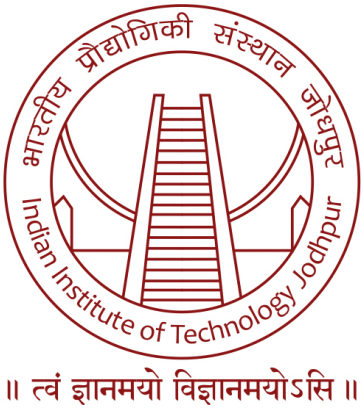
Telepresence in Virtual Reality

A Project Report Submitted by

Rajat Rawat

in partial fulfillment of the requirements for the award of the degree of

Masters of Technology in Computer Science



Indian Institute of Technology Jodhpur

Computer Science Engineering

May, 2023

Declaration

I hereby declare that the work presented in this Project Report titled Telepresence in Virtual Reality submitted to the Indian Institute of Technology Jodhpur in partial fulfilment of the requirements for the award of the degree of Masters of Technology in Computer Science, is a bonafide record of the research work carried out under the supervision of Sumit Kalra. The contents of this Project Report in full or in parts, have not been submitted to, and will not be submitted by me to, any other Institute or University in India or abroad for the award of any degree or diploma.

Signature

Rajat Rawat

M21CS014

Certificate

This is to certify that the Project Report titled Telepresence in Virtual Reality, submitted by Rajat Rawat(M21CS014) to the Indian Institute of Technology Jodhpur for the award of the degree of Masters of Technology in Computer Science, is a bonafide record of the research work done by him under my supervision. To the best of my knowledge, the contents of this report, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Signature

Sumit Kalra

Acknowledgements

I want to acknowledge and express my sincere gratitude to Dr. Sumit Kalra, my advisor, who has been a wonderful source of support and encouragement throughout the entire journey of my thesis research. His assistance was very important during the research and writing phases of my thesis. This study has been successful due to his contribution to the research efforts. I am very much thankful to my parents as well, for their continuous emotional and moral support towards my bad time of life. This outcome would not have been accomplished without their motivation and assistance. I would like to express my appreciation to the friends i had made at my college and beyond. I liked their availability anytime I required advice. I want to show my gratitude to all of my family members for encouraging and supporting me whenever I needed them.

Abstract

With the advancement of technology, the interactivity also increases with time. Keeping this in mind, we proposed an interactive and immersive solution in the field of Telepresence which revolves around the concept of Virtual Reality. We come up with a solution which strike off the presence of humans in many tasks with the concept of Telepresence. There exist many problems ranging from exam invigilation, Outpatient department rounds, walkthroughs or tours of places, giving training to someone etc., which requires the presence of humans, but with the use of Virtual Reality also called as Telepresence, we solved this problem, in which there is no intervention of human required in invigilance purposes, everything can be done by sitting at a remote location in real time, and the tasks can be done virtually with head mounted devices such as Microsoft HoloLens, Google Glasses, Meta Oculus, etc. In our solution and for the demonstration purpose we used Meta Quest. When we are seeing something virtually, its better if the view itself is 360 degrees, which gives us the immersive experience of the entire room, like in the case of exam invigilation we require complete 360 view of the entire class, due to all these dependencies, we involved the concept of 360-degree view. As all these tasks involves no relation between the actual environment and the human, so the entire concept is done wirelessly, because in real life, human can be at remote location with no interaction with the actual environment where these tasks are happening in real time. In this, we had done Live Streaming of a 360-degree camera, namely Ricoh Theta V, and saw its live streaming in Meta Quest. Also, keeping the concept of remote location in mind, we made all these setups wirelessly, so that anyone who is in the remote location can see the 360 degree view of the actual environment in real time seamlessly. Also, to make it more interactive that depending upon the movement of the human wearing headset, as the headset orientation changes, we store those orientation of headset in Firebase Database and display those orientation on a Raspberry pi, connected to the same network with which the headset is connected, hence in real time we are able to see the headset's orientation changing.

Contents

List of Figures

List of Tables

Telepresence in Virtual Reality

1 Introduction

Virtual Reality is a very fast-growing technology and is used to create and make the feel of interactive-ness or immersive-ness. It has a very wide use and can be used in almost all the other domains of Computer Science, not only in science, even these can be used in medical also, one obvious use is Virtual Autopsy, as sometimes we don't have a dead body, but we need to teach students about the basics of Autopsy, there the concept of Virtual Reality comes to the rescue, as we can create a virtual body, and we can teach Autopsy on that virtual body, and it not only completes the basic requirement of a body, but also increases the interactivity, as it creates a more immersive experience to students, and it feels like they are operating a real body, in a real environment. With Virtual Reality, we can create a virtual environment also, which gives the feeling as we are actually present in that virtual environment. Virtual Reality can be experienced with the help of Head Mounted Devices such as Meta Quest, Google Glasses, Microsoft HoloLens, etc., each and every device has its own advantages as well as limitations. The best part about Virtual Reality is that, even by sitting in one single place we can experience the virtual tour of any place in the world, and it gives us the look and feel of as if we are actually present in that world. Virtual Reality is also used in gaming, as it comes with controllers, which are used in playing games, those controllers act as our virtual hands, with which we can interact with the objects present in the virtual environment. Also, with Head Mounted Device, we can also see the 360-degree view of the environment, which increases the level of interactive-ness to greater extent.

