

Real Time Communication using Embedded System

beyond Videoconferencing and towards Telepresence

S. R. Nalamwar
Computer Engineering
Department
AISSMS College of
Engineering,
Pune-01, India

Swapnil Kalhapure
Computer Engineering
Department
AISSMS College of
Engineering,
Pune-01, India

Abhitjit Khatake
Computer Engineering
Department
AISSMS College of
Engineering,
Pune-01, India

Suyash Gandhi
Computer Engineering Department
AISSMS College of Engineering,
Pune-01, India

Kunal Jain
Computer Engineering Department
AISSMS College of Engineering,
Pune-01, India

ABSTRACT

Telepresence is a service that enables realistic and immersive presence to the user. Though the service is no longer a new terminology, it has not been used widely due to the cost of implementation and mass manufacturing. This paper presents a telepresence system that focuses on building a robotic platform for telepresence. This system is built using open source hardware and software that makes the system affordable and cheap. This system is able to remotely operate over Internet and work in a scenario with audio-video conferencing capabilities. This paper describes a fast and affordable alternate conferencing solution using WebRTC open source project. The WebRTC project enables browser to browser real-time communication. Today several vendors provide telepresence devices, but at a cost that the general public cannot buy. This paper showcases an alternate implementation of telepresence system that comprises of a robotic platform and an audio-video conferencing solution. This system is built using off-the-shelf hardware systems and open source software stacks. However the implementation showcased is meant only for demonstration purpose, the same system with some tweaks can be deployed in actual real-life scenario where the operator can reach the intended destination remotely.

General Terms

Telecommunication, WebRTC, Real Time Communication, Intel Edison, Authentication

Keywords

WebRTC, audio, visual, Telerobotics, telepresence, eleoperation, video conferencing.

1. INTRODUCTION

In this era of Global Communication, Real time communication has proved to be a boon for worldwide communication. Many big I.T organizations and Industrial sectors use Video Conferencing for managing various projects linked worldwide. Meanwhile Internet has bought high speed communication network in every corner of the world spurring the development of better conferencing solutions for remote Internet operations. Video conferencing, one such video operation, is quite useful in Educational Institutions as well as Medical sectors where time is a big constraint.

The basic scenario in a video conference is that people communicate with each other being stationary i.e. they cannot move. In case when they want to monitor the surroundings or something has to be shown, the view point of the camera being fixed makes it difficult for the users on the both sides to effectively interact with each other. If the person moves, the camera and the display do not move in traditional systems, which cause disengagement in the conversation. General solution to this problem is addressed by using robotic platform that can be controlled remotely with cameras and microphones for videoconferencing. This solution is altogether called as telepresence system.

Telepresence can be defined as a set of technical methods which allows a person to feel as if he is present at place other than his true location. Commercial telepresence robots are focused on the concept of telecommunication using robotic platforms that can be remotely controlled using smart-phone or laptops. These systems are not affordable to general public, but are more for masses laypeople like IT professionals, executives, engineers etc. The telepresence system proposed and built in this paper is called as "Isaac". Isaac is an open source telepresence robot built using off-the-shelf hardware and open source tools to make it cheap and affordable.

This paper discusses implementation of a telepresence that is built using some of the affordable off-the-shelf hardware and open source software modules and API's. Furthermore this paper also discussed the use of WebRTC technology for video conferencing and remote control of robotic platform. This paper is organized as follows: Section 2 provides a brief description of related work done in recent past. In section 3 will discuss the overview of "Isaac", the telepresence system we have designed. Section 4, describes the hardware and software details of the robot and finally we have discussed how we implemented the telepresence system using the technologies stated in the various sections.

2. RELATED WORK

The application of telepresence has been receiving significant attention from the robotics community. The concept of telepresence was first proposed by Marvin Minsky, who was the founder of MIT's artificial Intelligence laboratory in the OMNI magazine in 1980 and proposed various needs and applications. In an effort to create mutually immersive communication using telepresence robots, HP built its early version of Surrogate in 2002, which percept audio and video

of being in a remote location. Aided by improved technologies in network communication, Cisco launched Cisco Telepresence video conferencing system in 2006. The system provide high definition video and realistic sound displays, allowing remotely connected users feel as if they are in the same room.

In the recent years more and more telepresence products have been pushed in the market, notably the Anybot QB, a commercially available telepresence device that connects through the Internet. Following the development of telepresence various people and organization started making semi-automated telepresence systems that included various modifications such as person following system, automation of audio-visual attention and overcoming various occlusions.

