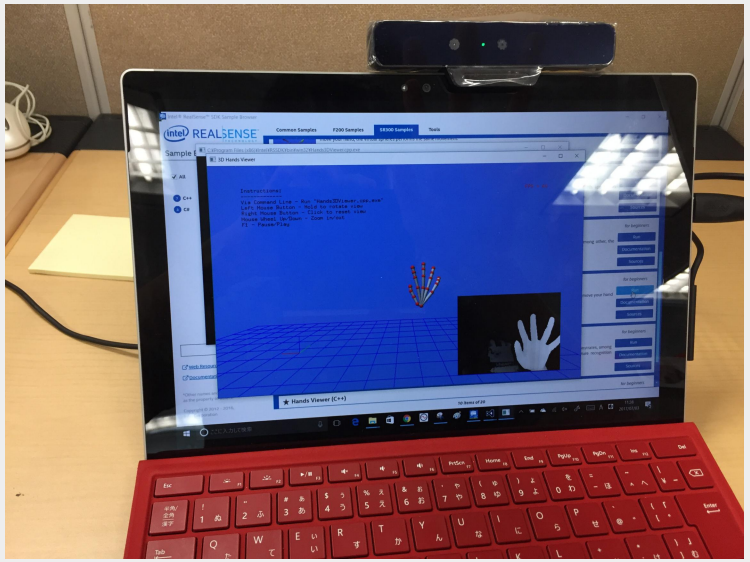
Introduction



- With the development of intelligent interaction systems, related equipment and computer graphics, **somatosensory interaction** has entered the field of vision of most people as a brand-new, behavior-rich interaction method.

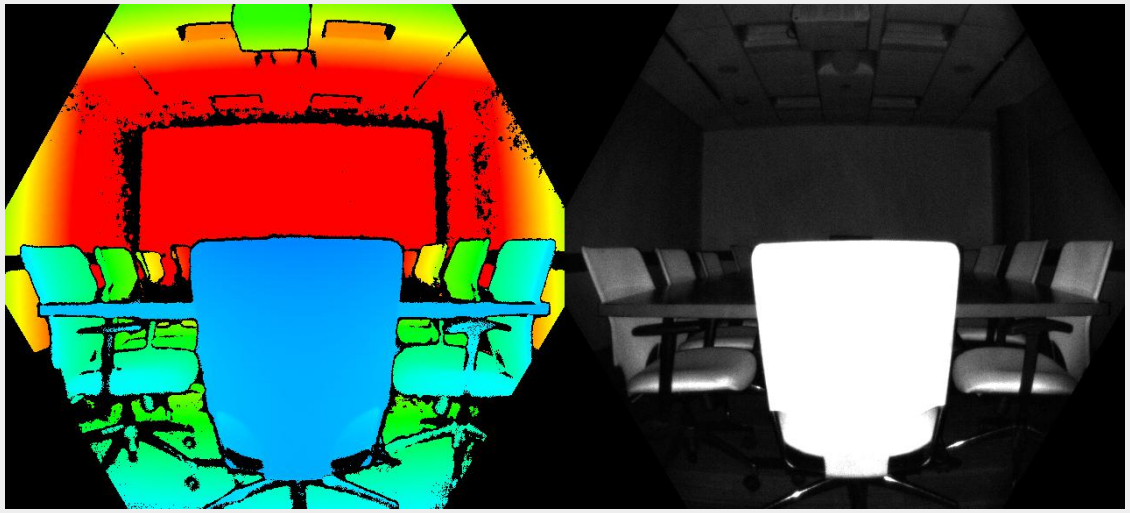
- Compared with the traditional **human interface**(WIMP), somatosensory interaction weakens people's reliance on the mouse and keyboard, allowing users to focus more on the semantics and content expressed by their actions or gestures. Therefore, somatosensory interaction is more intelligent and more ergonomic.

Related Work

- At present, **depth cameras** are mainly divided into three types: structured light depth camera, binocular vision depth camera and TOF depth camera.

