

VR Based Smart Gaming Using Virtual Network Computing

Tejas Kharche¹, Akshay Chaudhari², Kaustubh Joshi³, Saurabh Panse⁴, D.O. Shamkuwar⁵

^{1,2,3,4} Student, Computer Department, SITS College of Engineering, Narhe, Pune, Maharashtra

⁵ Asst Prof, Computer Department, SITS, Narhe, Pune, Maharashtra

Abstract- Current generation likes to various games in various views. Better experience of game gives satisfaction to the user. In existing gaming systems, the user gives input from the mouse for viewing in the game. The current system does not allow the user to feel the environment.

Gaming experience can be enhanced by using this mobile gyroscope along with VR headset. Most recent smartphones are equipped with gyroscope sensors to measure the movement for device tracking. This inspired us to develop a gaming system which will allow the user to feel that he/she is actually playing the game by interact with that three-dimensional (3D) environment.

Key Words: Virtual Network Computing, Screen Mirroring, joystick, Gyroscope, Remote Procedure Call (RPC).

1. INTRODUCTION

Virtual network computing is a popular tool for sharing an application, allowing users to access graphics displays remotely. It has been quite difficult to boost display update rate through quick screen image encoding in VNC systems. It supports access to resources of a remote system through sharing the desktop of the remote system.

Virtual Reality term is normally used for the 3D environment that allows the user to enter and interact with alternate realities. In existing Gaming systems, the user gives input from the mouse for viewing in the game. The current system does not allow the user to feel the environment. Gaming experience can be enhanced by using this mobile accelerometer along with VR headset.

We will be creating an android VR VNC viewer for playing computer games. This application will run on a mobile platform (Android). The android VR VNC viewer will provide a panoramic view of the game environment. The view of the environment will be mapped onto what is seen on a set of larger screens. By doing this, the areas of the users which are normally blocked by the frame of the screen will be displayed. The android VR VNC viewer allows the player to look around from his current stationary position.

2. RELATED WORK

The paper [1] gives an extension of VNC for effective real-time collaboration. For Extension of Ultra-VNC server, the Java Viewer client is described. The collaboration server provides remote system resources. The collaboration participants can use the remote system resources through

the extended viewer. For effective collaboration, three kinds of access authorities such as administrator, worker, and spectator are provided. The administrator grants or modifies authorities of the users, controlling the collaboration server. The workers can access the remote resources provided by the collaboration server, while the spectators cannot handle the resources, they only monitoring the remote desktop of the collaboration server. To support those access authorities, the Ultra VNC server and the Java Viewer are extended. In this they present new features to support the one-click connection to the VNC server from the VNC clients, which enables the collaboration participants to connect to the collaboration server without the overhead.

In paper [2] a number of video encoders are integrated into a prototype system, and most suitable codec for mobile VNC is investigated. The existing RFB protocol is extended to integrate video codec's. The overall system architecture is modified from serial operation to parallel. And the coding section is again shrinking the encoding period of screen images. JPEG is the most suitable for mobile VNC in terms of both complexity and compression ratio.

A novel hybrid encoder model [3] with traditional VNC system which gives the windows desktop screen image in traditional mode or in streaming mode and the sent can be cast using H.264 with RTP packets depending on the current heuristic decision. And when using traditional VNC system and current frame differ being below a threshold, sending rate will be reduced to optimize network bandwidth usage. Considering smartphones computing ability and battery usage limit, this model is really suitable in that it ensures only a few resources are needed to implement it.

Design of an Air mouse which does not require a mouse pad and works on the hand gestures is proposed in [4]. This is a tilt-based approach. The change of acceleration or a tilt in the direction of Z-axis is used for click and release of the cursor of the mouse.

The AT-MEL AVR microcontroller [5] family and on the Microchip's Ethernet integrated circuit ENC28J60, showing that a low-cost microcontroller can implement the network functions. They use of UDP as the transport protocol instead of TCP with fields to the RFB header.

