UNIVERSITY OF MASSACHUSETTS AMHERST

ECE-697SD-NetworkedEmbeddedSystDes

REMOTE COLLABORATION USING VR

PROJECT PRESENTATION

Team members：Xiaohao Xia, Yinxuan Wu, Xintao Ding, Xin Zhao

Dec 07, 2021

CONTENTS

01 Project Introduction

02 Design and Contributions

03 Project Demo

01 Project Introduction

1. PROJECT INTRODUCTION

1.1 MOTIVATION

- Point cloud surface tracking
  - Virtual Reality star

- VR remote tracking
  - Facebook Oculus Horizon Workrooms

1. PROJECT INTRODUCTION

1.2 TECHNOLOGY

- **Visual Studio 2019, Python:**
  - Depth camera captures objects to generate point cloud files
  - Multiple point cloud files to achieve stitching

- **Powershell**
  - Use python to set up a FTP file server in the AWS cloud.
  - Use shell scripts to set up two clients for uploads and dowloads

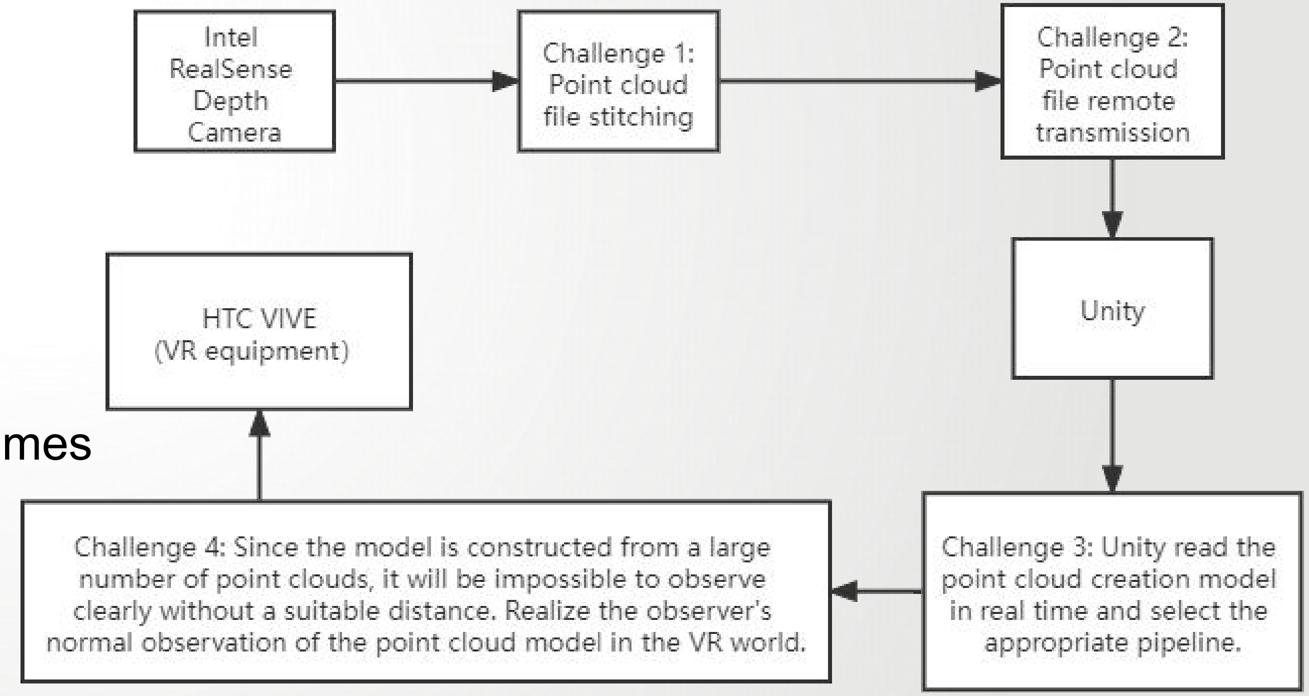
- **Unity, C#**
  - Unity reads the point cloud file to generate a model
  - Set the VR viewing angle and movement function, you can use the VR device to realize the movement of the observer and observe the point cloud model.
  - Import project files into HTC VR headset

- **HTC Viveport**
  - Configure the computer environment to enable the VR device to read Unity project files

1. PROJECT INTRODUCTION

1.3 THE MAIN CHALLENGES OF THE PROJECT

- **Challenge 1:**
  - Point cloud stitching
- **Challenge 2:**
  - Point cloud remote transmission
- **Challenge 3:**
  - Unity reads the point cloud in real time
- **Challenge 4:**
  - Set the appropriate number of Unity reading frames and VR viewing angle



Intel RealSense Depth Camera

Challenge 1: Point cloud file stitching

Challenge 2: Point cloud file remote transmission

Unity

Challenge 3: Unity read the point cloud creation model in real time and select the appropriate pipeline.

