ISSN:0975-9646

Android Based Remote Control of Mobile Devices Using VNC System

Rashmi A. Kalje, Prof. S. P. Kosbatwar

Smt. Kashibai Navale college of Engineering,
University of Pune, India.

***Abstract*— Remote control systems are a very needful element to control and monitor devices quickly. This paper implements a new design for remote control of Android mobile devices, analyzing the various alternatives and in search of the best solution in each case. Although the part of remote control, in case of mobile devices, has been slight explored, it may provide important advantages for testing software and hardware developments in several real devices. It can also allow an efficient management of various devices of different types; The main idea behind the proposed architecture was the design of a method to be used as a platform which provides the services needed to perform remote control of mobile devices. As a result of this research, a proof of concept was implemented. An Android application running a group of server programs on the device, connected to the Wi-Fi network depending on availability. This servers can be controlled through a small client written in Java and runnable both on desktop and web systems.**

***Keywords*— Android, Encoding, Remote control, Remote visualization, RFB, VNC**

I. INTRODUCTION

The project idea includes presenting android based remote control of mobile devices through VNC. This project proposes and analyzes different architectural approaches for the implementation of remote control systems of mobile devices using the Android software stack. In this work, we propose a fast screen sharing method to improve screen update rate in mobile VNC systems. In case of mobile devices, high complexity video compression techniques cannot be employed due to their strict computation limit. However, the bandwidth limitation requires a certain level of compression ratio. Thus, there exists a trade-off between encoder complexity and compression ratio for fast mobile VNC systems. We first integrate various video encoders into our prototype system, and explore their suitability for mobile VNC. Also, the existing RFB protocol for VNC is extended to easily integrate video encoders in a backward-compatible way. We additionally propose a new modified region coding method which transmits only modified regions between current and previous screen images. It can further reduce encoder computation and resultantly increase screen update rate. We implemented a prototype mobile VNC system actually, and its practical performance is widely evaluated.

In recent years, there have been popularly released a variety of multimedia mobile devices such as smartphone and tablet PC. The service that the today's mobile devices provide to users is almost comparable to desktops or similar devices. This trend requires the necessity of collaboration among these systems such as sharing multimedia contents and applications. VNC has been used as a tool for a multi-platform application suite allowing users to access graphic displays remotely. It is based on the thin-client architecture and uses the RFB (remote frame buffer) protocol for sharing a screen between distinct devices. The client software runs on the local user's machine while the server part operates at the target host. In this way, VNC brings remote display to the local user's machine. Successful application of VNC to mobile devices should guarantee rapid screen image transfer from server to client.

Fig 1 Remote connection between android device and local PC

In above figure 1, it shows the remote connection between android enabled tablet and local PC or laptop. While other remote computing solutions and VNC are mature and very effective solutions to provide remote access to a Smartphone from a PC, they are not explicitly designed for remote access from a Smartphone. Remote access from a PC is spontaneously derived from natural tendency to a user since the local PC provides a user interface that is equivalent to that of the remote Smartphone. The overall screen display is shown on the desktop/ laptop and the user controls the remote Smartphone using a mouse and a keyboard at the local desktop. In such a scenario, the overall user experience for operating remote computing is close nearly to that of using a local PC. Mobile VNC poses new encoding schemes while the information can be accessed anywhere and anytime, hence VNC is required maintain the remote control between the devices, but along with the VNC, different encoding schemes are also necessary to provide screen update rate. While performing the remote visualization, numbers of alternatives are available, selecting one as best among them is the important issue. And, for that screen image coding is used to select the best alternative when numbers of alternatives are available and their consequences cannot be forecast with certainty. Using unique user Id and password or an IP address, access is granted or denied for particular subject to object.

II. RELATED WORK

In [2], it proposed that user will be able to access and manipulate the desktops of remote computers through a VNC viewer that will be provided on the user's cell-phone.
In [5], the remote computer's desktop is accessed from the normal mobile phone. The remote desktop connection can be made wireless and can be accessed from any part of the world. We can control the remote computer as like our normal local computer by using a java enabled mobile phone.
Many of previous works have studied efficient screen image encoding. Screen image compression is typically required to transfer screen image data with limited network bandwidth, and an appropriate encoder should be carefully selected in terms of compression ratio and speed. Original VNC uses only six encoding algorithms where all implement lossless compression. It implemented a new encoding scheme called 'tight'. The 'tight' uses data analyzer and a set of data filters as preprocessors to improve adaptation ability in encoding.
VNC is a client-pull system where the server sends screen image data to the client only in response to the screen update request of the client. Since it prevents surplus updates, it is probably suitable for thin-client system such as VNC. In high latency environment, however, the update request from the client can be delayed, and it badly affects screen update performance. In C. Taylor and J. Pasquale [4], the message accelerator is employed to mitigate the effects of network latency. The message accelerator requests more frequently updates to the server while forwarding screen update data to the client.

