Hafeez Ullah, s8haulla@stud.uni-saarland.de (491772138718)

AR Teleportation

Augmented Reality Teleportation. The real environment is augmented by a virtual environment and if you walk into a virtual environment, it gives a feel of you are teleported into another environment.

Abstract

Ever since Pokémon Go popularized augmented reality apps, developers have been trying to build the "next big thing". The possibilities for virtual teleportation are endless. It's just one more way that the immersive tech revolution will upend our lives, transforming the way we work, learn, and play.
My project "AR Teleportation" is one of the similar ideas in augmentation reality. The basic idea is you are in your personal space which is a real environment and when you use this mobile application it augments the virtual environment in your real environment. Now walking into that virtual environment by holding a camera towards virtual environment direction. On the mobile interface, it feels like you are teleported into another environment. For the technologies I have used; Google cloud API, Unity (latest version), ARCore & Unity Portal. ARCore is an open-source tool to create augmented reality applications. Even though it's a free library, it provides a rather rich set of features for GPS tracking, supports both cameras. Android mobile version 7 at least is required to run the application on a mobile phone. In the end, I was able to develop an application with these technologies.

Background

Augmented reality is achieved through a variety of technological innovations; these can be implemented on their own or in conjunction with each other to create augmented reality. They include Augmented Software, Sensors, and input devices, hardware components.

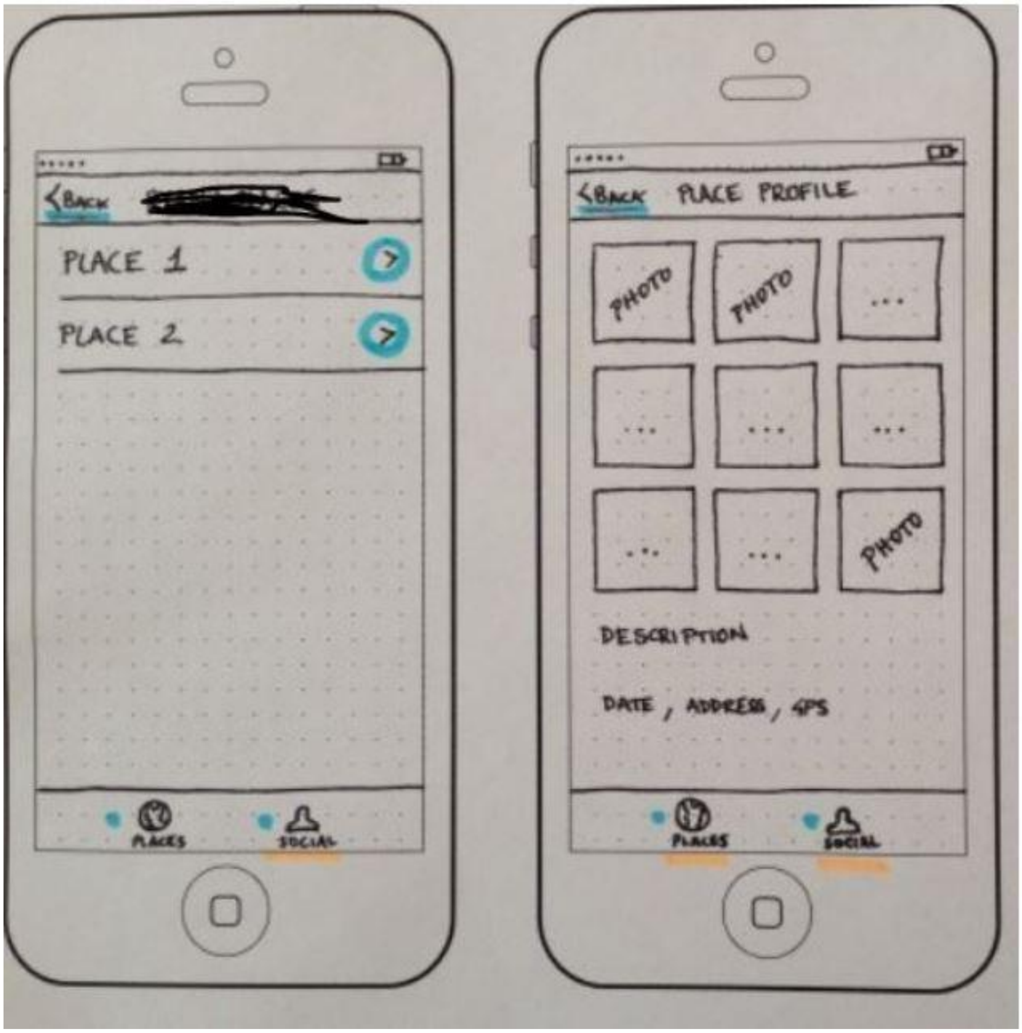
There has been some work done in this Teleportation field previously and some authors may come up with good results and there are very few papers that have been written on it as a technology is a new big thing.[1] Telecollaboration involves the teleportation of a remote collaborator to another real-world environment where their partner is located. The fidelity of the environment plays an important role in allowing corresponding spatial references in remote collaboration. We present a novel asymmetric platform, Augmented Virtual Teleportation (AVT), which provides high-fidelity telepresence of a remote VR user (VR-Traveler) into a real-world collaboration space to interact with a local AR user (AR-Host). AVT uses a 360° video camera (360-camera) that captures and live-streams the Omni-directional scenes over a network. The remote VR-Traveler watching the video in VR headset experiences live presence and co-presence in the real-world collaboration space. The VR-Traveler's movements are captured and transmitted to a 3D avatar overlaid onto the 360-camera which can be seen in the AR-Host's display. The visual and audio cues for each collaborator are synchronized in the Mixed Reality Collaboration space (MRC-space), where they can interactively edit virtual objects and collaborate in the real environment using the real objects as a reference. High fidelity, real-time rendering of virtual objects, and seamless blending into the real scene allows for unique mixed reality use-case scenarios. Our working prototype has been tested with a user study to evaluate the spatial presence, co-presence, and user satisfaction during telecollaboration. Possible applications of AVT are identified and proposed to guide future usage."