Augmented Reality is also an immersive experience but it does not create any virtual environment, so there is not that much immersive-ness compare to that of Virtual Reality. The main difference between Virtual and Augmented Reality comes from the fact that, Augmented Reality does not provide any virtual environment, still it is widely used in various applications like product displaying, suppose we want to buy a sofa online but we don't know whether that sofa fit in our room or not, so to make sure it before buying, Augmented Reality gives us the flexibility, that in the phone itself on which we are viewing that sofa, we can see how that sofa fits in our room, for that we just need to put our phone camera on that direction where we need to put that sofa in room, and hence it will show us how well that product will fit in our room, but these all augmentation happens in our phone only, and there is no virtual environment created in this case. This is called as augmentation because the computer-generated image is superimposed into the real world, as in our case the computerized image is sofa and the real world is our room. In order to experience Augmented Reality, one does not need any high-end configuration device, only the smartphone which supports Augmented Reality would be suffice to experience it. One other major application of Augmented Reality is in the field of education. Like we can teach Alphabets, in interactive way in which for each character we can show the actual object which start from that character which not only teach students alphabets but also, they are able to learn easily and interactively, similarly we can also teach complex terms like Human Ear, Brain, etc., in much more interactive way.

Telepresence is one of the most important application of Virtual Reality, which is also the current need of the hour, as there are many problems which require human presence, but with the concept of Virtual Reality, the dependency of physical presence of a human gets eliminated. Tasks like exam invigilation, Outpatient department rounds, giving training about parts of machine to a new employee, giving tour of a place to someone, all these tasks require physical presence of human right now. Keeping this in mind, we introduced the concept of Telepresence which uses Virtual Reality under the hood, this will ultimately strike off the presence of human in these tasks, thereby reduces the dependency of human, as for example, in Outpatient department rounds, doctors need to perform rounds so as to see each patient and hence in this case doctor need to be present there physically in order to see all the patients, but with the help of Telepresence, even if the doctor is located at a remote location, can easily be able to invigilate all the patients remotely, and it gives the feel as if they themselves are invigilating the patients physically.

Live Streaming is an essential component in Telepresence, as all in all, everything boils down to what the person is seeing virtually, is nothing but a live feed of that actual environment. So, in order to make a seamless and smooth view of the actual environment, Live Streaming is the most important and crucial component in Telepresence, which revolves around the concept of Virtual Reality. For Live Streaming there are many cameras available but the one which able to capture the environment in 360-degree, is well suited for our purpose, as the Head Mounted Device such as Meta Quest, requires spherical image or video stream, and with 360-degree camera like Ricoh Theta V or Ricoh Theta Z1, we can able to see the 360-degree view of the environment because 360-degree wraps the entire environment in the form of sphere and 360-degree camera is a two-sided camera which captures that environment and Head Mounted Devices are used to view these Spherically wrapped images or videos in an immersive or interactive way.

Here we had taken the Live Streaming from a camera and saw that live stream in a Head Mounted Device wirelessly. For Live Streaming we used ***Ricoh Theta V*** which captures the Live feed of the environment where the camera is actually present in a 360-degree way. In order to view this live feed, we used ***Meta Quest 1*** and ***Quest 2***. Also, keeping in mind the main idea behind all this is to eliminate the physical presence of human and make the virtual environment so that it gives the exact look and feel of as if all of these happening in an actual environment in real time, we made all these setups wirelessly, so that anyone who is in the remote location can see the 360-degree view of the actual environment in real time seamlessly.

In the further sections, we discussed the ups and downs, advantages and limitations of each and every approach we used for our task. Right from the beginning to the end, we discussed why we used that approach, what outcome we get from it, how much time it took to get the result, what are the limitations of that approach, which acts as a motivation to think of a new approach and rectify the previous approach. Each and every approach involves some novel idea and by implementing each idea we step by step build our path to reach to the final result.

Further, we talked about how we fetched the headset's coordinates and depending upon the movement of a person the coordinates of headset changes and we send those coordinates to Firebase Database and

from there we fetched those coordinates into Raspberry pi. Hence we show, how the real time movement of a person who is wearing a headset got tracked and the corresponding changed values got displayed in Raspberry pi, which in turn indicates how the Telepresence work in Virtual Reality.