Akansel Cosgun et al.[1] presented a method for a mobile robot to follow a person autonomously where there is an interaction between the robot and human during following. The remote user connects to the robot via wireless internet and communicates with others using Skype. Nadia Magnenat et al.[2] discussed the various forms of telepresence, how they are related and different from each other and how autonomy takes place in telepresence.

Sina Radmard et al. [3] presented a blockage handling system, applied to telepresence platforms, that simultaneously reduces the control burden on the operator and ensures the continuity of the visual interaction. Katherine M. Tsui et al. [4] focused on how a telepresence robot operator, the people with the robot, and the robot itself collaborate so that the operator reaches his/her intended destination. Keng Peng Tee et al. [5] presented an audio-visual attention control system to establish focus of attention for a pan-tilt telepresence robot. Their system is able to detect attentions automatically and reliably in a telepresence scenario.

The backbone of all the above mentioned telepresence systems include video conferencing capabilities. Most of these mentioned systems use Skype, Face-time or some other video-conferencing tools that follow the client-server model for their transfer of audio-video streams. Since there is usage of servers for handling of data, these systems have a single point of failure i.e. the servers used for video-conferencing. To overcome many such issues and lessen the dependency on servers, Google released an open source project for browser-based real-time communication known as WebRTC in May 2011. This has been followed by ongoing work to standardize the relevant protocols in the IETF and browser APIs in the W3C. Since the need of servers is very minimal, user experience is not dependent on servers anymore.

Since its birth in 2011 various developments have been done in making WebRTC available to everyone. Sergej Alekseev et al. [6] presented an approach for analyzing the behavior of WebRTC - based applications, typically being used for direct audio or video browser-to-browser communication. Alan Johnston et al. [7] illustrated and discussed a number of issues that are specific to WebRTC enterprise usage. Deploying multimedia communication based on WebRTC API for more than browser-to-browser connection is not inherently supported and creates new challenge in terms of media stream distribution as well as membership and conferencing operation control overhead. Many conference models presented by Wajdi Elleuch[8] can support multi-party communication between end- points over an IP networks.

Jukka K. Nurminen et al. [9] reviews whether the new HTML5 and WebRTC standards are fit for P2P video

streaming evaluates the performance challenges and proposes solutions. Finally, they provide preliminary experimental data on WebRTC which measures the performance of such a system in a laboratory environment. Now we move on to a simpler way of implementation of telepresence system using WebRTC and Intel's mobile processor board in the next section.

3. OVERVIEW OF DESIGNED SYSTEM

Isaac is basically an open source telepresence robot that has been developed with a motive to make the telepresence system more affordable to all. The system comprises on one side, of a robotic platform, to which a smart-phone or tablet has to be connected via Wi-Fi hotspot and on the other side a smartphone/tablet/desktop that can be used as controller of the robotic system. The smartphone/tablet acts as the eyes and ears of the robot, by using its camera and microphone and sending the feed to the robotic platform. The robotic platform basically has an Intel mobile processor board, which hosts a WebRTC server. The same microprocessor, using some extra hardware controls the actuation of motors. The robotic platform is driven by two motors, as a differential drive mechanism.

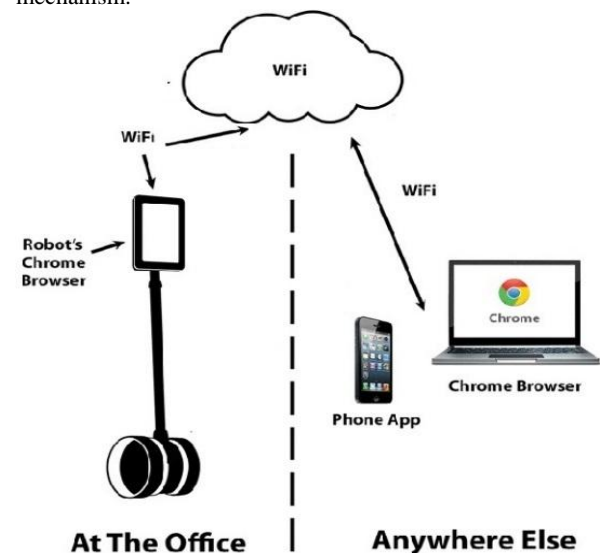


Fig.1: Basic scenario of Isaac telepresence system.