In another research [2]. Virtual reality systems typically allow users to physically walk and turn, but virtual environments (VEs) often exceed the available walking space. Teleporting has become a common user interface, whereby the user aims a laser pointer to indicate the desired location, and sometimes orientation, in the VE before being transported without self-motion cues. This study evaluated the influence of rotational self-motion cues on spatial updating performance when teleporting, and whether the importance of rotational cues varies across movement scale and environment scale. Participants performed a triangle completion task by teleporting along two outbound path legs before pointing to the unmarked path origin. Rotational self-motion reduced overall errors across all levels of movement scale and environment scale, though it also introduced a slight bias toward under-rotation. The importance of rotational self-motion was exaggerated when navigating large triangles and when the surrounding environment was large. Navigating a large triangle within a small VE brought participants closer to surrounding landmarks and boundaries, which led to greater reliance on piloting (landmark-based navigation) and therefore reduced - but did not eliminate - the impact of rotational self-motion cues. These results indicate that rotational self-motion cues are important when teleporting and that navigation can be improved by enabling piloting."

Relatable to my project is quite a bit similarity like in the idea of teleportation. Above mentioned paper technique is way advance in concept and implementation, but my AR teleportation idea is just a beginner level idea. Paper is using 360-degree view with wearable, but my project will focus on just mobile, you can get 360 views just but rotating camera in a virtual environment. But surely for future work, I can implement those advanced ideas.

Project Concept

A new addition to the augmented reality technology that lets you virtually teleport to locations around the world. A 360-degree visual interface to become "a new way of communication". Through this amazing technology, now it will be possible for you to travel to places across oceans without physically crossing the space. Or take a complete 360-degree virtual tour or your new office or home (even before it is built), sitting at your comfortable sofa. In the application, you save a few places in forms teleported location which is now extended concept. Right now, I am only focusing on 1 location just for the project and then you select that place where you want to teleport yourself, and then you do it by your mobile camera. ask someone to scan you and your mobile gives you feel of teleportation as it replaces your



background with your selected background. For now, it offers only 1 location, and future work multiple locations can be the option in extended work. you start the app and select a place which is, for now, will be only 1. can be added more lately. So after selecting a place it is like that place has been augmented virtually on your real environment and it is interactive in a way that it shows some entry point to that place like in a form of hallway or door entrance and when you walk to that entrance now you feel like you are in a new place but in reality, you are still at your place e.g home. but it gives a feel of some new place. The same way you can exit a place and yes it works like it. Experience in a new environment will feel like you are there, and it is 3D. Furthermore, I chose this project it comparatively was able with other options given on my wiki page. Indoor navigation was extremely time-taking and was very difficult and another project is anatomy which I found some students are already doing it. So, that's how I made my mind to do this project.