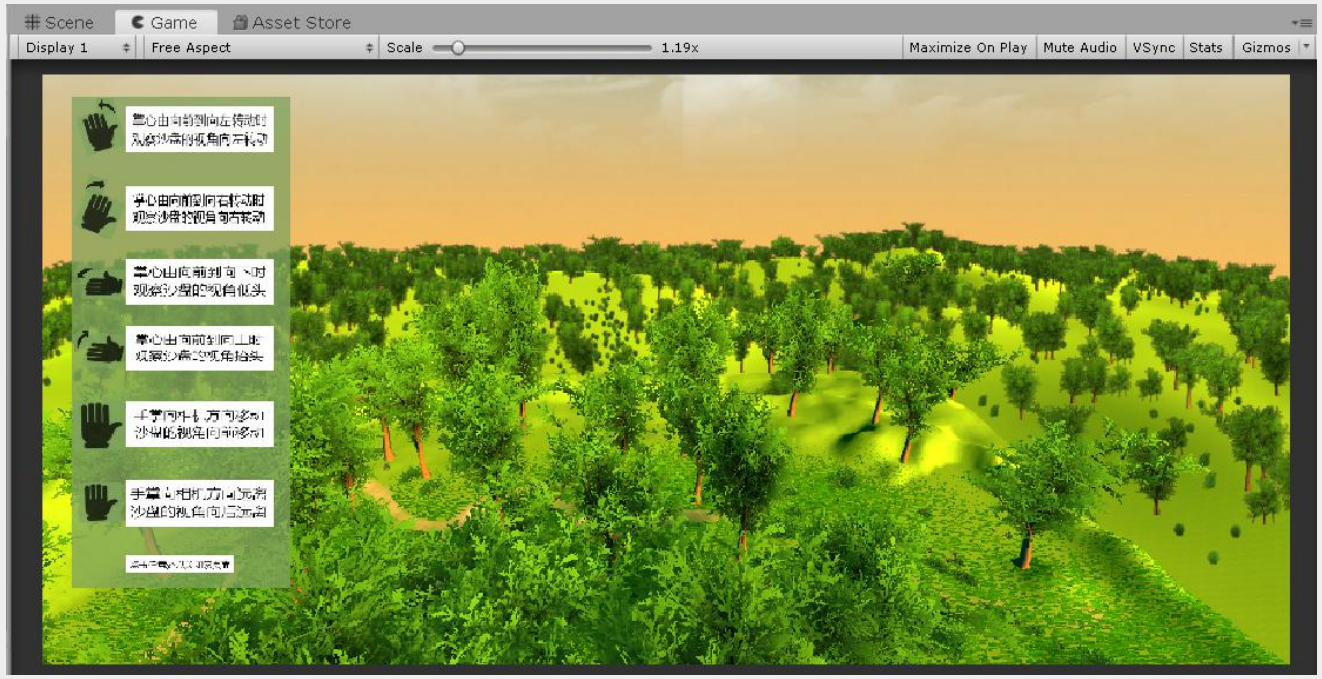
- (1) Structured-light, representing companies--Obi Zhongguang, Apple (Prime Sense), Microsoft Kinect-1, Intel RealSense, Mantis Vision, etc.
- (2) Stereo, representing companies--Leap Motion, ZED, DJI.
- (3) Time-of-flight method (TOF), representing companies--Microsoft Kinect-2, PMD, SoftKinect, Lenovo Phab.