Challenge 4: Since the model is constructed from a large number of point clouds, it will be impossible to observe clearly without a suitable distance. Realize the observer's normal observation of the point cloud model in the VR world.

HTC VIVE (VR equipment)

Figure 1 – Main challenges of the project

1. PROJECT INTRODUCTION

1.4 PROJECT COMPLETION

Can realize point cloud splicing, splicing into a complete model

+

Can realize remote transmission of point cloud

+

Can realiza that real-time reads point cloud to create the model by Unity

=

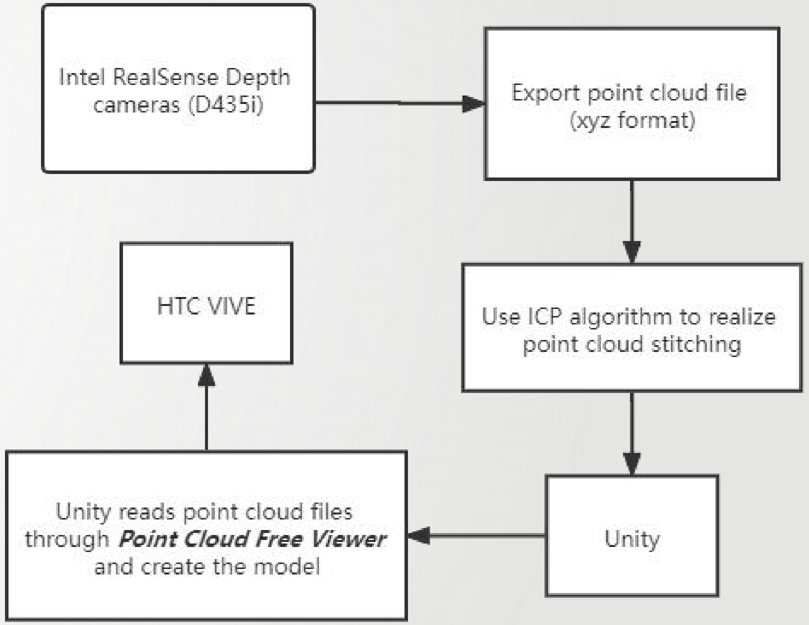
Can realize that real-time remoting tracking of point cloud object by VR

02 Design and Contributions

2. DESIGN AND CONTRIBUTIONS

2.1 DEVELOPMENT LOGIC

- **Part 1: Point cloud - *Challenge 1*:**
  - Export point cloud
    - Python open3d, *pyrealsense2.points.export_to_ply("<name>.ply",* <frame_name>)
  - Point cloud stitching
    - ICP algorithm
- **Part 2: Remote transmission - *Challenge 2*:**
  - Point cloud remote transmission
- **Part 3: Modeling - *Challenge 3, Challenge 4*:**
  - Unity reads the point cloud in real time
    - Unity tool: *Point Cloud Free Viewer*
  - Set the appropriate number of Unity reading frames and VR viewing angle
    - Development C# script
- **Part4: Project optimization**
  - Solve point cloud crossing, parameter adjustment, etc.



Intel RealSense Depth cameras (D435i)
Export point cloud file (xyz format)
Use ICP algorithm to realize point cloud stitching
Unity
Unity reads point cloud files through *Point Cloud Free Viewer* and create the model
HTC VIVE

Figure 2 – Development logic

2. DESIGN AND CONTRIBUTIONS

2.2 DEVELOP CONTRIBUTIONS

- **Part 1: Point cloud - *Challenge 1*:**
  - Use Intel Depth cameras to capture the target and export the point cloud file through Open3d **(Yinxuan Wu)**
  - Use ICP algorithm to realize point cloud stitching **(Yinxuan Wu)**

- **Part 2: Remote transmission - *Challenge 2*:**
  - Design a script, use Powershell to start the server to realize remote transmission of point cloud files **(Xin Zhao)**

- **Part 3: Modeling - *Challenge 3, Challenge 4*:**
  - Develop Unity project, use *Point Cloud Free Viewer* to read point cloud files to create models **(Xiaohao Xia)**
  - Develop Unity C# script, Unity can customize the path and customize the frame number reread function to refresh the point cloud model, and realize remote real-time reading **(Xiaohao Xia)**