Project Implementation

Re-Tracing project implementation of AR Teleportation

1. Download Unity AR core package from AR Core SDK

ii. Start by adding setup a new project in it. Select 3D project

iii. Delete directional lights and cameras to make the environment clear

iv. switch to android. Go to file and build settings then click to switch to the android

v. Import AR core package. Go to Assets > import package>custom package and you select package here in your drive

vi. Disable multi-threaded rendering because it does not support it

vii. In XR setting enable AR Core support

viii. Disable real-time and mixed lighting in windows menu > lightning

ix. Create a new folder from assets it contains all the project details and implementation, name it any whatever you prefer "AR Teleportation"

x. Create a "scenes folder" in it so we can implement scenes in it. These steps finish setting for AR core

2. show the image produced by the camera

i. create a new game object and name it AR core device

ii. add AR core session from components. it is responsible for AR core session

iii. create session config because it is dependent. Create a new folder named "Config" inside AR Teleportation

iv. disable light estimation in config folder

v. Use session config with AR core session components

vi. Create a new camera as a child of Core AR device name it "AR Core camera". modified it as the main camera

vii. add track pose driver in its components and select color camera

viii. ad AR Core background render which is responsible for rendering the camera image on screen

ix. build device. edit > project settings>players>package name "Hello AR Teleport"

x. Go to file > add a scene in the project and click build and run and create build folder where APK will be placed in it

3. AR core and draw them on screen

i. create a controller that will manage our session create a new folder named " scripts" inside AR Teleportation

ii. create new c# file in script and name it "AR controller"

iii. create a new game object and name "AR controller" as well

iv. Attach AR controller script to game controller

v. Open AR controller script to use AR core related functions

vi. Write code in C# not writing here but in visual studio

4. Draw 3d mesh that represents plane

i. start by creating new 3d cube

ii. create new scripts that handle 3d mesh for grid visualizer and attach to grid game object

iii. copy contents of scripts into visualizer c#

iv. make a new folder 3D and import it from the drive. this is 3d door which is a path to teleportation world

v. attach camera and Ar portal to the AR controller script

vi. if you build a device you can place the door on top of any plane.

5. create the world behind Door and looks augmentation on real environment

i. I am going to use the sponza scene which is very popular in computer graphics

ii. Create a folder "texture " in the main folder

iii. create another folder "sponza" in which you store scenes.

iv. create another folder named "materials"

v. drag and drop 3d mesh into a scene. check as per requirement

vi. add sponza as a child of the portal now you should be able to walk inside a building. till now we are using unity default material for the portal and the building