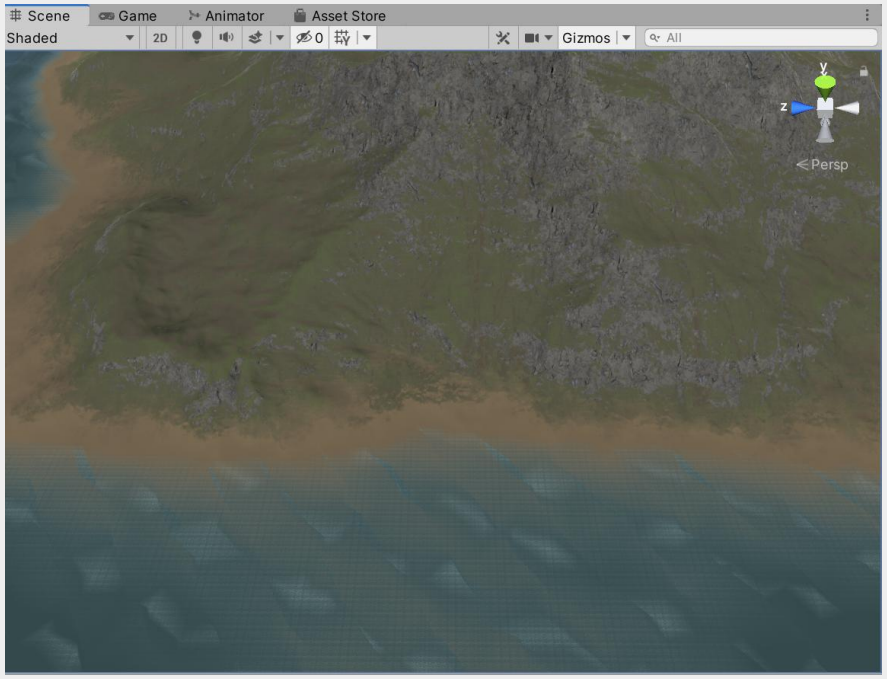
Related Work



- However, somatosensory interaction based on depth camera is a technology that has emerged recently, thus the algorithm optimization of **gesture recognition** based on depth camera are few, not to mention the applications that combine gesture control and **virtual sandbox** display.

Proposal

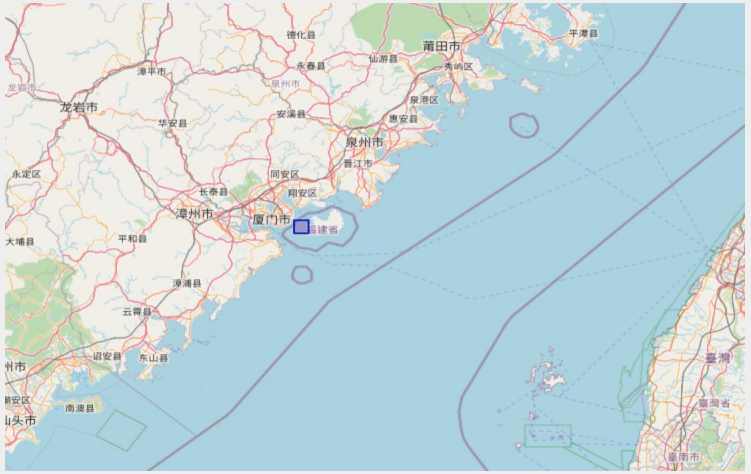


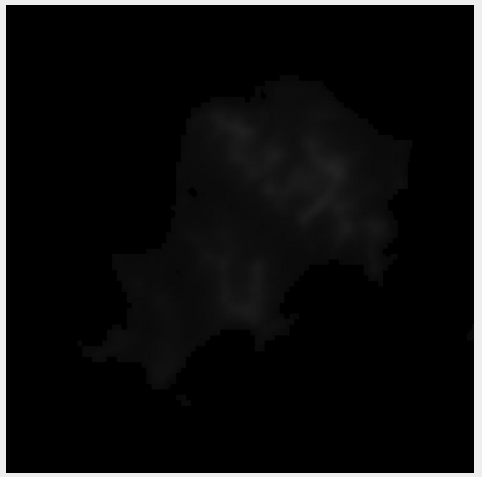


- In order to solve the problems mentioned above, we decided to use **Intel Realsense SR300** and its API to define common human-computer interaction actions, use **Unity 3D** to build a forest sand table model and an island sand table based on the real world in the scene and use ray tracing to render the scene lighting.
- As for the optimization of the algorithm, due to the fineness of hand movements, we considered adding relevant factors when updating the position of the main camera in the scene to eliminate jitter of camera.

Implementation

Terrain Generation





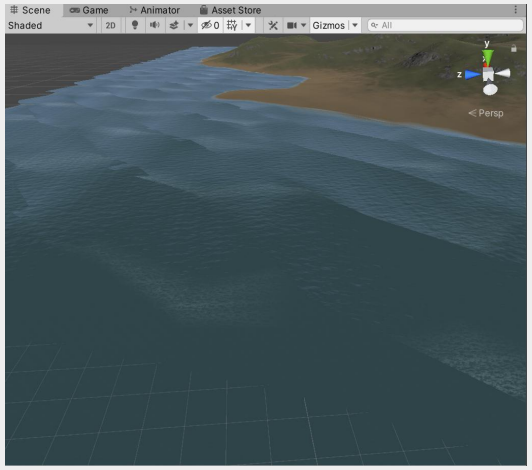
Island height map in the Taiwan Strait area (Turn up the brightness to see the island texture)