2 Motivation

Consider a scenario where the doctor need to see all the patients, hence the doctor every day goes on a outpatient department round in order to see the condition of each of its patients, hence in this case the doctor need to be present there for all the time, which ultimately counts as a dependency of physical presence of a person at that location.

Another scenario is in the education field, like in heavy mechanical devices, if we want to give operational training of that machine to a student or to a new employee, rather then being present over there, the teacher can teach all these concepts through a remote location also, so in this case, student need not depend on the instructor to be physically present over there.

One most important aspect comes in hazardous environments, like in case of hostage situation or in military operations, the soldiers need to go in the room, there-by putting their life on risk, without knowing what will be present there, which ultimately puts life of both, hostage as well as soldiers in danger, so in order to get rid of this extremely dangerous situation, telepresence can be used, in which we can invigilate the room and can easily see, where the hostages are tied, also it helps soldiers to know in which room the terrorists are present.

Keeping in mind about these problems, helps us to build a motivation, which ultimately reaches to the idea of creation of telepresence with the involvement of 360-degree camera for entire view and a Head Mounted Device for better immersive and interactive results.

3 Problem definition and Objective

Problem is to stream the live feed of a remote location through Ricoh Theta V 360-degree camera, and view it in Virtual Reality with the help of a Head Mounted Device in a real time environment and depending upon the movement of person, as the coordinates of headset changes, those changed values should indicate in which direction the person is moving, which gives us the feel of as if, we are seeing it by actually being present over there.

The main objective of this problem, is to strike off the physical dependency of a human, so that one can able to take or provide its service even by being present over a remote location.

Another objective is to make everything quite immersive and interactive by creating a virtual environment, which provides us a feel of as if we are physically present in that environment for which a Head Mounted Device is used to display all the virtual environment contents.

We need to provide all these objectives in a real time environment with smooth and clear stream, because in certain applications like mechanical machines operational training, clear visibility of the actual

environment is a must, so as to properly see each parts of the machine.

Keeping the problem statement in mind, we wirelessly live stream the contents of actual environment in a virtual environment in real time, and as the person wearing headset moves, correspondingly we will tell in which direction the user is moving along with the coordinates where the user moves.

4 Methodology for Live Streaming

In order to reach to the final end result, we encountered many challenges in our approach, then we figure out the solution of that challenge, that solution itself has its own advantage and disadvantage, then we rectify its shortcoming and proposed a new solution with some better alternative in each and every phase.

Below are the sequence of approaches we used from start to end, and for each approach we discussed why we used that approach, what outcome we get from it, how much time it took to get the result, what are the limitations of that approach, which acts as a motivation to think of a new approach and rectify the previous approach. Each and every approach involves some novel idea and by implementing each idea we step by step build our path to reach to the final result.

4.1 Live Streaming to YouTube Studio (Approach 1)

4.1.1 About the Approach

In this approach, we first of all connected our 360-degree Ricoh Theta V camera with the computer. Now, with the help of OBS Studio software, we captured the live stream of that camera inside OBS Studio. OBS Studio is a software which is use to make videos, live streaming from webcam or from any other external camera like Ricoh Theta V. The stream which we get is a 360-degree stream, as Ricoh Theta V is a 360-degree camera, and it produces spherical stream in output. After this, we presented the 360-degree live streaming of OBS Studio in YouTube Studio, by using the Stream key of YouTube Studio inside OBS Studio. Now, in order to see the entire 360-degree view in an immersive way, we used Meta Quest 1.

4.1.2 Why we used this Approach

This approach fulfills our requirement of Telepresence, in which we need to present the 360-degree view of a remote location and saw it on Meta Quest 1. So, anyone can see the live stream of a room in which camera is present, we just need to provide the URL of live stream, and with that URL, anyone can view that live stream, seamlessly.

4.1.3 Time taken for this Approach

It takes around one month to understand the Live Streaming architecture of YouTube as well as to implement this approach to get a fully functional 360-degree view of remote location using Ricoh Theta V on Head Mounted Display Oculus Quest 1, by capturing the video feed with OBS Studio and host it on YouTube server.

4.1.4 Limitations of this Approach

The only limitation with this approach is that, the server which we used in this approach is YouTube server, so rather than using someone else server, we should make our own server and host that video on that server, and from that sever which is running locally on our computer localhost URL, we can able to virtually Live Stream without using any external server.

