



ISBN	978-81-929866-6-1
Website	icsscet.org
Received	25 – February – 2016
Article ID	ICSSCCET117

VOL	02
eMail	icsscet@asdf.res.in
Accepted	10 - March – 2016
eAID	ICSSCCET.2016.117

Implementation of an Android Phone Based Video Streamer Using LIVE555 Streamer and FFMPEG

Neethu R¹

¹Assistant Professor, Department of Electronics & Communication Engineering, Karpagam Institute of Technology, Coimbatore

Abstract- This paper presents the methodologies used to implement an android phone based live video streamer using LIVE555 media server and FFMPEG which is used as a video convertor module. The camera on the mobile phone is used to capture real time video image which is then streamed over the Wi-Fi network in such a manner that the video can be viewed using the standard VLC media player anywhere within the range of the network..

Keywords- Live video Streaming, 3GP Android, MPEG4, LIVE555, FFMPEG

1. INTRODUCTION

With the Android listed as the best-selling smart phone platform worldwide in Q4 2010 by Canalys, a lot of effort and hard work have been put to explore new ideas and concepts, using its essentially free development tools and the Android Open Source Project (AOSP) which is tasked with the maintenance and further development of Android has been able to come up with large collection of applications. Owing to the vast scope that androids hold, a lot of vendors are into android business contributing to tight competition in this field which has henceforth made available android devices with best quality and functionalities at affordable cost.

The main objective of the paper is to first familiarize the users with the android application development platform and then to implement a LIVE555 video streamer on Android based mobile phone. The best part is that implementation is done completely using open source software tools, software modules and technologies.

Figure 1 shows the setup of the proposed system. The Android mobile phone's inbuilt camera is used to capture the video which is in 3GP format and has to be converted into MPEG4 format before being streamed over the network using the LIVE555 media streamer. Streaming over the IP network is done using phone's inbuilt Wi-Fi. The video can then be viewed on remote stations using the open-source VLC media player.

This paper is prepared exclusively for International Conference on Systems, Science, Control, Communication, Engineering and Technology 2016 [ICSSCCET 2016] which is published by ASDF International, Registered in London, United Kingdom under the directions of the Editor-in-Chief Dr T Ramachandran and Editors Dr. Daniel James, Dr. Kokula Krishna Hari Kunasekaran and Dr. Saikishore Elangovan. Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage, and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honoured. For all other uses, contact the owner/author(s). Copyright Holder can be reached at copy@asdf.international for distribution.

2016 © Reserved by Association of Scientists, Developers and Faculties [www.ASDF.international]

Cite this article as: Neethu R. "Implementation of an Android Phone Based Video Streamer Using LIVE555 Streamer and FFMPEG". *International Conference on Systems, Science, Control, Communication, Engineering and Technology 2016*: 585-589. Print.

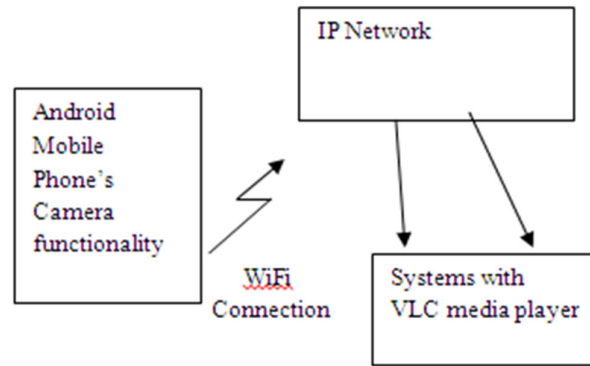


Figure 1 System Deployment Setup

2. Android Architecture Overview

The Android software stack is shown in figure 2. The android architecture can be subdivided into five layers: Android uses a heavily modified GNU/Linux kernel version 2.6. On top of the Kernel is the Libraries component that contains the various C/C++ Libraries used by various applications. Android Runtime is another vital component that is composed of two major parts: The Dalvik virtual machine and Core libraries. The next layer is the Application Framework which provides the API's. The Android Applications, are written in Java

Unlike other mobile operating systems like Apple's iOS, Palm's web OS or Symbian, Android applications are written in Java and run in virtual machines. For this purpose Android features the Dalvik virtual machine which executes its own byte code.

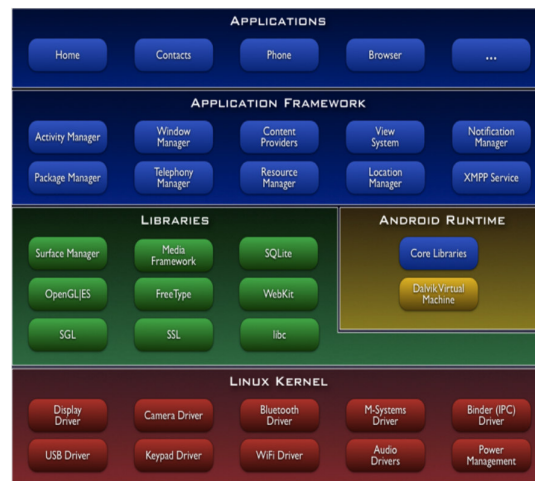


Figure 2: Android System Architecture

Dalvik is a core component, as all Android user applications and the application framework are written in Java and executed by Dalvik. It is designed specifically for Android and optimized for battery powered mobile devices with limited memory and CPU processing resources. The java source code in .java format is compiled by java language compiler (javac) to byte code format that is .dex format.