Paper [6] gives Remote Desktop Client Platform (RDCP) solution which allow PC applications to be easily adapted to Smart TV application environment. We have used Virtual Network Computing (VNC) as a core of RDCP. This method retains original functionalities of Smart TV. Implementation

of this solution has been done using Smart TV SDK that contains cutting-edge web technologies with the last updates of HTML5. By using proposed RDCP solution, the VNC ported Smart TV could extend its usage into education and entertainment field.

3. SYSTEM OVERVIEW

We are creating the gaming system which enables us to play any Computer game effectively with the help of VR headset. We are connecting the computer to Android mobile via Wi-Fi and cast computer screen on the mobile using VNC in the split screen format. The system will get the rotational movements as input from the gyroscope sensor of the mobile phone and the obtained 3D coordinates will be converted into the respective 2D coordinates and passed as input to the computer. When the user rotates his/her head movements in the air, accordingly the character in the game will also rotate his/her head. With this system, the player will experience 360° view of the game.

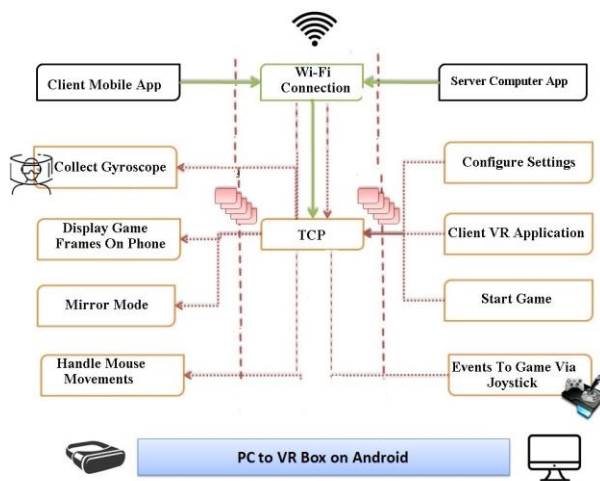


Fig-1: Diagram for System Architecture

The figure 1 shows the architecture diagram of the system overview. This system is casting the computer screen on the Android mobile phone. Android phone will continuously capture head movements with the help of gyroscope and send it to the computer device.

The connection between Android mobile and computer will establish with the help of Wi-Fi technology. All the data between the computer and Android mobile will be transmitted with use of socket programming mechanism. The view perspective of a character in the game will change according to player's head movement.

4. ALGORITHM USED

1. Start the system
2. Initialize the Wi-Fi access point
3. Initiate the client app and server app
4. Connect to android mobile to Wi-Fi access point

5. Start the server
6. Start computer game
7. Enter server IP and port no in android app
8. while (app is open)
9. If method is screenshot then capture screenshot and send it to mobile app.
10. Else if method is transfer-data then receive the gyroscope reading from android mobile and move computer's mouse pointer.
11. End while
12. Stop.

5. EXPERIMENTAL RESULT

Our test bed consists of a smart phone and a PC, which are connected by a Wi-Fi network. The VNC server and client are running on a smartphone.

The different resolution for screen casting is given below table.

Table 1: Screen Casting Results

Sr.No	Resolution	Screen Casting For	CPU Overhead	Network Congestion	FPS
1	512*384	Windows	1.8	70	35
2	512*384	Game	2	75	30
3	800*600	Windows	2	86	29
4	800*600	Game	2.2	100	27
5	1024*768	Windows	4.5	200	25
6	1024*768	Game	5	220	23
7	1366*768	Windows	5.6	230	21
8	1366*768	Game	6	260	18
9	1920*1080	Windows	10	490	13
10	1920*1080	Game	11	520	10

The table 1 shows the clear comparison for better screen casting results for different screen resolutions.