The processor board on the platform and the smart-phone has to be connected in the same Wi-Fi hotspot. The WebRTC server on the processor board has a static IP address that has to be entered into the smart-phone's browser. This opens a website hosted on the processor that uses the camera and microphone of the smart-phone.

The user, on the other hand, can connect to the robot using the IP address of the same processor board. The system being in development stage works only in same Wi-Fi hotspot region. So when the user enters the IP address of the robot into the browser on smartphone/tablet/desktop, a webpage asks for authentication details. This ensures the robot is used by only restricted number of known people. The user is displayed with a video feed from the robot and some control buttons. Using this control buttons the user can drive the robot. Also on the, side of the robot, the browser on the smart-phone/tablet displays the video feed on the user's device. Thus the user can now video conference and drive the robot simultaneously.

Fig.1 shows how the Isaac telepresence system works. In an office scenario a person can drive this robot to a meeting on some other place sitting at his desk. In the further sections we

will be describing the hardware software details of the systems working. Implementation of Isaac requires a wireless network via which the semi-autonomous robot can be controlled from a distance. It is a drive-by-wire system which enables a user to log into the robot and control it. A user can simultaneously video conference and drive the bot thus increasing the Field of view.

4. IMPLEMENTATION

4.1 Robotic Platform

The robotic platform is made up of three major components: Intel's Edison microprocessor board, Motor driver shield and motors. Apart from the platform, houses a battery pack and a charging circuit. As discussed before, the Intel's Edison board is responsible for hosting the WebRTC server and controlling the actuation of the robot.

Intel Edison is a tiny computer offered by Intel as a development system for Roboticians. The development board contains a dual-core Intel Quark x86[2] CPU at 400 MHz communicating via Bluetooth and Wi-Fi.

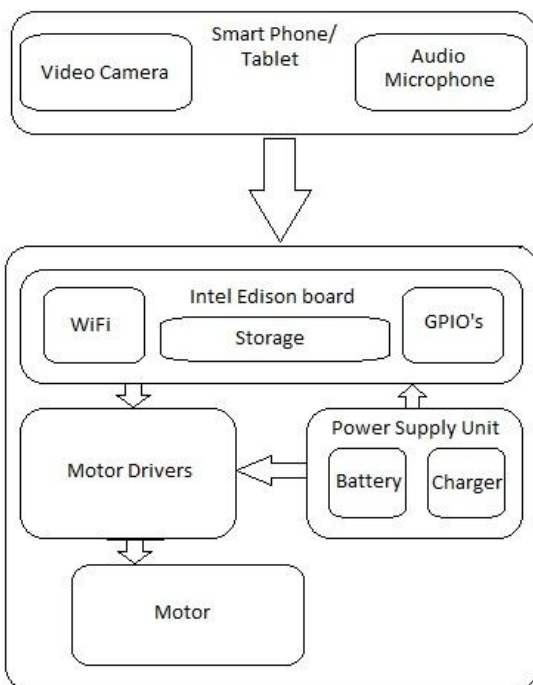


Fig.2: Hardware Organization of Isaac telepresence system

This development board is of 35.5 x 25 x 3.9 mm size making it the smallest available computer in the market. One can easily upload a Linux distribution on the board and use it for various applications. Edison board has Arduino compatible GPIO pins making it easy to use. We developed a motor driver shield that can be easily plugged into the board. A Shield is basically modular circuit board that piggybacks onto development board to install it with extra functionality. A motor driver is responsible to manage motor actuation and power requirement.

Fig. 2 shows how the hardware system is organized into the Isaac telepresence system. In our prototype we have used general geared DC motors to actuate the robotic platform. There are two motors mounted to give a differential drive mechanism for the robot to maneuver. The Edison connects to

the mobile hotspot directly, if configured during the initial stages, to the Wi-Fi hotspot and has a static IP address

The user can enter the static IP address in the browser and place it on the robot as shown in Fig. 3.



Fig 3: Isaac prototype with a smartphone.

4.2 Software overview

In recent years, we have seen new platform for development of real time communication: The Browser embedded applications or Web applications. Among these, WebRTC is one which enables direct peer to peer multimedia communication between browsing peers. WebRTC is free, open source project that enables real time communication via using JavaScript API's. Meanwhile with the increased bandwidth now it is possible to communicate at high speeds with peers, enabling WebRTC applications to stream data at higher data rates. By using WebRTC and HTML5, we can easily achieve a communication channel between two peers in real-time without the need of the intermediate server, making the video conferencing system more affordable.