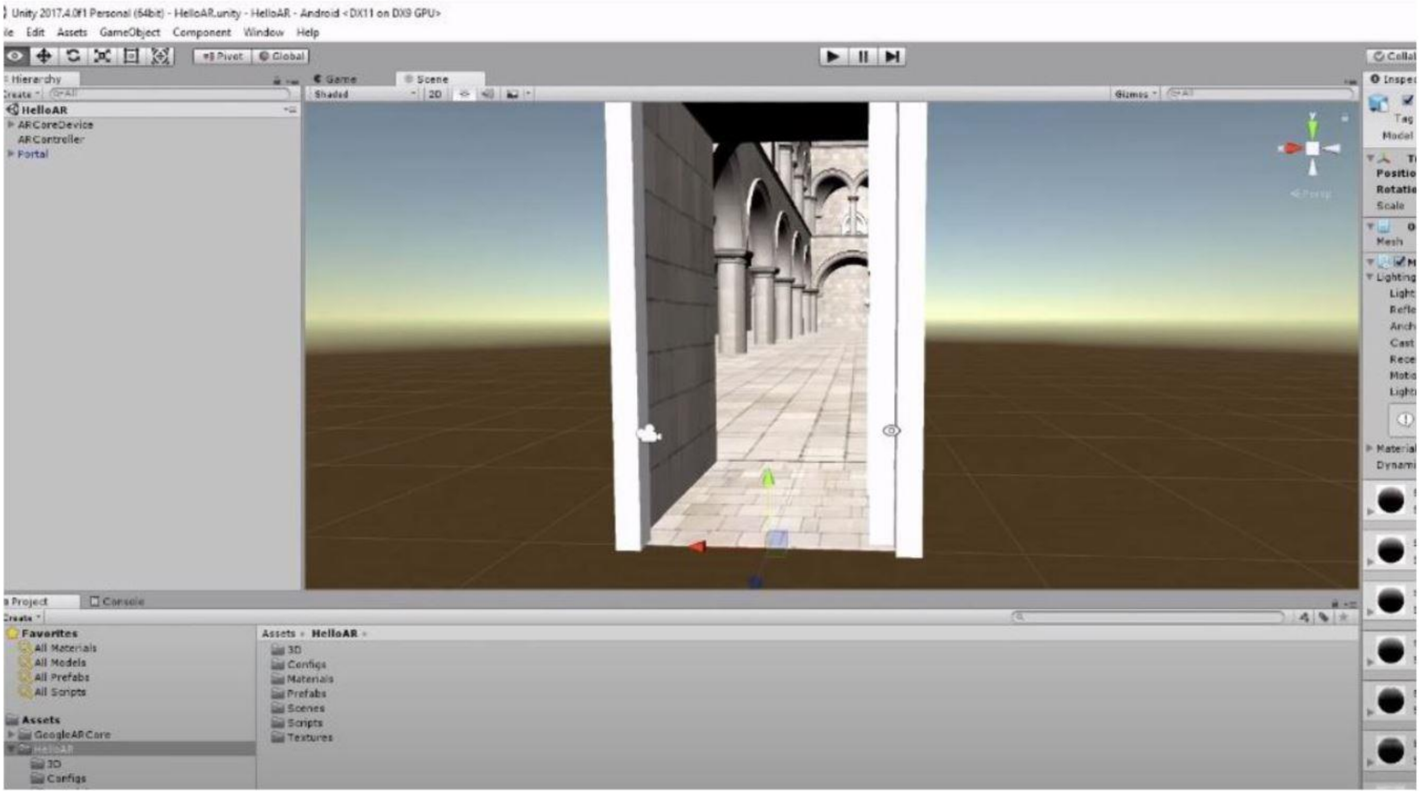
vii. copy shaders to materials eg sponza shader

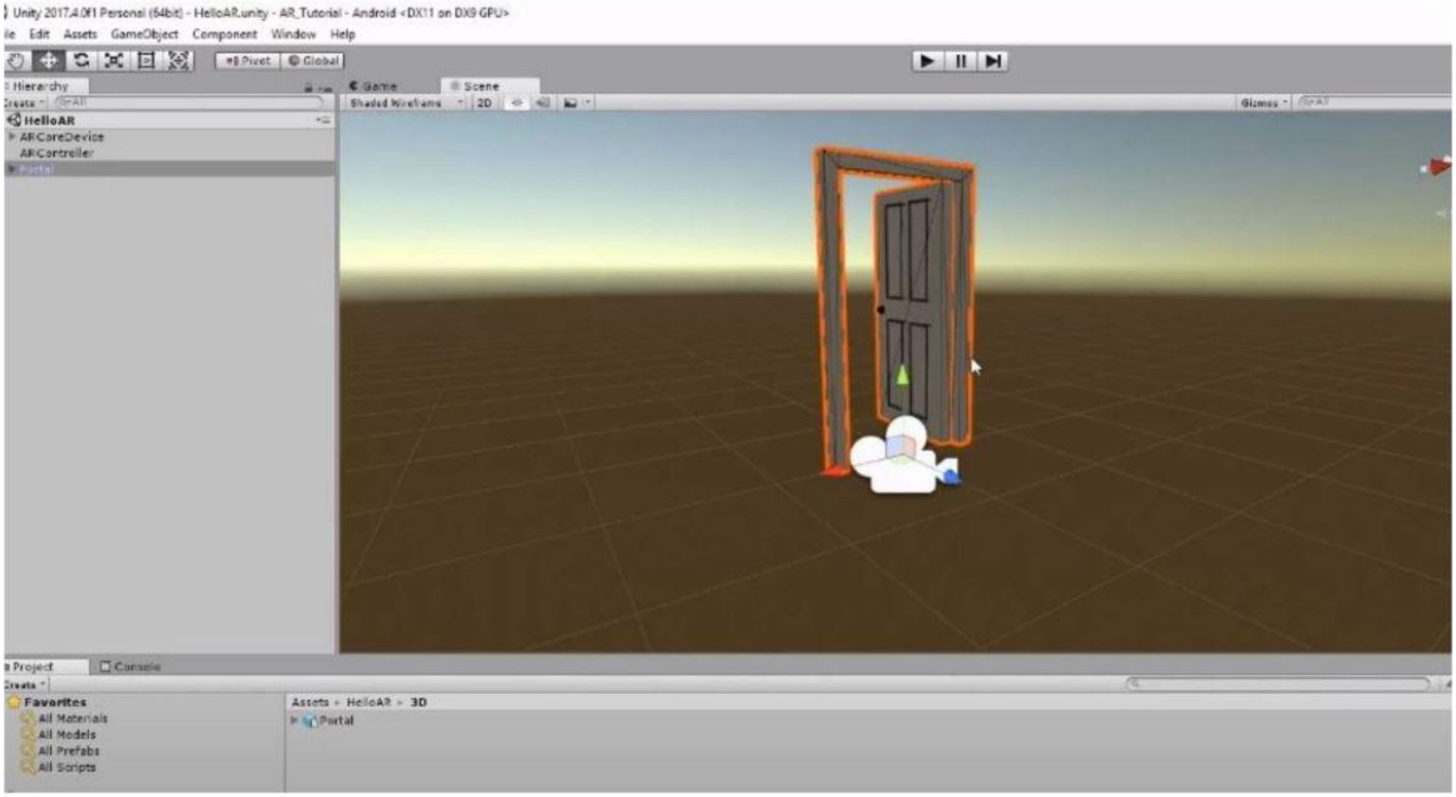
viii. use additional render target or use stencil buffer which is more optimal