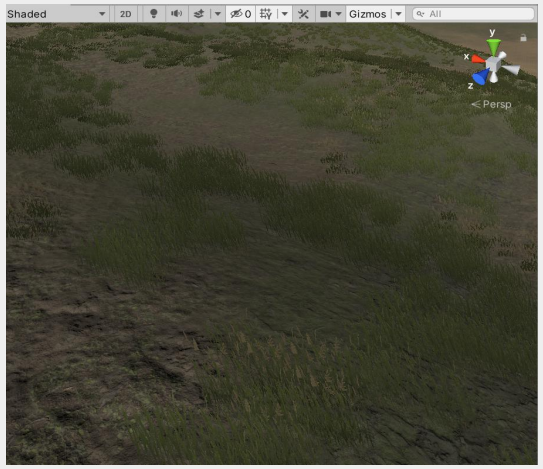
- Import the obtained grayscale image in PNG format into PhotoShop, and then modify the size of the canvas to 1024px*1024px, because the size of the height map in Unity's terrain tool must be 2 to the power of N. Unity can use the raw format height map to generate the 3D terrain model.

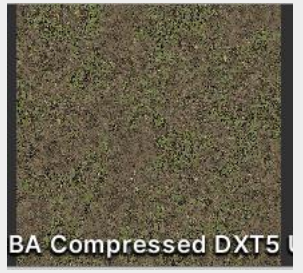
- Finally, create a new Terrain GameObject in Unity, and then find Import Raw in the Settings interface of Terrain, and import the raw file generated before, then Unity can automatically render the corresponding 3D geographic model corresponding to the height map.

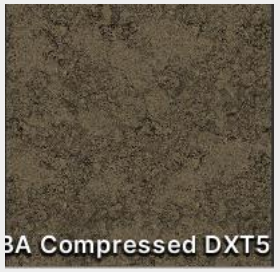
Implementation

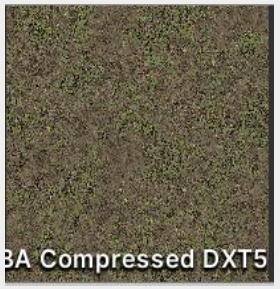
Terrain Rendering











- Perform texture mapping on the preliminarily rendered 3D model in blocks and regions.

- Ocean effect: Use the Water4Advanced component, set the height to 50, which can cover the wrinkles and burrs on the shore that are easy to generate when the height map is used.

Implementation

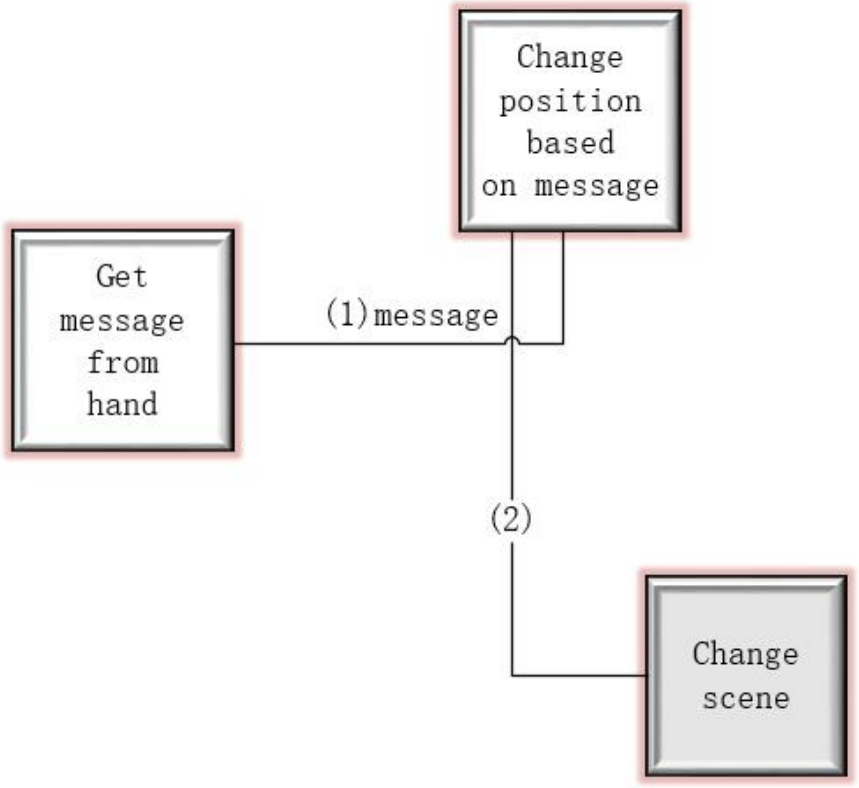
Interactive war chess

- In order to make our sand table model more realistic, we have added interactive war chess, such as tanks, armored vehicles etc, and can control its movement and direction. It is achieved by using a Game Manage script.