For the implementation of Isaac, we used EasyRTC which is an open source distribution of WebRTC, with specialized API's for noise cancellations for perfect video conferencing experience. The EasyRTC is JavaScript based library which can be compiled using nodeJS which is an event-driven I/O server-side JavaScript environment based on Google chrome's V8.

The robotic platform is equipped with several software modules hosted on Clayton Linux distribution on the Edison board. Algorithm 1 is basically a program sequence to make the telepresence robot run smoothly while at the same time receiving command from the user under web application.

The nodeJS basically hosts an https webserver which binds itself to port 80. Whenever a client enters the IP address of the robot, which is static, an option is prompted asking if the device is to be kept on the robot or is being used for controlling the robot. On the selection of the option the user is redirected to the corresponding page.

ALGORITHM 1:

Isaac control algorithm

Require: Internet connection

While connect_to_internet do

Begin

Accept access control from remote user

Stream video and audio

If input keyboard or button detected
then

Receiving signal from remote user

Sending signal to motor controller

then

do move

else

stop

end if

end while

5. CONCLUSION AND FUTURE WORK

We have presented an alternate approach for implementing real time communication with open source hardware and software modules that make the system more affordable. Isaac is the first iteration of generations of telepresence robots to come in near future. The lessons learned while developing Isaac laid foundations for next iterative design. Isaac telepresence enable a user to implement the full scope of video conferencing as one can freely maneuver the robot from one location via the Internet.

Our further interests are focused on making this system autonomous and more innovative by adding new features like semi-autonomous navigation feature and facial tracking system to make it more interactive.

6. ACKNOWLEDGMENT

We would like to thank Prof. R.T. Nemade and Prof. D.P. Gaikwad for contributing in the implementation and designing of the system and guiding us through the project.

7. REFERENCES

- [1] Nadia Magnenat Thalmann , Zerrin Yumak , Aryel Beck "Autonomous Virtual Humans and Social Robots in Telepresence", 2014 IEEE 16th International Workshop on Multimedia Signal Processing (MMSP), Jakarta, Indonesia, Sep. 22-24, 2014
- [2] Sergej Alekseev, Christian von Harscher, Marco Schindler, "Finite State Machine based Flow Analysis for WebRTC Applications", 978-1-4799-4233-6/14/ 2014 IEEE
- [3] Florian Rhinow, Pablo Porto Veloso, Carlos Puyelo, Stephen Barrett, Eamonn O Nuallain, "P2P Live Video Streaming in WebRTC", 978-1-4799-3351-8/14/ 2014 IEEE
- [4] Katherine M. Tsui, Adam Norton, Daniel J. Brooks, Eric McCann, Mikhail S. Medvedev, and Holly A. Yanco "Audio-Visual Attention Control of a Pan-Tilt Telepresence Robot", 2013 13th International Conference on Control, Automation and Systems (ICCAS 2013) in Kimdaejung Convention Center, Gwangju, Korea Oct. 20-23, 2013.
- [5] Akansel Cosgun, Dinei A. Florencio and Henrik I. Christensen "Autonomous Person Following for Telepresence Robots", 2013 IEEE International Conference on Robotics and Automation (ICRA) Karlsruhe, Germany, May 6-10, 2013
- [6] Alan Johnston, John Yoakum, and Kundan Singh, Avaya Inc, "Taking on WebRTC in an Enterprise", 0163-6804/13/ 2013 Communications Magazine April 2013
- [7] Wajdi Elleuch, "Models for Multimedia Conference between Browsers based on WebRTC ", 2013 Sixth International Workshop on Selected Topics in Mobile and Wireless Computing, 978-1-4799-0428-0/13/ 2013 IEEE
- [8] Jukka K. Nurminen, Antony J. R. Meyn, Eetu Jalonen, Yrjo Raivio and Ra'ul Garc'ia Marrero, "P2P Media Streaming with HTML5 and WebRTC", 978-1-4799-0056-5/13/ 2013 IEEE
- [9] Sina Radmard and Elizabeth A. Croft "Overcoming Occlusions in Semi-Autonomous Telepresence Systems", 978-1-4799-2722-7/13/ 2013 IEEE