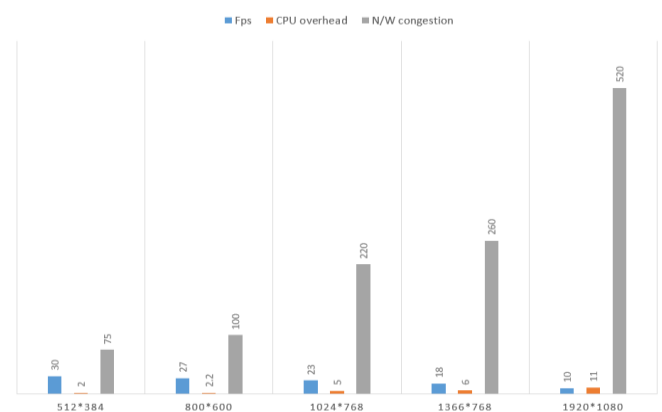


Fig-2: Plot of Screen Casting Results

6. RESULTS

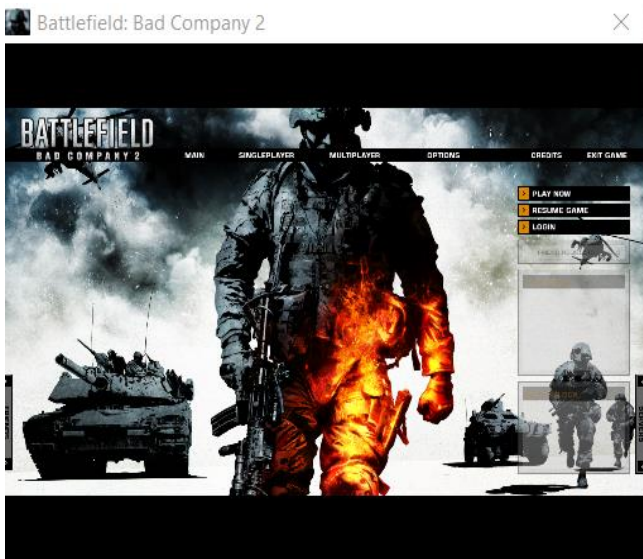


Fig-3: Game running on computer in mono mode



Fig-4: Server side login tab

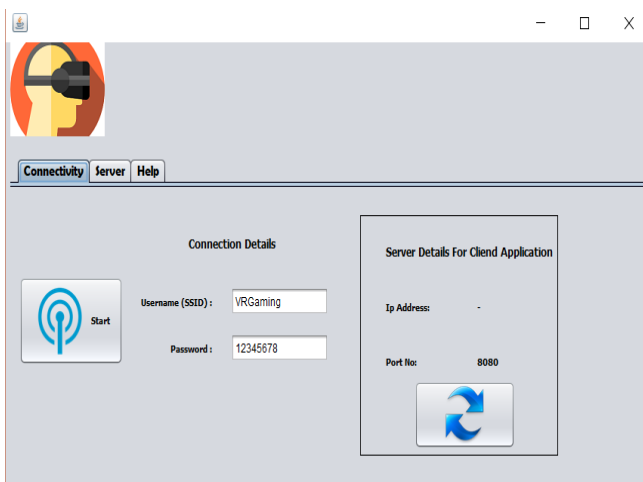


Fig-5: Server side connectivity tab

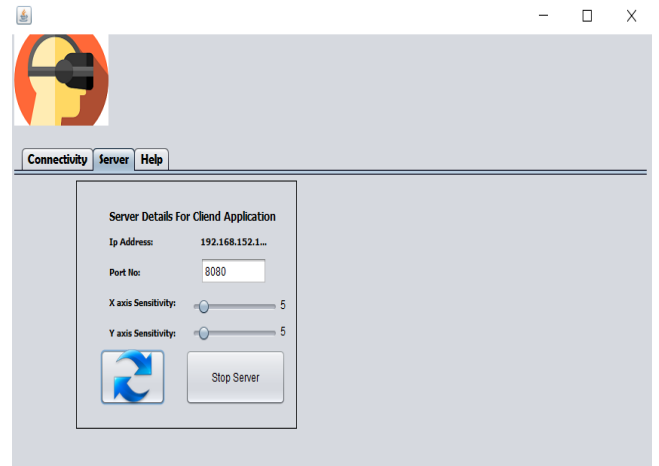


Fig-6: Server tab

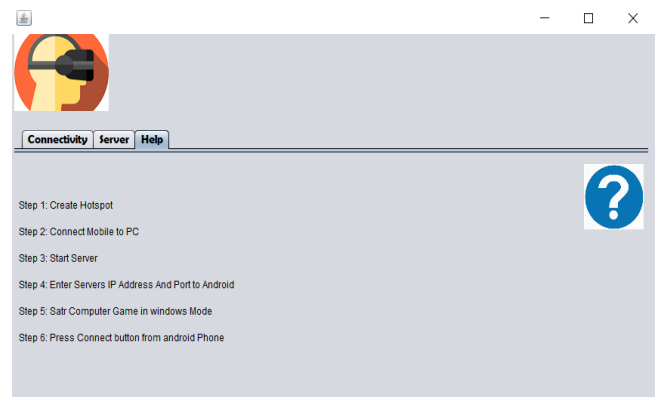


Fig-7: Help tab

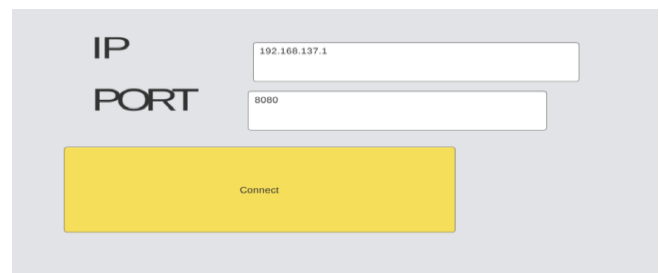


Fig-8: Client side connectivity tab



Fig-9: Client side in stereo mode

7. CONCLUSION

Here we developed an application that uses the display and sensors of existing android phone to transform it into a portal to PC games, so that we can play our PC games in virtual reality on android mobile. This VR structure is mostly concerning the mirroring Windows that appear on Android Smartphone in the split-screen VR mode. It also makes sure to create a virtual 3D effect on games. System can be implemented in very low cost and provides better speed to send data on VR mounted phone.

REFERENCES

- [1] Tae-Ho Lee, Hong-Chang Lee, Jung-Hyun Kim, Myung-Joon Lee, "Extending VNC for Effective Collaboration", School of Computer Engineering and Information Technology, University of Ulsan 680 - 749 San 29, Mugeo 2-Dong , Nam-Gu, Ulsan Korea.