- In the subsequent optimization, it was realized that on the basis of automatic path finding, all war chess units can have different pitch effects according to different slopes. This is mainly obtained by the vector product of the tangent of the ground and the direction of the car. The actual driving direction of the car in one frame is realized by the tangent direction of the ground. At the same time, a smoothing operation is added, that is, the car is driving in the direction of the tangent line in every frame, instead of completely following the tangent line. This is to avoid the sharp changes in the slope in some areas.

Implementation

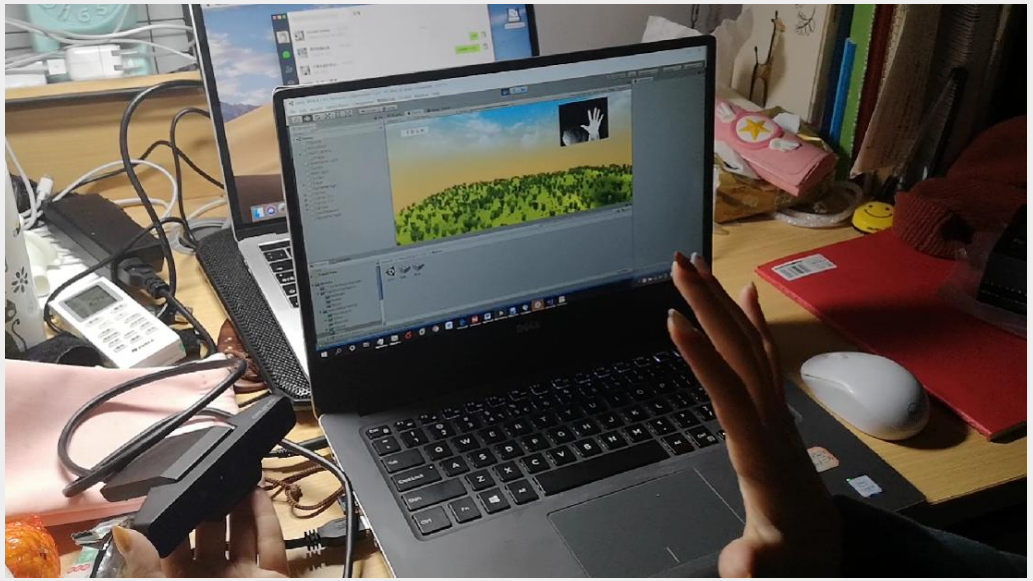
Gesture control



- Using gestures to control the sandbox mainly contains SendMessage and Tracking.
- A trigger is used in **SendMessage** to obtain information. When a hand movement is detected, a trigger is activated and the information is transmitted.
- In **Tracking**, after receiving the trigger information, it will update the camera position in real time in each round of calls according to the acquired hand position change, so as to realize the control of the camera.