4.2 Live Streaming to Twitch TV (Approach 2)

4.2.1 About the Approach

In this approach, we first of all connected our 360-degree Ricoh Theta V camera with the computer. Now, with the help of OBS Studio software, we captured the live stream of that camera inside OBS Studio. OBS Studio is a software which is use to make videos, live streaming from webcam or from any other external camera like Ricoh Theta V. The stream which we get is a 360-degree stream, as Ricoh Theta V is a 360-degree camera, and it produces spherical stream in output. After this, we presented the 360-degree live streaming of OBS Studio in Twitch TV, by using the Stream key of Twitch TV inside OBS Studio. Twitch TV is a streaming platform similar to YouTube Studio. Now, in order to see the entire 360-degree view in an immersive way, we used Meta Quest 1.

4.2.2 Why we used this Approach

This approach fulfills our requirement of Telepresence, in which we need to present the 360-degree view of a remote location and saw it on Meta Quest 1. So, anyone can see the live stream of a room in which camera is present, we just need to provide the URL of live stream, and with that URL, anyone can view that live stream, seamlessly.

4.2.3 Time taken for this Approach

It takes around 2 weeks to understand the Live Streaming architecture of Twitch TV as well as to implement this approach to get a fully functional 360-degree view of remote location using Ricoh Theta V on Head Mounted Display Oculus Quest 1, by capturing the video feed with OBS Studio and host it on Twitch TV server.

4.2.4 Limitations of this Approach

The only limitation with this approach is that, the server which we used in this approach is Twitch TV server, so rather than using someone else server, we should make our own server and host that video on that server, and from that sever which is running locally on our computer localhost URL, we can able to virtually Live Stream without using any external server.

4.3 Live Streaming using client and local Server (Approach 3)

4.3.1 About the Approach

In this approach, we made a server using Python, Flask in the backend. In the fronted we used HTML for displaying our 360-degree video content on local server. Then, we connect our 360-degree Ricoh Theta V camera with the computer and with the help of Open CV library of Python software, we captured the live stream of that camera inside our local server. Code is designed in such a way that it can distinguish between whether the webcam needs to be enable or the Ricoh Theta V needs to be enable. The stream which we get is a 360-degree stream, as Ricoh Theta V, is a 360-degree camera, and it produces spherical stream in output.

4.3.2 Why we used this Approach

This approach eliminates the drawback of Approach 1, which uses YouTube Studio as server, hence in this approach we are able to make our own local server. So, from that sever which is running locally on our computer localhost, we can able to virtually Live Stream without using any external server. So, anyone can see the live stream of a room in which camera is present, we just need to provide the URL of server which is doing the live stream, and with that URL, anyone can view that live stream, easily.

4.3.3 Time taken for this Approach

It takes around a month to make this client-server model as well as to implement this approach to get a fully functional 360-degree view of remote location using Ricoh Theta V, by capturing the video feed and hosting it on local server.

4.3.4 Limitations of this Approach

The main limitation with this approach is that, the live stream which is presented can't be seen in an immersive 360-degree manner in Meta Quest 1. This is because, the video is coming directly from the Ricoh Theta V, and is shown in 2-D manner on local server in the form of M-JPEG (Movable JPEG), but in order to see that stream seamlessly on Meta Quest, we need to convert that M-JPEG into mp4, but no conversion happened here, that's why even on Quest also, we saw the same streaming as we saw on our local server i.e, in 2-D view only, which is the major drawback of this approach.

4.4 Live Streaming using WebGL (Approach 4, Failed)

4.4.1 About the Approach

In this approach, we used Unity software to write the code in C Sharp language for Live Streaming. Then, we connect our 360-degree Ricoh Theta V camera with the computer and the code which we wrote, helps us to detect the presence of external camera. Then we build the application by switching Player to

WebGL component of Unity software, and on running it shows the output of the program in a tab of Web Browser.

4.4.2 Why we used this Approach

We thought that, this approach eliminates the drawback of Approach 1 and 2, which uses YouTube Studio and Twitch TV as servers, hence in this unity helps us to view the output in Web Browser and no server is required in this approach. Then we thought that by using Meta Quest with Link Cable we can easily see the 360-degree Live Streaming on Quest.

4.4.3 Time taken for this Approach

It takes around around one month to code this logic and showing the output in Web Browser using the WebGL component of Unity software.

4.4.4 Limitations of this Approach

The main limitation with this approach by which this approach is called as a failed approach is that, we can't be able to see the 360-degree Live Streaming on Web Browser using WebGL, this is because, WebGL only support Webcam on Web Browser, and no other external camera is supported by WebGL. By showing this result, this approach is considered as a failed approach as we can't be able to see even a 360-degree view of Ricoh Theta V camera, with this approach.