- [2] Ha-Young Ko, Jae-Hyeok Lee, Jong-Ok Kim, Member, IEEE, "Implementation and Evaluation of Fast Mobile VNC Systems", IEEE Transactions on Consumer Electronics (Volume: 58, Issue: 4, November 2012).
- [3] Ligang Wu and Wenjun Wu, "Casting Windows to Android Based on Hybrid Encoder Model", Consumer Electronics, Communications and Networks (CECNet), 2013 3rd International Conference on 20-22 Nov. 2013.
- [4] Daniel Zinca, "Design of a modified RFB protocol and its implementation in an ultra-thin client", in Electronics and Telecommunications (ISETC), 2010 9th International Symposium on 11-12 Nov. 2010.
- [5] Sohail Ahmed, Mohammed Abdullah Zubair, Irshad Basha Shaik "Accelerometer based Wireless Air Mouse using Arduino Micro-Controller Board" Communication Technologies (GCCT), 2015 Global Conference on 23-24 April 2015.
- [6] Baul Kim, Rustam RakhimovIgorovich, PrimovEldorIsmoilovich, Yunjung Park, DugkiMin "Smart TV Usage Extension Method using Screen Division and Virtual Network Computing"2013.
- [7] <https://docs.unity3d.com/ScriptReference/XR.XRSettings.html>
- [8] <https://docs.oracle.com/javase/7/docs/api/java/awt/Robot.html>
- [9] <https://docs.unity3d.com/ScriptReference/Gyroscope.html>
- [10] <https://www.javatpoint.com/socket-programming>