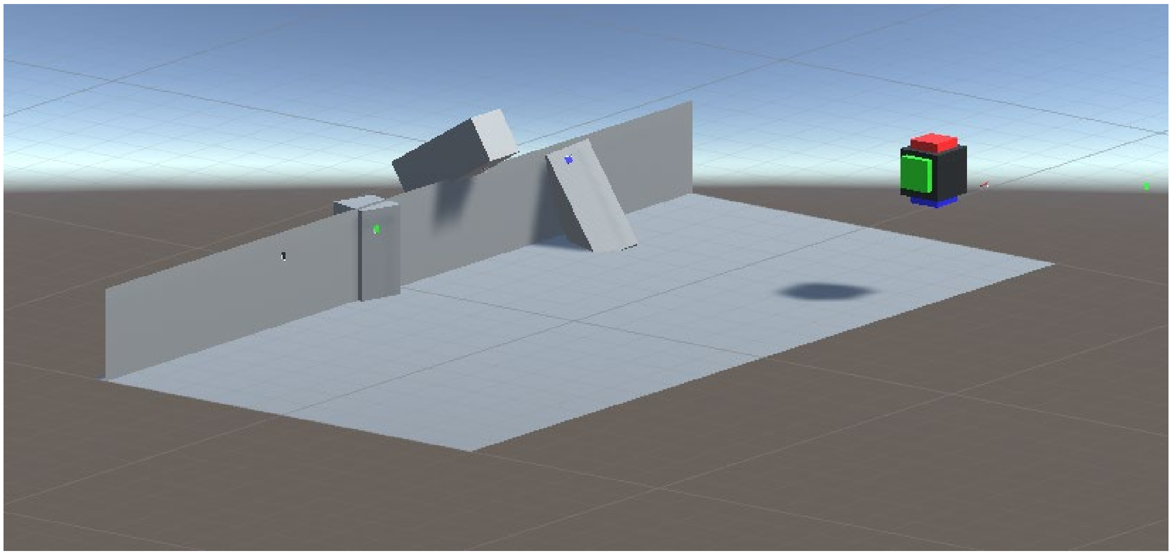
Multiplayer AR Alignment Prototype



Introduction

You have gotten the task of writing a multiplayer enabled Augmented Reality (AR) alignment prototype in Unity. In this project, the players must be able to interact with different virtual QR codes and observe a model being aligned to said QR codes (see the attached video *AlignTest.mp4*). The alignment must be synchronized between the players, and everybody is able to trigger an alignment.

The assignment should be made using Unity version 2019.4.11f1 (LTS).

Task overview

The assignment consists of the following tasks:

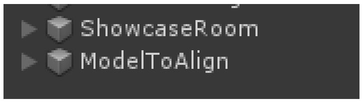
1. Import assets
2. FPS Controller setup
3. Mirror setup
4. Model alignment
5. Metadata

Each task will be explained in detail in the following sections.

Tasks

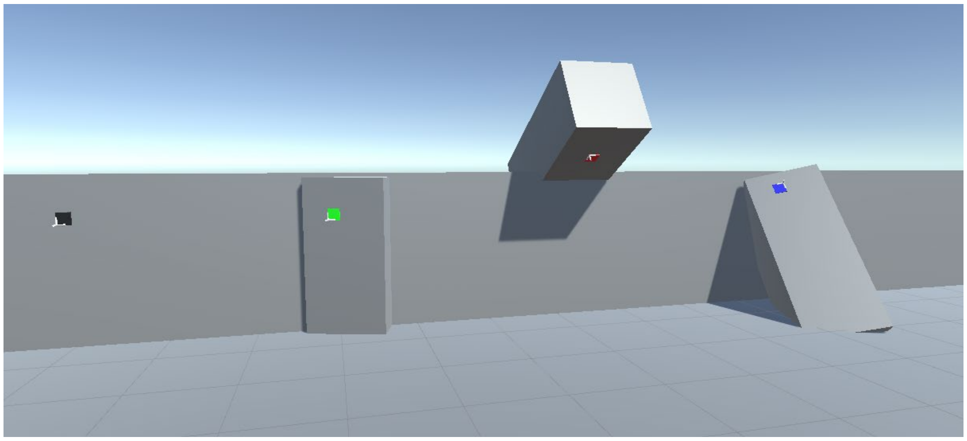
1. Import Assets

You are provided with a unity package *AlignTestNNE.unitypackage*; the package contains the following elements:

