**Can we detect VR / AR handedness?**

**Zemin XU**

**Assignment:**

To implement a “black box” detecting user handedness by observing some Interactions.

**How to run the experience:**

You can use WASD or arrow keys to navigate in scene, up and down direction keys are going forward and backward, with left and right keys for rotating. We will use mouse as controller in this experience and if we press ‘ctrl’, we can put the controller forward.

**File organization:**

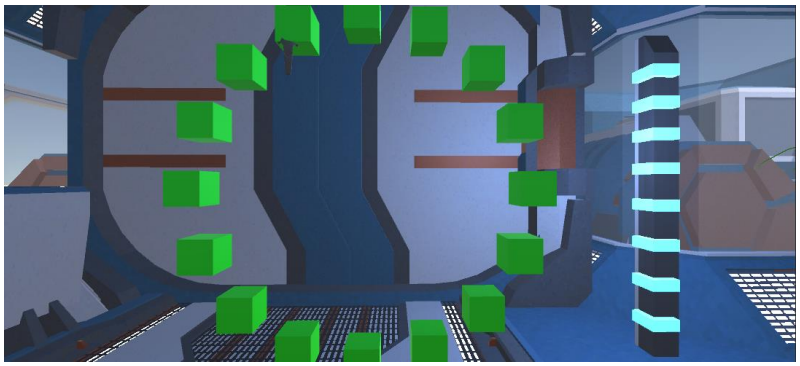
The machine learning script is inside the “Machine Learning” Folder, and the running files for Unity is inside the “Assets” Folder. The UserData is in ‘Assets/Data/UserData.csv’ file.

**Interaction design:**

In this experiment, I will use pointer like a VR controller to manipulate like in a VR environment. To know the handedness there should be a single controller with one button onto it. I cannot use left button or right button as a feature. My basic idea is to use the offset of translation of controller and rotation to know the user’s handedness.

The experiment is inspired by the paper[1]. The first mission is open a door. Normally a right-handedness will open right door directly. Therefore, the translational offset of controller of x is useful.

The second mission is to draw a circle with controller. In Unity, I make a circle combination of cubes for testing handedness. For a right-handedness, his score of activating cubes when drawing circle will be better than that of using left-hand. What’s more, the right-clock direction and left-clock direction will be useful and I use offset of Euler rotation to denote it.



Therefore, the collected features and label are the following, with the last item as label that we need to set at the beginning in GameManager script to train data. Each time when user open the door of Mission 1 and touch the cube while doing Mission 2, the datafile will be updated.

```
private static string[] headers = new string[FEATURE_NUM + 1]
{
    "controller_movement_offset_x",
    "controller_movement_offset_y",
    "controller_movement_rotation_offset_x",
    "controller_movement_rotation_offset_y",
    "touched_cube_num",
    "is_right_handedness"
};
```


The assets of modeling and UI are mostly coming from AssetStore[2] with the model of controller from sketchfab[3].

The next experiment I would like to add is force-control test, which is explained in the paper[1] as well. The finishing time will be interesting for predicting handedness for this task.

**Machine Learning:**

This is a binary classification problem (left-handedness or right-handedness) with several features. I use SVM(support vector machine) method to do training because this is also a supervised learning. I try using GridSearch to find the optimal parameters(linear or non linear), and the theoretical accuracy is about 80%.

```
C:\Users\Public\anaconda3\envs\sklearn\python.exe "C:/Users/Zemin XU/Documents/Handedness/MachineLearning/svm.py"
Best parameters:  {'degree': 5, 'gamma': 'scale', 'kernel': 'poly'}
Theoretical score:  0.8166666666666668
Actual score:  0.7800829875518672
```

For machine learning part I mainly use Scikit-Learn package and numpy for SVM algorithm and pre-processing task.

I would like to try KNN algorithm which is also suitable for this training, which will use distance as metric for prediction.

References :

[1] ResearchGate

[2] Sci-Fi Styled Modular Pack | 3D Sci-Fi | Unity Asset Store

[3] Newsfeed - Sketchfab