4.5 Live Streaming using Skyway server (Approach 5)

4.5.1 About the Approach

In this approach, we first of all connected our 360-degree Ricoh Theta V camera with the computer. Now, with the help of Skyway server, we captured the live stream of that camera inside Skyway server. The stream which we get is a 360-degree stream, as Ricoh Theta V, is a 360-degree camera, and it produces spherical stream in output. The Skyway server gives us the URL, so that it can be seen by anyone whosoever is connected on same network. Now, in order to see the entire 360-degree view in an immersive way, we used Meta Quest 1, so we connected the Quest 1 with Link Cable connected to the desktop, and in the Quest Browser we write the URL which we get from the Skyway server, and hence we are able to see the 360-degree live streaming of camera on Meta Quest 1.

4.5.2 Why we used this Approach

We used this approach so as to test whether this approach will give us good results in terms of resolution or not, because by looking at the results of this approach we need to go to the next approach, but since this approach won't give us good resolution output in 360-degree view, so we won't go to the next approach.

4.5.3 Time taken for this Approach

It takes around 2 weeks to implement this approach to get a fully functional 360-degree view of remote location using Ricoh Theta V on Head Mounted Display Oculus Quest 1, by capturing the video feed and host it on Skyway server

4.5.4 Limitations of this Approach

There are three main limitations of this approach, the first one is that the server which we used in this approach is Skyway server, so rather than using someone else server, we should make our own server and host that video on our local server, and from that sever which is running locally on our computer localhost URL, we can able to virtually Live Stream without using any external server.

The second limitation is that, there is heavy resolution issue and the 360-degree view coming is quite zoomed, and it is not clearly visible. This is because of network issue.

The third limitation is that, there is lag of around 8-10 seconds, this is again because of network issue, which is causing delay in live stream in output.

4.6 Live Streaming using Quest 1 Link Cable (Approach 6)

4.6.1 About the Approach

In this approach, we used Unity software to write the code in C Sharp language for Live Streaming. Then, we connect our 360-degree Ricoh Theta V camera with the computer and the code which we wrote, helps us to detect the presence of external camera. Now, in order to see the entire 360-degree view in an immersive way, we used Meta Quest 1, so we connected the Quest 1 with Link Cable to the desktop and output is visible in Quest 1 in 360-degree seamlessly in real time.

4.6.2 Why we used this Approach

This approach eliminates the drawback of Approach 1, 2 and 5, which uses YouTube Studio, Twitch TV and Skyway as servers, hence in this unity helps us to view the output in Unity itself and no server is required in this approach.

4.6.3 Time taken for this Approach

It takes around 2 weeks to implement this approach to get a fully functional 360-degree view of remote location using Ricoh Theta V and displaying it on Head Mounted Display Oculus Quest 1.

4.6.4 Limitations of this Approach

The first limitation of this approach is that, we need to connect Meta Quest 1 with the Link Cable to Desktop, hence this is not a wireless solution, which is the most important need for our problem.

The second limitation is that, it does not support Air Link, which means we cannot be able to connect Meta Quest 2 wirelessly to Desktop, hence the main requirement of our problem is not getting fulfilled.

4.7 Live Streaming using Quest 2 Air Link (Approach 7)

4.7.1 About the Approach

In this approach, we used Unity software to write the code in C Sharp language for Live Streaming. Then, we connect our 360-degree Ricoh Theta V camera with the computer and the code which we wrote, helps us to detect the presence of external camera. Now, in order to see the entire 360-degree view in an immersive way, we used Meta Quest 2, so we connected the Quest 2 with Air Link to the desktop and output is visible in Quest 2 in 360-degree seamlessly real time in a wireless manner.

4.7.2 Why we used this Approach

This approach eliminates the drawback of Approach 1, 2 and 5, which uses YouTube Studio, Twitch TV and Skyway as servers, hence in this unity helps us to view the output in Unity itself and no server is required in this approach. Also, it eliminates the drawback of Approach 6, which uses Link Cable, and hence this Approach uses wireless connection so we can easily able to see the 360-degree view in Quest 2 wirelessly in real time.

4.7.3 Time taken for this Approach

It takes around around 2 weeks to implement this approach to get a fully functional 360-degree view of remote location using Ricoh Theta V on Head Mounted Display Meta Quest 2.

4.7.4 Limitations of this Approach

The only limitation of this approach is that, there is no server involved, and the range of this approach depends upon the network range of the Wireless LAN to which it is connected with.

5 Methodology for Tracking Headset Coordinates

So, in the previous section we discussed how the Live Streaming was done, now we get the live feed of our surrounding in the VR headset, now next step is to get the coordinates of the headset or to track the headset coordinates position.

Now, in order to track the headset position, we had written a C sharp script which basically did our job, in which we tracked the position of our device in the VR space. Now, in order to store the values of the coordinates we used some variables to hold information about the VR device, its position, and whether or not the device position has been chosen.

Also we used method to get information about the device being used to track the VR space and gets the device for tracking the head position, and the current position of device is captured frame by frame.