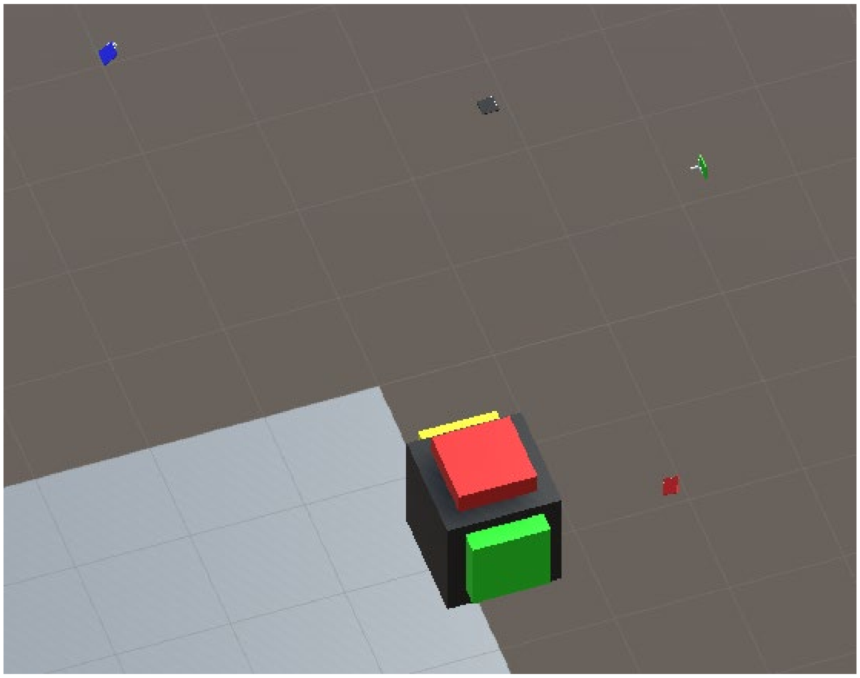


ShowcaseRoom
ModelToAlign

ShowcaseRoom is the model in which the players will be able to walk around. It contains 4 QR codes with a unique color each.



ModelToAlign is the model that needs to be aligned. It also has 4 QR codes with colors corresponding to the ones in the Showcase room.



2. FPS Controller setup

In order to allow the players to move around, you need to use a FPS controller.

You can use the Standard Assets FPS controller or code your own.

3. Mirror setup

'Mirror' is the Networking library that you need to use for the multiplayer implementation.
You can download Mirror through the Asset Store here:
https://assetstore.unity.com/packages/tools/network/mirror-129321

You do not have to setup a dedicated server. We expect the solution to work locally on the same machine by having two or more builds running simultaneously (where one is a host, and another is a client).

4. Model alignment

The players should be able to walk around in the Showcase room model and interact with the 4 different QR codes. Once a QR code is selected, the *ModelToAlign* needs to be moved and rotated in a way that its corresponding QR code (same color) has the same position and rotation as the selected QR code in the *ShowcaseRoom* model.

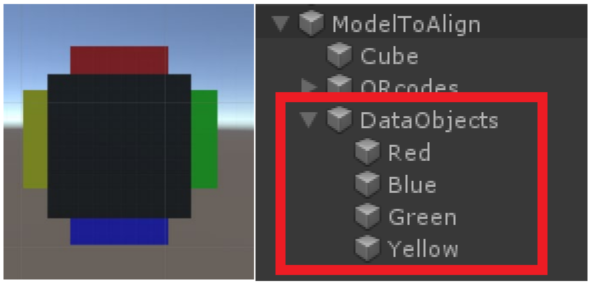
In order to interact with the showcase QR codes you should use Raycasting. It can be triggered by a keypress or a mouse click.

Refer to the provided video, *AlignTest.mp4,* in case you have any questions about how the result should look.

It is important to note, that the alignment should be synchronized between the connected players, such that if e.g. player A triggers the alignment, then player B should be able to see the result as well and vice versa.

5. Metadata

In the provided *ModelToAlign* GameObject you will see four different colored cubes in the *DataObjects* child.



Each cube has its own associated metadata in csv format, located here: https://nnedigitaldesignstorage.blob.core.windows.net/candidatetasks/Metadata.csv?sp=r&st=2021-03-15T09:12:39Z&se=2024-11-05T17:12:39Z&spr=https&sv=2020-02-10&sr=b&sig=oyj3Qyg4W42%2BO0d7YqmjxmKk0k%2BLVmE243ixdLaq3gk%3D

When a player selects one of the data-cubes, a UI panel should appear for visualizing the associated metadata.
E.g. if you select the red cube, you should see its data entry: “ID: 1, Name: Red, Info: A Red Cube [...]”.

Note: The csv file should be fetched during runtime.

Requirements

- You should use Unity version 2019.4.11f1 (LTS)
- If you are unsure or do not understand a part of the assignment, make an assumption on how you understand the task and document it

Deliverables

You need to deliver the following files in a single zip-folder:

- Windows executable build (.exe)
- The Unity Project
- Readme explaining controls, considerations or any other comments that you want to add (e.g. any potential assumptions that you have made)