III. PROPOSED METHODOLOGY

A. *Mobile VNC Architecture*

The goal of our system is to provide the services needed to perform remote connection between android enabled mobile device and desktops/laptops. It consumes the existing VNC system in our application. Enhanced encoding technique (MPEG and RFB) are used to improve system performance.

User can be able to transfer the files between mobile device and Desktops/Laptops User can able to debug the android device. User can able to view processes, status, services of android device. User can execute the services provided by the server. Remote VNC application can be used to perform remote control of mobile devices with the help of thin client. Mobile VNC require following functionalities to perform the processing:

1) User first download and install Remote VNC application on mobile device.
2) Start the application.
3) Provide unique IP/Code/URL to Facilitator (other User)
4) User will enter the IP/Code/URL in to browser and get the access of remote server.
5) User will request different services available on client's mobile device.
6) Service will respond to client's request.

Following services provided by the architecture are as follows:
1) Remote visualization service
2) Application management service
3) Service and process management service
4) File management service
5) Firmware management service

We implement a prototype system to evaluate proposed methods practically. Our proposed system can be used on mobile operating system platform which can be used for android enabled mobile devices or tablet PC's which having installed VNC application. It consists of VNC client and VNC server which are based on open source android platform.
The following figure 3 describes the overall mobile VNC architecture. How the server sends the updated screen to the client is implemented in several steps:
1) VNC server keeps the client requests in a separate frame buffer which is called as RFB (Remote Frame buffer) protocol.
2) After receiving the screen update request from client, the server captures the screen image from framebuffer. It can be used for further processing.
3) VNC server uses one of the improved encoding technique, the color space of the screen images is then converted into YUV format before compression because most video encoders supports YUV format which is easy to compress.
4) The captured screen image data is encoded and send form server to client with a framebufferUpdate message.
5) The client then decoded and receives the updated bit stream.
6) The procedure continues until the connection between client and server is finished or the connection is lost.

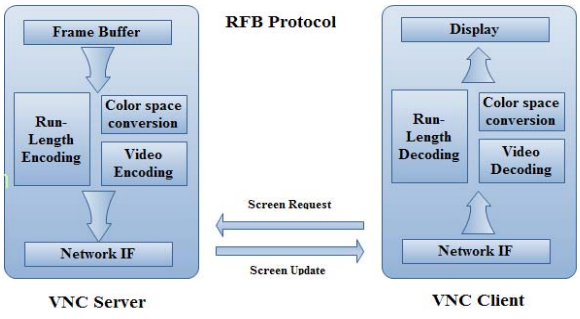

Fig. 2 Overall mobile VNC Architecture

B. *Hierarchical Region Detection Algorithm*

In the first step of modified region coding, a screen image data is segmented into unit rectangles which are fixed size blocks. Then, difference detection between current and previous screen images is performed for each unit rectangle. If all pixel values are identical, the unit rectangle is regarded as a skip block, which does not need to be transmitted. If any difference is detected, the unit rectangle is encoded and is transmitted to the client as usual.

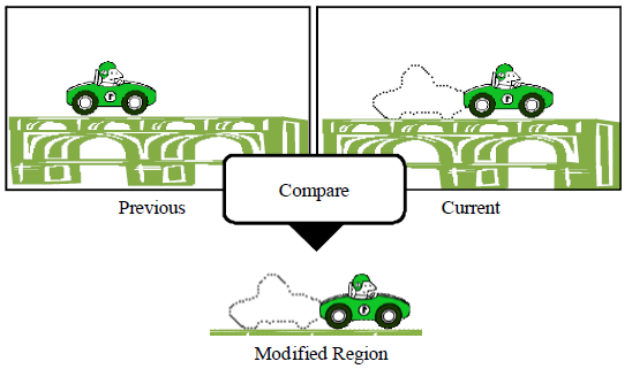

Fig 3. Modified Region Detection[7]

We first propose a hierarchical 3-step region detection algorithm to measure the screen update rate, which can be performed into three steps.