ix. try to run the project. see for errors

x. now you can enter to teleport world with door

disclaimer: I am not explaining code. Written as per need


Unity 2017.4.0f1 Personal (64bit) - HelloAR.unity - HelloAR - Android <DX11 on DX9 GPU>
File Edit Assets GameObject Component Window Help
Pivot
Global
Hierarchy
Create
HelloAR
ARCoreDevice
ARController
Portal
Game
Scene
Shaded
2D
Gizmos
Project
Console
Favorites
All Materials
All Models
All Prefabs
All Scripts
Assets
GoogleARCore
HelloAR
3D
Configs
Assets > HelloAR
3D
Configs
Materials
Prefabs
Scenes
Scripts
Textures


Unity 2017.4.0f1 Personal (64bit) - HelloAR.unity - AR_Tutorial - Android <DX11 on DX9 GPU>
File Edit Assets GameObject Component Window Help
Pivot
Global
Hierarchy
Create
HelloAR
ARCoreDevice
ARController
Portal
Game
Scene
Shaded Wireframe
2D
Gizmos
Project
Console
Favorites
All Materials
All Models
All Prefabs
All Scripts
Assets > HelloAR > 3D
Portal

Hafeez Ullah, s8haulla@stud.uni-saarland.de (491772138718)

Design Decisions about building prototypes of AR Teleport project

Pathway to a virtual place that is augmented in a real environment. The idea of the app is to experience virtual places on your phone, and you stay in your place and experience virtual places in your comfort zone. It's like the idea of Pokémon Go where it is augmented in real-world environments. It provides the 360-degree experience of the virtual place which is augmented in the real environment. In the application, the user has a multiple place option where the user wants to have a virtual experience. Out of available places, options users select one. Now, virtually a door augmented in a real-world environment which is a pathway to the virtual selected place. User Enters the virtual environment through this door by walking in it. By walking into it, it presents the experience of having into a virtual world which augmented in a real environment. The virtual world is also a real-world place but in augmented form. It is possible to build this app by Unity and Vuforia and I 'll be using these tools to create this prototype application. There are few challenges as well in the prototyping of the application. Detection of Surface, need a plane surface to spawn 3D models of Door and virtual place, Creating multiple places in one environment.