- **Part4: Project optimization**
  - Record test results, optimize the project, deal with point cloud out-of-bounds issues, and look for point cloud compression algorithms **(Xintao Ding)**
  - Experiment report writing **(All members)**

University of Massachusetts Amherst

03 Project Demo

3. PROJECT DEMO

3.1 POINT CLOUD STITCHING

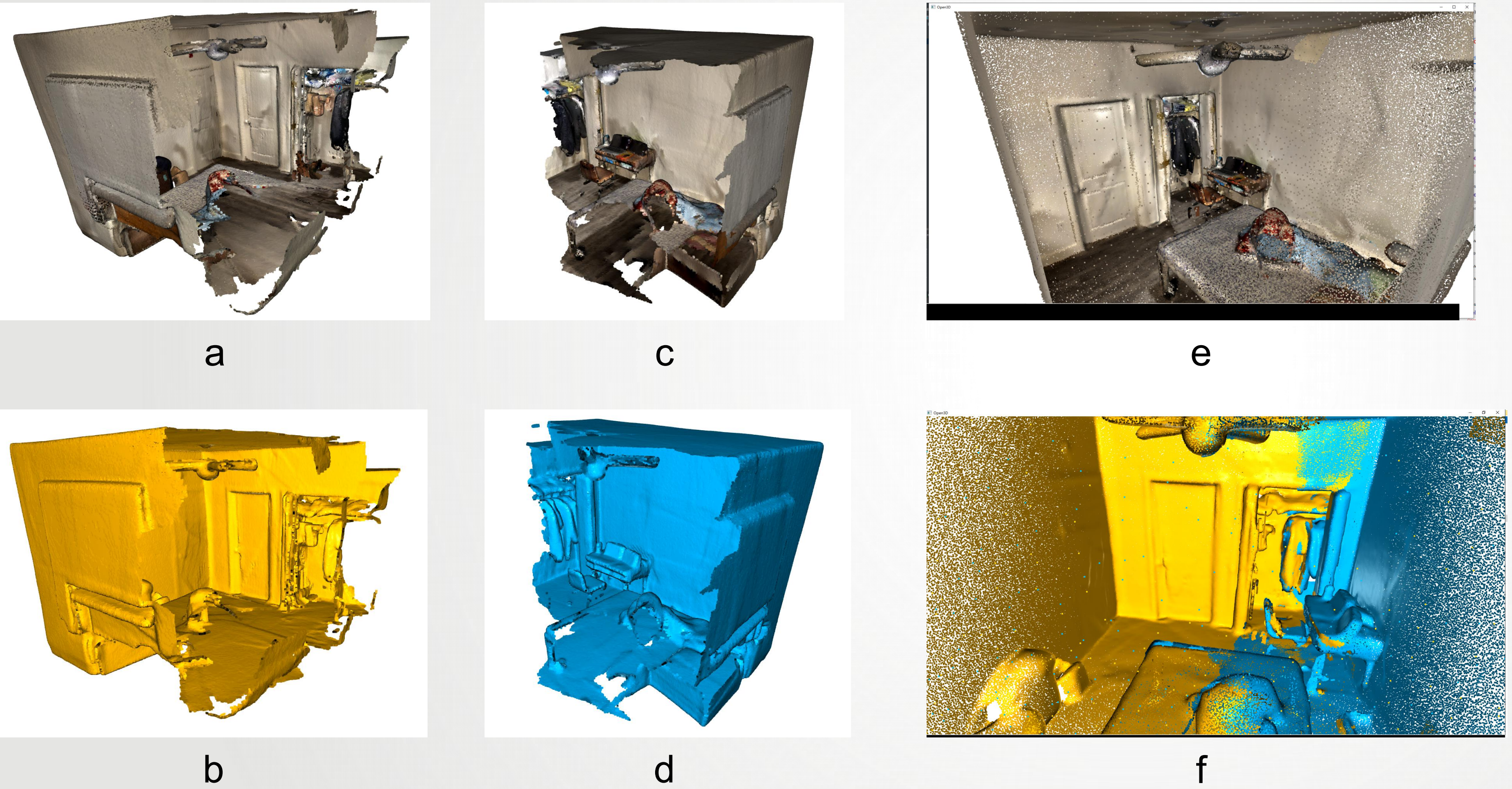


Figure 3 – Results of point cloud stitching (bedroom)

- Due to the objective limitations of the depth camera equipment's shooting capabilities, in order to demonstrate the effect better, we will separately display the realization of point cloud stitching technology at first.