Finally we displayed the changing coordinates of the tracked headset onto the screen in real time. With this we successfully able to track and fetch the coordinates of our headset.

6 Methodology for Sending Headset Data

Now, we have our headset coordinate data with us, but we want the data to be displayed on some other device, so for that we first require that the data should be saved on to a remote server, and from there we can fetch that data to some other device.

So, we head towards the Firebase console first and there we made a Real time Database where we provided the read and write values to true which means we can provide the values and the values can be updated in real time as the headset orientation changes.

Since, for storing data onto a remote database server we used Firebase database, initially we used the Firebase plugin in our code so that we can able to use the functionality of the Firebase Database. Next we made the reference or the object of the database and we passed our data to it, which contains the real time changing coordinate values of the headset whose values keep on changing as the user moves his/her head in any direction.

Now, as the headset is moving, data keeps on changing continuously and that data is being sent to Firebase and there also we are able to see the data changing continuously. Hence, from this we confirmed that our headset tracking data is successfully sent to Firebase database.

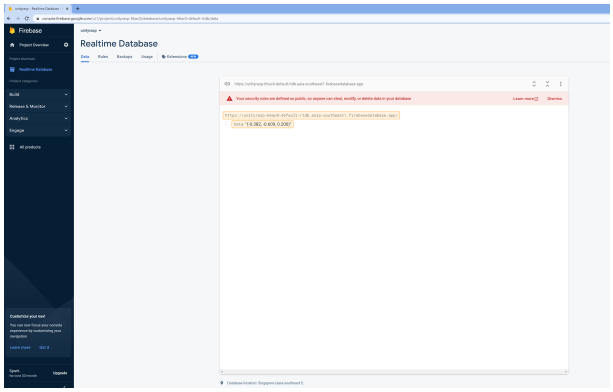


Figure 6.1: Capturing data in Firebase

7 Methodology for Retrieving Data

Now, our data is there in the remote database of Firebase, in order to retrieve that data in some other computer, we used Raspberry pi.

To retrieve data from database onto raspberry pi, we had written a python program. So the program is designed to read data from a Firebase real time database and tells in which direction the headset is moved. Program has a while loop that runs continuously. Inside the loop, the program retrieves the current data stored under the 'Data' key in the Firebase real time database some method. The retrieved data is then processed. The program then compares the new values with the previous values. If there is a change in any of the values, the program prints a message indicating which direction changed. If there is a change in any of the values the program updates the old value with the new values and prints

a message indicating the update.

Hence in this way we created a python program inside raspberry pi that is used for monitoring changes in our VR headset or tracking movements of VR device in real time.

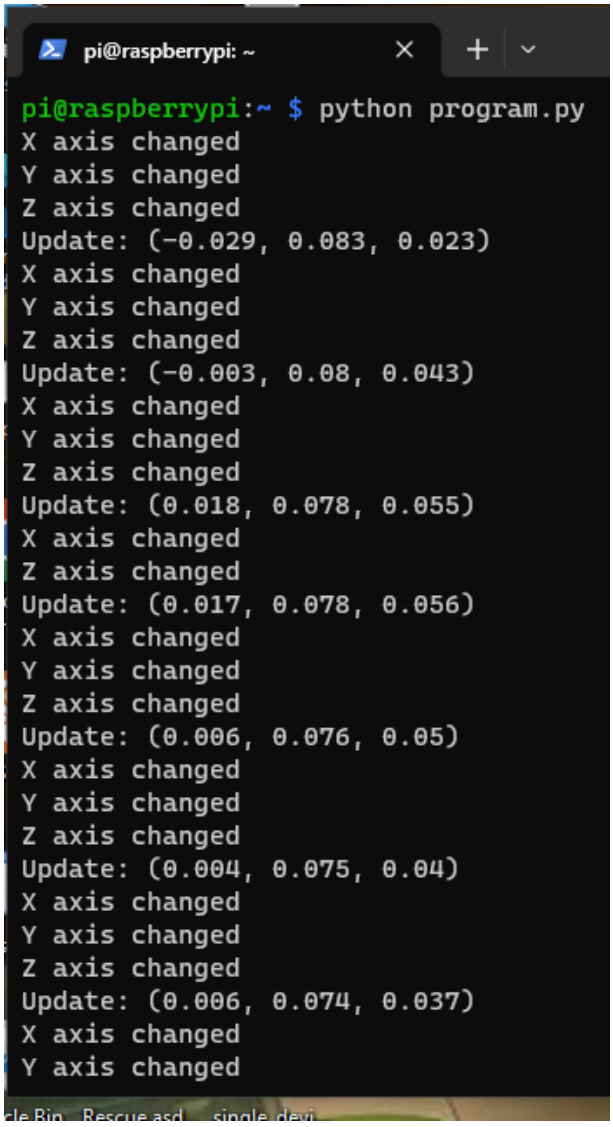


Figure 7.1: Retrieving data from Firebase

8 Experimental Results

Below are the screenshots of the result which we get from each and every approach right from the Approach 1 of YouTube studio to that of Approach 7 of Air Link in Quest 2, screenshot results of all the approaches are shown below.