Implementation

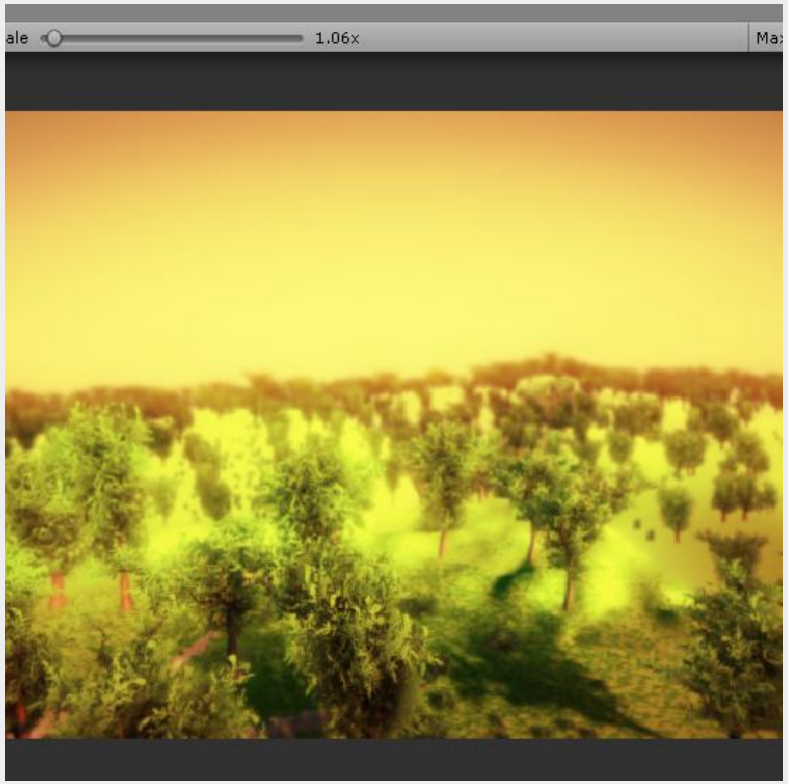
Gesture control

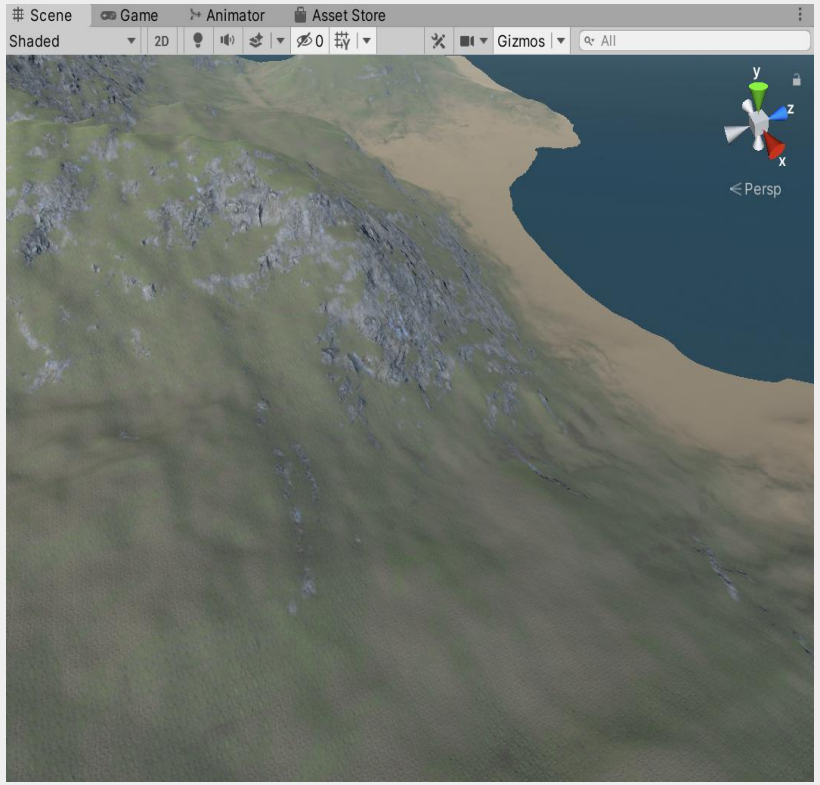


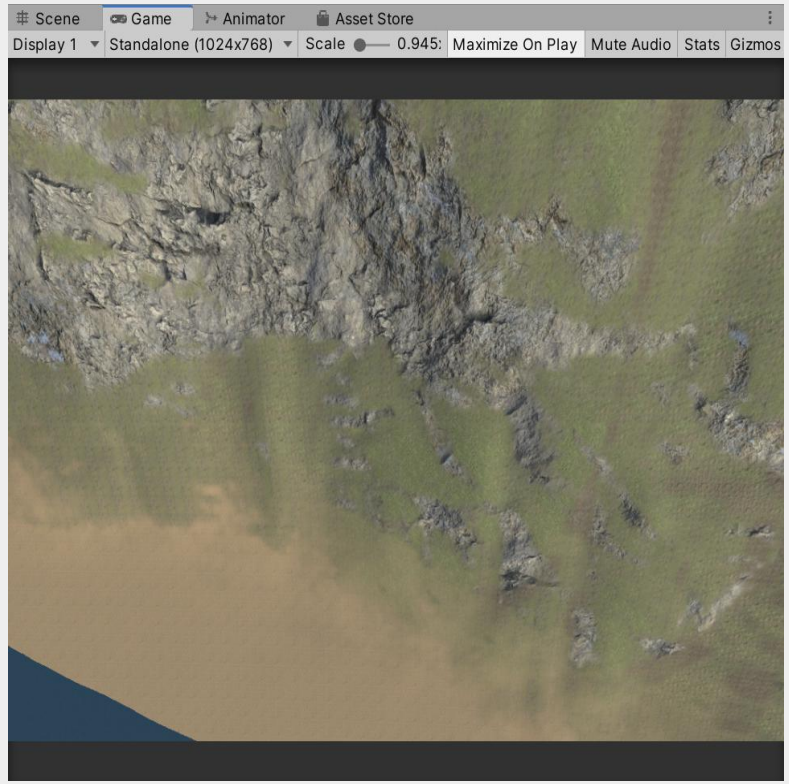
- **Problem:** Because the hand movements are delicate and complicated, the camera will have some unneeded **jitter** when it moves with the gestures, such as shaking up and down and shaking left and right.

