Instructions for running the code

We can get the Unity project file (named PointCloud) and the client script (named dowload.scp.ps1) for point cloud remote transfer in **Code of Unity** file in this repository or at the following URL
https://drive.google.com/drive/folders/1TOiRl-3IoJbEuXr7BsY0GNKiuseSH0jN?usp=sharing

Real camera control & Remote transfer - Server settings

First clone the first three Python files in **Pyrealsense2_ControlCameras**. They are *01 SingleCamContinuousOutput.py*, *02 ControlUsingZ&X-DoubleCamContiOutput.py*, *03 RemoteSingleCamContinuousOutput.py*.
The roles of these three Pythons are shown in the file names. The first two contain code for remote transfer of point clouds that is commented out, and if uncommented, can be transformed into code for camera export of point clouds that includes remote transfer functionality. The point cloud files exported by these three codes are in the local folder. Start one of the first two codes and set the Unity3D read point cloud path to that folder to read the point cloud exported by the camera in real time. Start the third code to transfer the point cloud exported by the camera to AWS server in real time.
It should be added that the other code in the Pyrealsense2_ControlCameras folder is mainly used for testing and camera functions TUTORIAL.

Remote transfer - Client settings

First, we download the dowload.scp.ps1 script under the path of receiving the remotely transmitted point cloud.

Then open powershell, cd to the path of dowload.scp.ps1, running the script, that is ***./dowload.scp.ps1 out***

Unity3D reads the point cloud

Use Unity to open the PointCloud project, and note that the Unity read path should be the location of the point cloud file.

Unity3D import VR devices and using VR

We can download viveport at *www.vive.com/setup* and install VR equipment according to viveport's instructions.
Set up the VR device and run the Unity project to achieve point cloud tracking in the VR headset.