3. Android Application Development Tools

For the development of Android Apps several tools are required. Of that the most important one is the Eclipse IDE, which is a virtual mobile device that runs on users computer and it lets them develop and test Android applications without using a physical device. Android Development Tools (ADT) is a plugin for the Eclipse IDE that is designed to give a powerful, integrated environment to build Android applications. In addition, an Android SDK starter package that contains the various software development tools and the libraries are required by the Eclipse to compile and package the application into the Android Package file (Apk). If the application makes use of hardware features like camera, testing has to be done with actual phone..

Cite this article as: Neethu R. "Implementation of an Android Phone Based Video Streamer Using LIVE555 Streamer and FFMPEG". *International Conference on Systems, Science, Control, Communication, Engineering and Technology 2016*: 585-589. Print.

If the ADT plugins are added properly, the Eclipse Dalvik Debug Monitor Server icon (DDMS) will appear. Detection of the phone can then be verified through the DDMS Icon. If the drivers and phone settings are installed correctly, and the connection is successful, a mobile icon with its ID will be displayed under the Devices section as shown in Figure 3

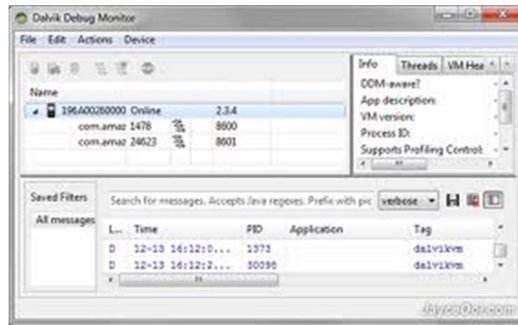


Figure 3: Eclipse's DDMS which shows the android phone as detected

4. Video Camera Streaming System

Figure illustrates the implementation of a video streaming system over an IP network using an Android phone. There are basically three modules used in the implementation: a video recorder module, an FFMPEG module and a LIVE555 streamer module.

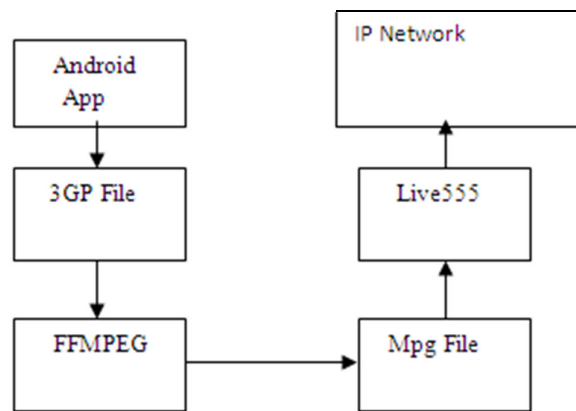


Figure 4: Video streaming Modules

The video recorder module is an Android application which records video in 3GP format. An FFMPEG library is used to convert 3GP file to MPEG format as Live555 server does not support streaming of 3GP files. The converted MPEG files are stored in a file (e.g. on a SD Card). Upon request by the user, the LIVE555 video streamer will fetch the video from the file and then it is streamed over the network and can be played by any system at the other end that has VLC media player installed in it.

The main reason why Live555 is used because it is an open source package, and so it can be modified to suit the system (e.g. cross compiled for ARM processor as used by the Android phone) and to meet customized requirement.

5. Implementation Results and Discussion

Combining the Video Recorder module, FFMPEG module and the LIVE555 streamer module running on Android phone (Samsung Galaxy Young using Android version 2.3.6), an MPEG4 video streaming system is implemented.