Step 1: The unit rectangle is sampled by a 2*2 pixels in both horizontal and vertical directions. Compare the pixels. If modified pixels are detected, then stop. Otherwise go to step2.
Step 2: The unit rectangle is sampled by a 4*4 pixels in both horizontal and vertical directions. Compare the pixels. If modified pixels are detected, then stop. Otherwise go to step3.
Step 3: The unit rectangle is sampled by a 8*8 pixels in both horizontal and vertical directions. Compare the pixels. If modified pixels are detected, then stop. Otherwise the unit rectangle is a final code block.

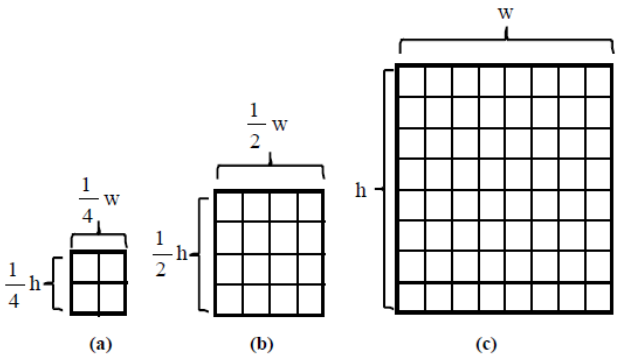

Fig. 4 Hierarchical 3 step region detection method: (a) Step 1, (b) Step 2, (c) Step 3 [7].

IV. CONCLUSIONS

In this paper, we implemented a proposed system for mobile VNC, and described practical performance evaluations. It provides the less complexity than previously developed system. VNC guarantees connection between client and server. In particular, various alternatives have been analyzed to perform the most related aspects, determining their advantages and disadvantages. As a result of the study process, the main features have been recognized, and the architecture should offer the best procedures to carry out the exchange of data. The remote visualization will be delegated to a VNC system only if the device allows the use of this software. Otherwise, the system will have to implement a native method to capture screen output directly.

REFERENCES

[1] [1 Android. http://www.android.com Retrieved March 1st, 2011.
[2] Archana Jadhav ,Vipul Oswal, Sagar Madane, Harshal Zope, Vishal Hatmode. VNC architecture based remote desktop access through android mobile phones, Vol. 1, Issue 2**,** 2012
[3] Cheng-Lin Tsao, Sandeep Kakumanu, and Raghupathy Sivakumar School of Electrical and Computer Engineering Georgia Institute of Technology Atlanta, GA 30332, USA (clt-sao,ksandeep,siva)at ece.gatech.edu, "SmartVNC: an effective remote computing Solution for smartphones" in 2011
[4] C. Taylor and J. Pasquale, “Improving video performance in VNC under
high latency conditions,” Proc. of the International Symposium on Collaborative Technologies and Systems (CTS 2010), pp.26-35, May 2010.
[5] Esteve,Member, IEEE. R. Manikandasay, Remote desktop connection using mobile phone, Volume 2, Issue 8, aug 2013.
[6] Gonzalez Villan , Student Member, IEEE and Josep Jorba Remote control of mobile devices in android platform Angel, Gonzalez Villan, Student, IEEE and Joseph Jorba Estave, member, IEEE.
[7] Ha-Young Ko, Jae-Hyeok Lee, Jong-Ok Kim, Member, IEEE, Implementation and evaluation of fast mobile VNC Systems, IEEE Transactions on Consumer Electronics, Vol. 58, No. 4, nov 2012.
[8] H. Shen, “A high-performance remote computing platform,” Proc. of IEEE International Conference on Pervasive Computing and Communication (PerCom 2009), pp. 1-6, Mar. 2009.
[9] Justin Grover a,b,"Android forensics: Automated data collection and reporting from a mobile device", Digital Investigation 10 (2013) S12–S20, 2013
[10] Kiran Karra, Virginia Polytechnic Institute and State University, "Wireless distributed computing on the android platform", 2012
[11] K. V. Kaplinsky, “VNC tight encoder-data compression for VNC,” Proc. of the 7th International Scientific and Practical Conference of Students, Post-graduates and Young Scientists (MTT 2001), pp. 155–157, Feb. 2001.
[12] Tristan Richardson RealVNC Ltd (formerly of Olivetti Research Ltd ATandT Labs Cambridge), The RFB protocol, Version 3.8**,** 2010.