of media data formats.

In addition, FFMPEG provides highly efficient codec algorithm that can meet requirements of real-time video / audio analysis. As we are using Android 6.0–6.0.1 Marshmallow (API level 23), the Media Codec libraries is been provided by it. FFMPEG libraries are available in Media Codec. Media Codec class can be used to access low-level media codecs, i.e. encoder/decoder components. It is part of the Android low-level multimedia support infrastructure. It is more compatible and reliable than the MPEG as the name suggests (Fast Forward).

Audio coding standard contains AAC, MP3, AC-3, etc. Video coding standard contains H.264, MPEG2, VC-1, etc.

Sender Side
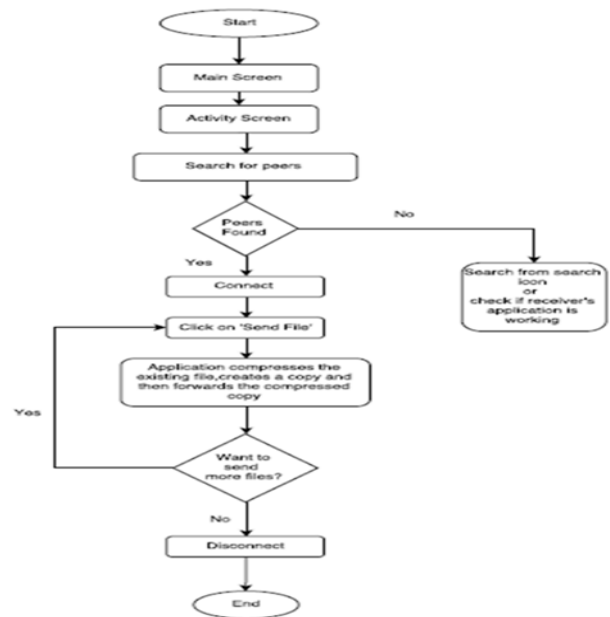
Fig 4: Process of sending the data/files

Receiver Side:
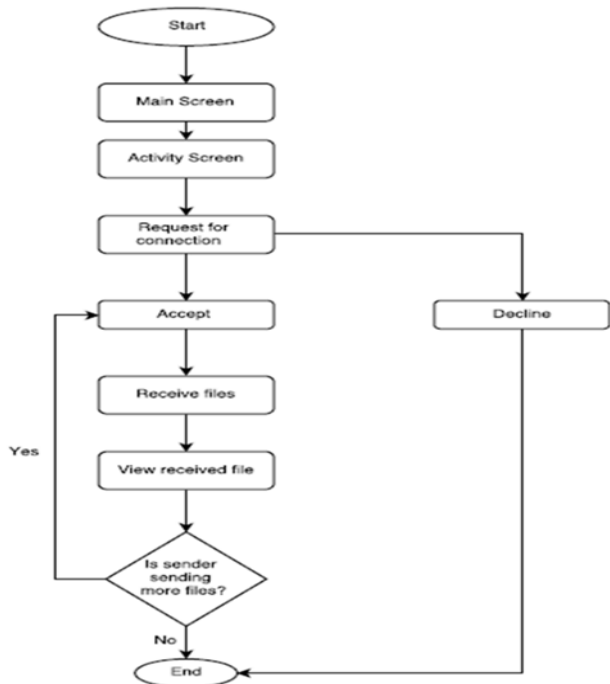
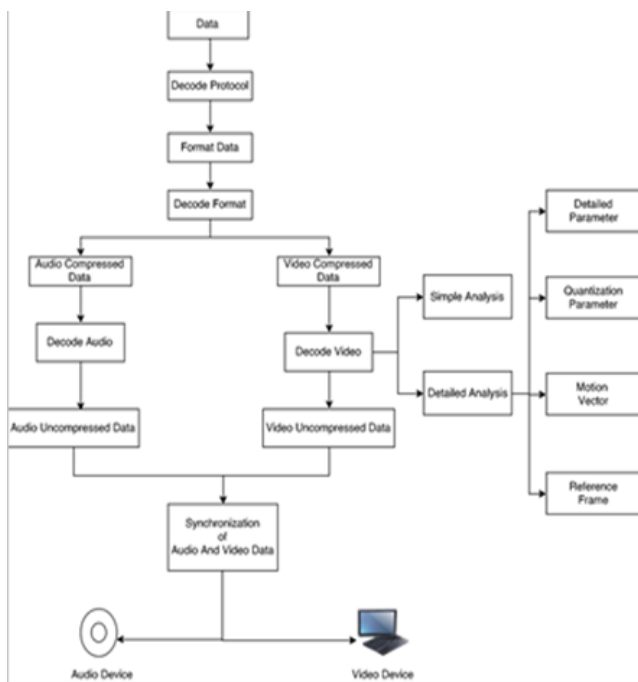**Fig 5:** Process of receiving the data/files

Fig 3: FFMPEG Compression

Proposed Architecture

Following is the architectural demonstration of the sender and receiver's end with respect with the flow of operations in the application.

Expected Outcome

For the proposed system the compression ratios of images, audios and videos are 60%, 70% and 50% respectively. Suppose we consider a file (mp3 file) of 8.46MB, after compression it gets converted to a file (mp3 file) of 1.44MB and is transferred to the selected peer.

CONCLUSION

This paper introduces the design and implementation of JPEG based image compression and FFMPEG-based real-time video and audio stream analysis system. The system support various type of transmission protocols, media container formats, images and video / audio coding standard; and it achieves real-time stream analysis requirement. However, the function of stream analysis may not complete. The method mentioned in this paper can be used as the basis of design more complex compression system.

REFERENCES

1. "Peer to Peer Content Sharing on Ad Hoc Networks of Smartphones", Piotr K. Tysowski, Pengxiang Zhao, Kshirasagar Naik. 978-1-4577-9538-2/11/\$26.00 ©2011 IEEE.
2. "Framework for Enhancing P2P Communication Protocol on Mobile Platform", Waheb A. Jabbar, M. Ismail, R. Nordin.
3. http://learnmem.cshlp.org/site/misc/tsg_JPEG_instructions.pdf
4. Comparison of different compression formats: https://homepages.cae.wisc.edu/~ece533/project/f06/aguilera_rpt.pdf
5. http://developer.android.com/reference/android/media/MediaCodec.html
6. The FFmpeg developers. FFmpeg Documen-tation. May 2013
7. Audio file format. http://en.wikipedia.org/wiki/Audio_format
8. Programming Android by Zigurd Mednieks, Laird Dornin
9. Android Programming: The Big Nerd Ranch Guide by Bill Philips & Brian Hardy
10. http://androidwarzone.blogspot.in/2011/12/ffmpeg4android.html
11. https://en.wikipedia.org/wiki/Wi-Fi_Direct
12. http://mvnrepository.com/artifact/com.googlecode.mp4parser/isoparser/1.0-RC-27