8.0.1 Approach 1

This approach is done by using YouTube Studio as a server and the live stream of camera is captured inside OBS Studio, and the live streaming in OBS Studio is presented in YouTube Studio.

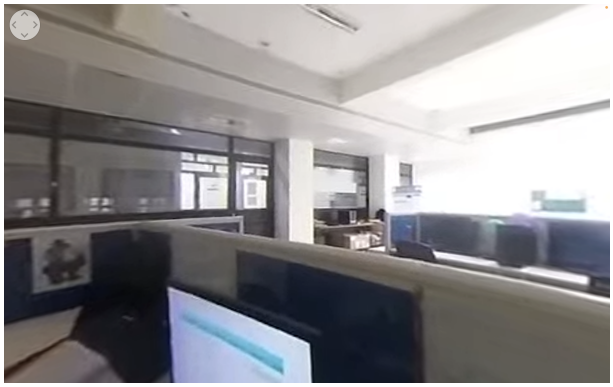


Figure 8.1: Approach 1

The picture is clear, but only drawback is that, we don't have a server of our own, we used YouTube's server.

8.0.2 Approach 2

This approach is done by using Twitch TV as a server and the live stream of camera is captured inside OBS Studio, and the live streaming in OBS Studio is presented in Twitch TV.

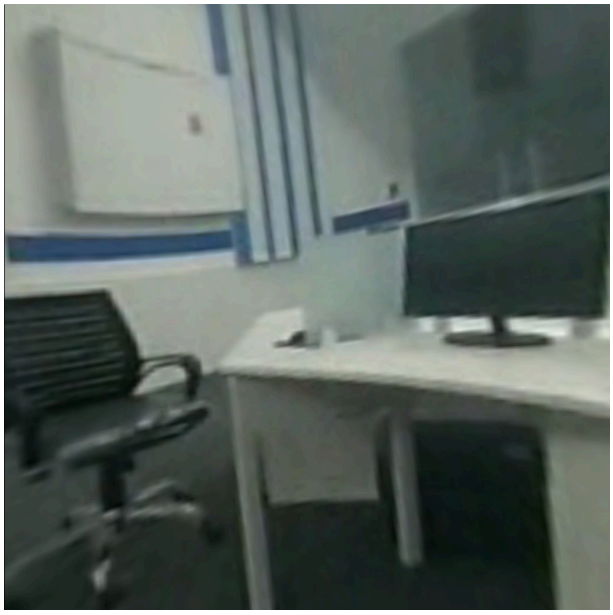


Figure 8.2: Approach 2

The picture is clear, but only drawback is that, we don't have a server of our own, we used Twitch TV's server.

8.0.3 Approach 3

This is done by making a local server, using concept of client server model, but here we don't have 360-degree view in VR Headset.

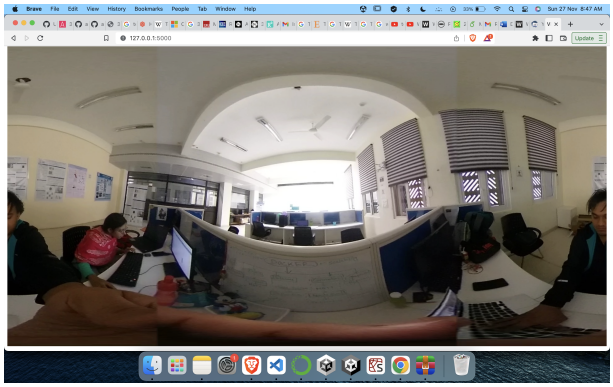


Figure 8.3: Approach 3

The picture is clear, but only drawback is that, we can't see its 360-degree immersive view in Virtual Reality Headset.

8.0.4 Approach 4

This approach is done by Live Streaming using Unity and WebGL platform on web browser.

Figure 8.4: Approach 4

The picture is clear, but major drawback is that, we can't even see the 360-degree view, because WebGL does not support any other external camera apart from Webcam.

8.0.5 Approach 5

This approach is done by Live Streaming the 360-degree Ricoh Theta V camera with Skyway server, and the streaming is then viewable in 360-degree in Virtual Reality Headset.