- **Solution:** On the one hand, it is necessary to add a **gentle factor** when updating the camera movement. Each position change should have a certain correlation with the previous position, rather than completely determined by this hand movement. This method can smooth the problem of camera shaking in the scene caused by the shaking of the hand to a certain extent.
- On the other hand, each time a hand position change is detected, only the change value of the change in the direction with the largest amplitude is returned, while other subtle changes are ignored.

Results







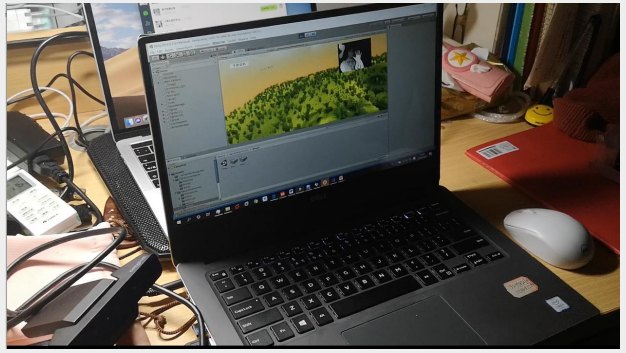
- forest sandbox
- island sandbox
- Finally, we designed two kinds of virtual sandbox, the first is **stylized rendering**, and the second is **photorealistic rendering** based on real world terrain.

Results

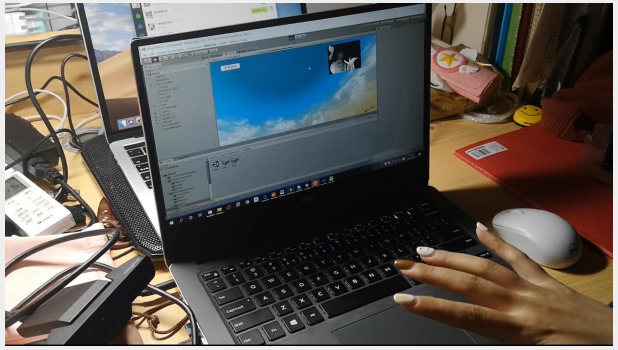
Operations	View Down	View Up	Turn Left	Turn Right	Forward	Back ward
Gestures	Wave down	Wave up	Turn wrist to the left	Turn wrist to the right	Push the palm forward	Push the palm back

- And we designed six different gestures to define camera movement, through which we can realize the control of the virtual sandbox based on somatosensory interaction.

