TIM2022 – Section: Augmented and Virtual Reality - TD02

TD02 – Interacting with objects in AR

I. Finish parts III and IV in TD01

If you do not have a compatible phone and computer, you can create a simulated test scene following the instructions of “Installation ARFoundationSimulation.pdf” on the Moodle.

The two scripts mentioned in the PDF are in the AR-simulator-scripts.zip on the Moodle

II. Occlusion in augmented reality

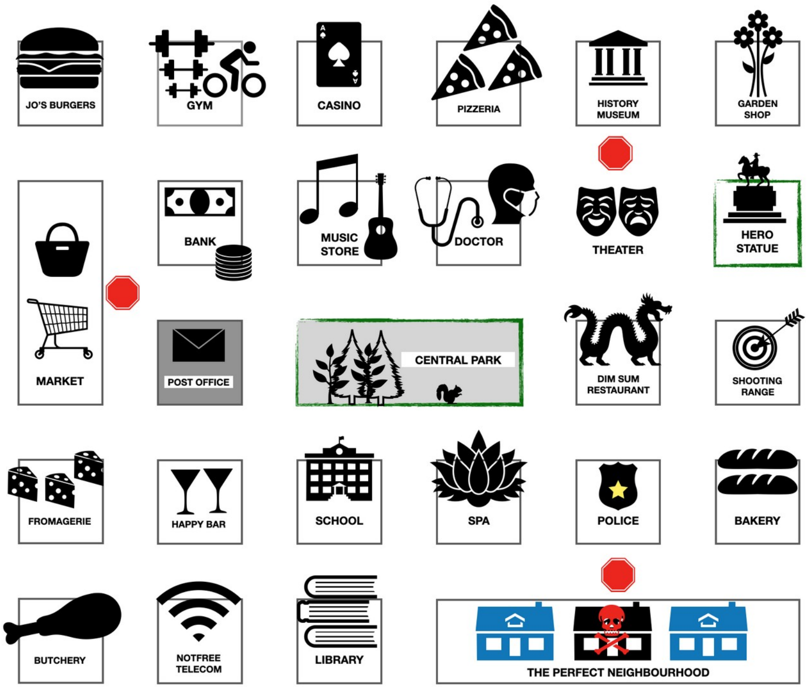
In augmented reality, we often see that the virtual objects “float” above and occlude the real world. This is because there is no model of the real world that tells us what is in front of or behind virtual object.

AR foundation provides functionalities to use the camera to estimate the depth objects, as shown in part IV of TD1 where we insert a robot on a plane. Try to activate the functionality, and test by blocking out part of the virtual robot with your hand.

III. Adding multiple image targets

Imagine you are designing an application to help the police investigate a crime using augmented reality. You want to be able to show 3D models of locations on a map. When you click on the location, you want to see a list of suspicious objects and people at a certain time.

- On the moodle you will find a selection of 3D models of buildings, objects, and furniture. You will also find a 2D map image that you can use as image targets.
  (*hint: the individual location icons may be too small to register as AR targets. You may need to use about a quarter of the map as an image target*)



- Compile and test the app on your device (or use the AR simulator)
- Try moving around the image targets and placing objects in front of the 3D models to test for occlusion

IV. Signaling and feedback

Using Lean Touch, experiment with different types of gesture events and multimodal feedback.

- Set up a timer code such that when the camera fixates on a target for a long time, the 3D models starts to jump, but stops jumping once the camera looks away (hint: check out methods like `WorldToScreenPoint` and `TileSelector` package on Moodle)
- Add a tap event that detects if the user taps on a 3D model that is currently fixated, and make the object change color when the tap is detected
- Add haptic feedback when the object is tapped (see Unity's `Input` system and `Handheld.Vibrate`)
- Add a sound feedback when the object starts/stops moving

V. Visualization and interacting

- Download from the Moodle the list of suspects `suspects.csv` and the list of objects `objects.csv`. A parser for these CSVs has been provided to you in the `CSVReaderScene.unitypackage`.
- Consider the description of the desired application in Part III.
  - Design an on-screen menu, where the user can specify a start and end time in the day (ex: 13h - 15h).
  - Add on-screen interaction on your device that allows the user to select a location on the map to see a list of people and objects that were present during this time.
  - Compile and test the application on your device

  Hint: to design on-screen menus, investigate the Unity `Canvas` object. An example canvas scene is given to you in `CSVReaderScene.unitypackage`