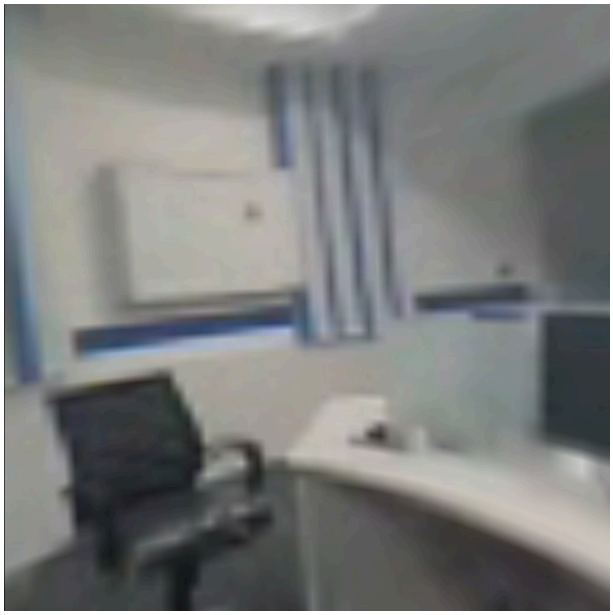


Figure 8.5: Approach 5

The two major drawbacks of this approah are that, first of all the picture is not all all clear it's pixelated, also, we don't have a server of our own, we used Skyway's server.

8.0.6 Approach 6

This approach is done by Live Streaming the 360-degree Ricoh Theta V with Meta Quest 1 using Link Cable.

Figure 8.6: Approach 6

The picture is clear, but only drawback is that, we have a wired connection of Meta Quest 1 with the computer.

8.0.7 Approach 7

This approach is done by Live Streaming the 360-degree Ricoh Theta V with Meta Quest 2 using Air Link.

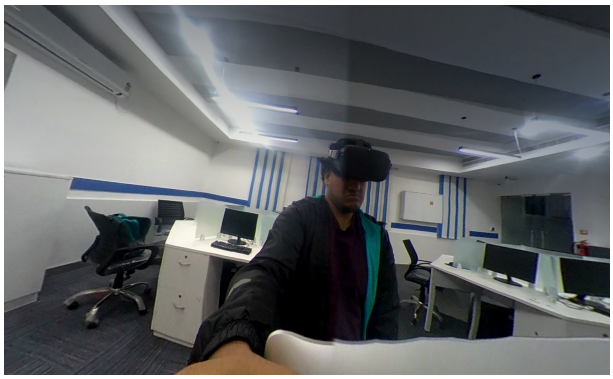


Figure 8.7: Approach 7

The picture is clear, as well as we have a wireless connection of Meta Quest 1 with the computer, however only the camera is in wired connection with computer, and the range of this approach depends upon the network range of the Wireless LAN to which it is connected with.

Approach	What used	Time taken	Limitation(s)
1	YouTube Studio	1 month	Used YouTube's Server
2	Twitch TV	2 weeks	Used Twitch's Serve
3	Client Server Model	1 month	360 view not possible in VR
4	WebGL and Unity	1 month	360 view not possible in VR
5	Skyway server	2 weeks	Skyway server, not clear and lag in VR
6	Quest 1 with Link Cable and Unity	2 weeks	Wired Connection
7	Quest 2 with Air Link and Unity	2 weeks	Only camera is wired, Bandwidth

Table 8.1: Tabulated comparative analysis of all the approaches used.

9 Conclusion and Scope of Work

In this model we proposed telepresence in Virtual Reality which is extremely important so as to remove the physical presence of humans in various tasks, which is the current demanding need of the hour. We saw various approaches which we used in order to Live Stream from 360-degree camera to Head Mounted Device. We used Ricoh Theta V as a 360-degree camera and Meta Quest 1 and 2 for Head Mounted Display. We proposed various approaches, ranging from using external server such as YouTube Studio, Twitch TV, Skwyay server, we also saw how we used the local server also, WebGL based approach, Quest Cable Link approach, and at the end we configured Quest 2 with Air Link, which ultimately gives us the wireless non server based result.

Also, we extended this model to advance level, in which we are moving in a room by wearing a Quest Headset on which we are seeing another room's architecture in 360-degree and as we are moving in our room, that movement got tracked and in some other computer we are able to see in which direction the movement is taking place. For this we tracked the values of our headset and send those values to a remote Firebase database server and from there we retrieved that data in raspberry pi which is connected with the same network with which the headset is connected. Hence the biggest scope of this work is that by moving in our room by wearing headset, feels like we are moving in some other room in order to do this, suppose we move in our room, we can use a motor vehicle which carries 360-degree camera in another room, so depending on our movement, the headset coordinate data got changed and it gets stored onto remotev database and then that data got fetched into raspberry pi which is present inside the motor vehicle, and that raspberry pi will give command to the motor vehicle that in which direction to move there by gives us the feel like we ourself are moving in that room in real time.