- We use the high-precision point cloud captured by the *iPhone 12 pro* to verify the feasibility of our point cloud stitching algorithm.

3. PROJECT DEMO

3.1 POINT CLOUD STITCHING

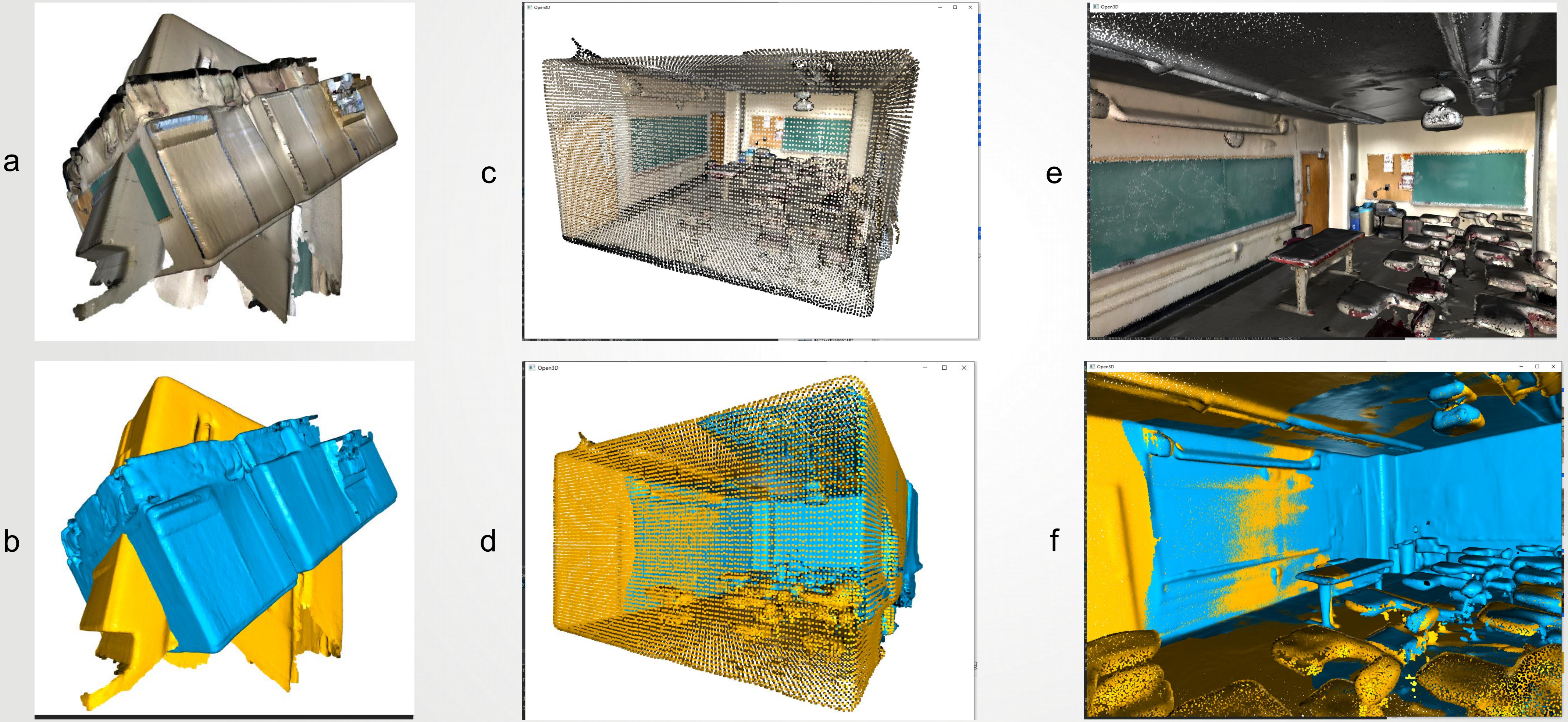


Figure 4 – Results of point cloud stitching (classroom)

3. PROJECT DEMO

3.2 REAL-TIME TRACKING POINT CLOUD

- In order to better demonstrate our real-time tracking of objects, we first use the local computer to display point cloud tracking, temporarily ignoring the impact of delay caused by network transmission. And we will demo the real-time remote tracking point cloud in the next section 3.3

3. PROJECT DEMO

3.3 REAL-TIME REMOTE TRACKING POINT CLOUD

- In addition, you can visit our Github for more details.

https://github.com/xinnnnzhao/ECE697SD

END

Thank you