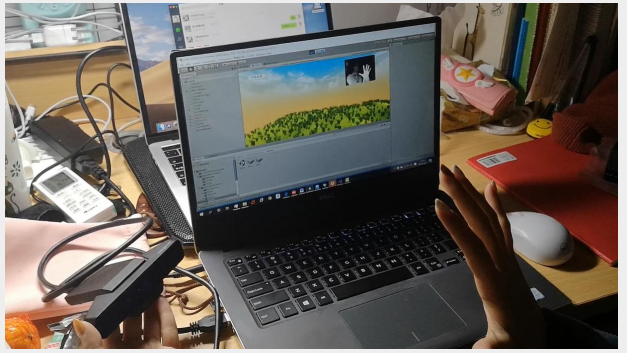
Results



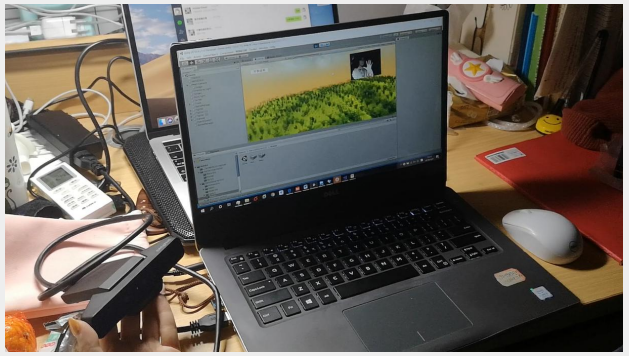
- Down



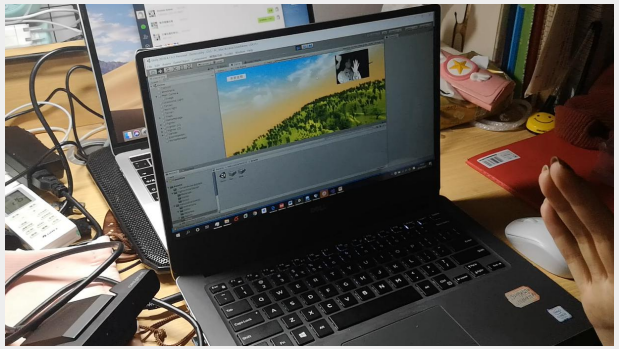
- Up



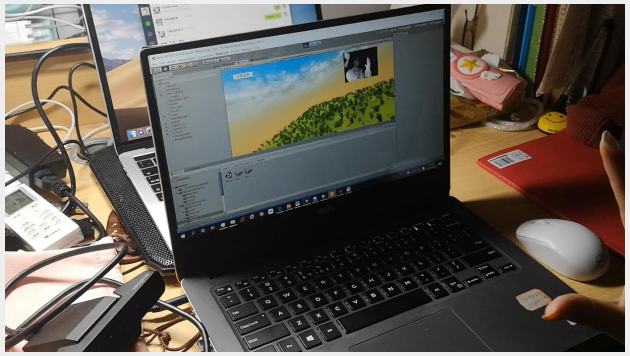
- Forward



- Backward

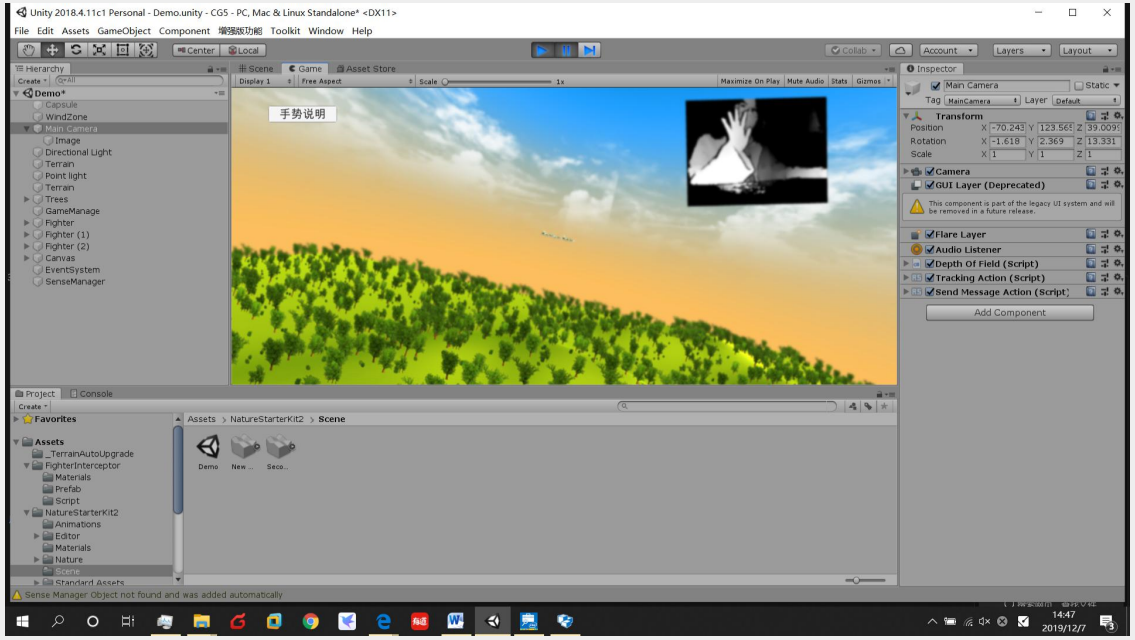


- Left



- Right

Results



- **In summary**, we use the depth camera to extract hand information, connect the hand motion with main camera motion, and control the sandbox in the way of somatosensory interaction.

- However, there are some limitations in our system. **First**, the control of wargames is not achieved through complete somatosensory interaction. **Second**, there still will be some jitter in the hand movement, which will cause the camera to shake to a certain extent, which makes the whole picture not so stable.