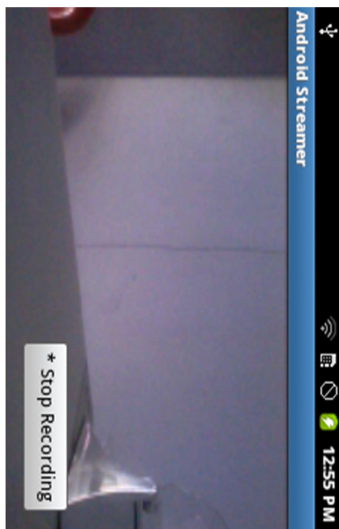


Figure 5: Screenshot of the screen once we run the program

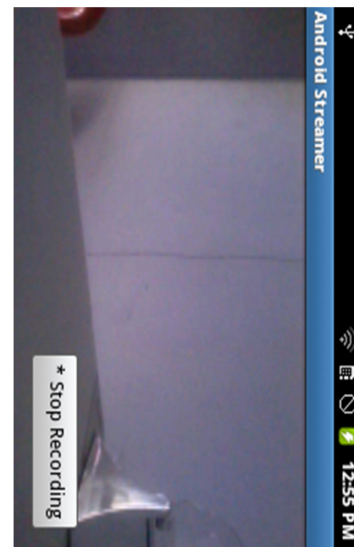


Figure 6: Screenshot which shows it is recording the video

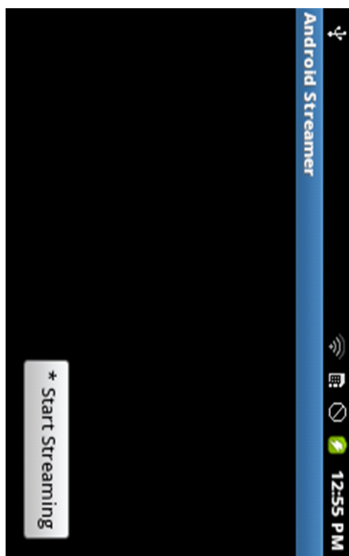


Figure 7: Screenshot which shows video recording is complete and now ready for streaming

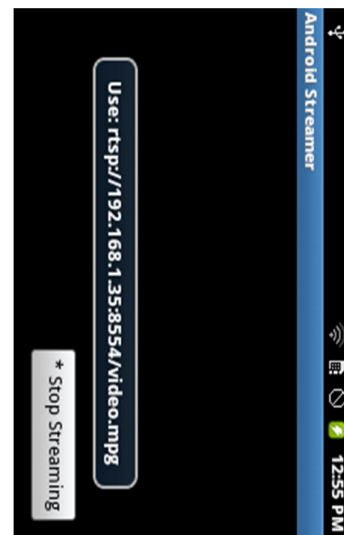


Figure 8: Screenshot showing video is getting streamed

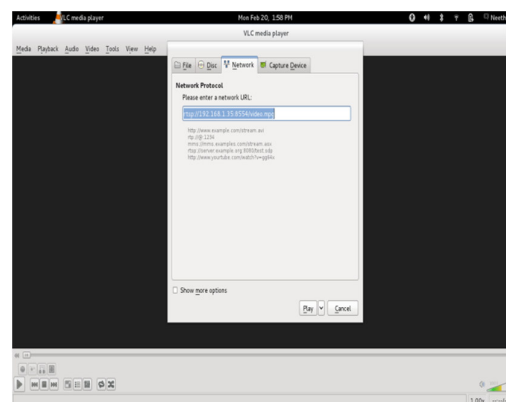


Figure 9: Screenshot of the vlc media player requesting the user to enter the details to play the streamed video

The implementation has the drawback that only after the video recording is completed, streaming is possible. This is due to the fact that, in FFMPEG only after the write operation is completed, the read operation can be performed. Another consequence is that the system will stop operation once the file size exceeds the maximum limit that the Android platform can support. The next motive should be to overcome these limitations, by bringing modification to FFMPEG or by using inter process communication methods.

6. Conclusion

This paper first provides an overview of the Android system, and then provides information about android application development platform and the various software components required for it. It then describes the implementation of a video streaming system using an Android phone as the video capturing device, by integrating a cross-compiled LIVE555 media server as the streaming server. The MPEG4 converted video is streamed over the network using Wi-Fi connection upon requested by the VLC media player.

References

1. N.Vun, Y H Ooi, "Implementation of an Android Phone Based Video Streamer", in proceedings of IEEE/ACM International Conference on Green Computing and communications & IEEE/ACM International Conference on Cyber, Physical and Social Computing, 2010.
2. N. Vun, M Ansary, "Implementation of an Embedded H.264 Live Video Streaming System", IEEE 14th International Symposium on Consumer Electronics, 2010.
3. Susu Xie and Bo Li*, Gabriel Y. Keung, "The Peer-to-Peer Live Video Streaming for Handheld Devices", in proceedings of IEEE Consumer Communications & Networking Conference, 2008.
4. Nicola Cranley and Mark Davis, "Study of the Behaviour of Video Streaming over IEEE 802.11b WLAN Networks", in proceedings of IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, September 2006.
5. C. Palau, J. Martínez-Nohales, J. Mares, B. Molina and M. Esteve, "On Mobile Video Streaming IPTV", 10th International Conference on Telecommunications 2009, ConTEL 2009.
6. Kihwan Choi', Kwanho Kim', and Massoud Pedram, "Energy Aware MPEG4 FGS Streaming", in proceedings of IEEE Design Automation Conference, August 2003.
7. Dimitrios Miras and Graham Knight, "Smooth Quality Streaming of Live Internet Video", in IEEE Global Telecommunications Conference, Globecom, January 2005.
8. Aravind.V, Archan Pratap Mishra, Sneha.S, "Enhancement of LIVE555 Media Server to support MPEG4 streaming", Technical Report IIITB-OS-2010-4b, April 2010.
9. Sun Microsystems, "Simplified Guide to the Java 2 platform, Enterprise Edition", white paper, September 1999.
10. Rajkumar Buyya, "Object- Oriented Programming with Java", Tata McGraw-Hill Edition, 2009.