Building a prototype requires a lot of design decisions and I have a simple design interaction for my application. I have an idea to keep it simple and easily understandable for the user of the application. I have come up after thinking critically that there are places where users can interact with my applications. there could be many designs to come with, but my thoughts to be stick with simple and easy to understand. So, there are 2 times when user interaction with design in my project. 1st is when interacting with the door. The door should be open so the user can enter it and experience virtual place augmented in a real environment. So, the door could be liked closed and you may knock it to open it, but it makes a bit tough for the user to understand what I should do with a closed door. so, I have come with the idea that the door should be open, and the user can see through the door about virtual location. there is no need to make the door closed. On the other side of the door, I plan to make a beach place for the user to have experience in a virtual place. Simple and understandable.

Hafeez Ullah, s8haulla@stud.uni-saarland.de (491772138718)

Git-Repository link of AR Teleportation

git repository linked in resources section.

Lessons Learned

1. Remember to think before you start coding! And plan before you start coding. Coding/programming is mostly about thinking of the best solution to solve a problem.

2. While you're coding you should think about writing beautiful code and comment a lot. The better the code is, the easier it will be to maintain in the long term.

3. Setup the coding environment and dependencies and SDK in advance. If even it takes 4 weeks to set up. Carefully do it to avoid any problems in a setup later in the end times of the project. This was the most difficult thing I did, and it took a lot of time for me.

4. Ensure that code is readable, Ensure it is not affecting the other programs.

The following are the key points learned while working in Unity. **So, it is specifically addressing Unity**

5. Decide on the scale from the beginning and build everything to the same scale.

If you don't, you may need to rework assets later (for example, animation does not always scale correctly). For 3D games, using 1 Unity unit = 1m is usually the best. For 2D games that do not use lighting or physics, 1 Unity unit = 1 pixel (at "design" resolution) is usually good. For UI (and 2D games), pick a design resolution (we use HD or 2xHD) and design all assets to scale in that resolution.

6. Make every scene runnable. Do this to avoid having to switch scenes to run the game so that you can test faster. This can be tricky if you have objects that persist between scene loads that are required in all your scenes. One way of doing this is to use make persistent objects singletons that will load themselves when they are not present in the scene. Singletons are described in more detail in another tip.

7. Always try to set up the following: GUI depths for layers, Scene setup, Build setup

8. Use assertions: Assertions are useful to test invariant in code and help flush out logic bugs. Assertions are available in Unity.Assertions.Assert class. They all test some condition and write an error message in the console if the condition is not met.

9. Code Documentation: Always document all your code and every step it helps you in the futures and for another person to easily understand your project

Vision and Outlook

There has been some work done in this Teleportation field previously and some authors may come up with good results and there are very few papers that have been written on it as technology is a new big thing. Through this amazing technology, now it will be possible for you to travel to places across oceans without physically crossing the space. Or take a complete 360-degree virtual tour or your new office or home (even before it is built), sitting at your comfortable sofa. Future work, for real-world applications, this feature can be introduced by airlines or tourism services. It works like a user can install an app for that airline or tourism services app, which could have this feature in their application. Users can experience an augmented environment given having many locations augmentation in that application. Users can have real-time experience and will help them to decide on their next trip. But they have to augment a lot of places and it should have the option to select one from many augmented locations. My application is at the basic level, it is possible to add a lot more location of different cities, beaches, airports, landmark places. There could be enormous categories and under that category, a lot of locations can be added. Then the user can choose any of the locations and can have an experience of teleportation.

Resources

Following is the link to my GitHub repository. It's on public.

https://github.com/hafeezullah05/AR-Teleportation_IAS

Video

Video on Google drive: https://drive.google.com/file/d/1IGVDOROBvJpxMzoxvD-nrAMG70edUIVp/view?usp=sharing

Download link: https://drive.google.com/uc?id=1IGVDOROBvJpxMzoxvD-nrAMG70edUIVp&export=download

References

[1] Taehyun Rhee, Stephen Thompson, Daniel Medeiros, Rafael dos Anjos, Andrew Chalmers. March 2020. Augmented Virtual Teleportation for High-Fidelity Telecollaboration. (March 2020). Retrieved March 24, 2020, from http://ieeevr.org/2020/program/papers.html#papers11

[2 Jonathan Kelly, Alec Ostrander, Alex Lim, Lucia Cherep, Stephen Gilbert .2020. Teleporting through Virtual Environments: Effects of Path Scale and Environment Scale on Spatial Updating. (March 2020). Retrieved March 25, 2020 from https://ieeevr.org/2020/program/papers.html#papers19