Egocentric Visitor Localization and Artwork Detection in Cultural Sites Using Synthetic Data

Santi Andrea Orlando[1,2], Antonino Furnari[1] and Giovanni Maria Farinella[1,3]
[1]*Department of Mathematics and Computer Science, University of Catania, Catania, Italy*
[2]*DWORD - Xenia s.r.l., Acicastello, Catania, Italy*
[3]*National Research Council, ICAR-CNR, Palermo, Italy*

SEN Tool V.2 Documentation

The Tool described in this work:
http://iplab.dmi.unict.it/SimulatedEgocentricNavigations/index.html , has been used to generate the dataset described in this link:
http://iplab.dmi.unict.it/SimulatedEgocentricNavigations/dataset.html
and available to download it.

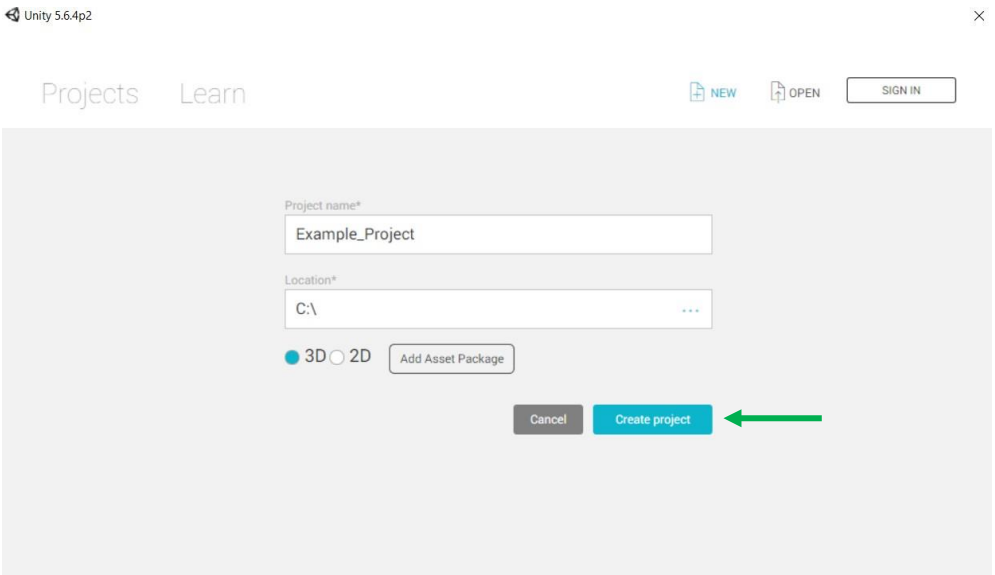
TOOL DESCRIPTION

This tool can be used to generate Dataset of Egocentric Navigation useful to study **Indoor Image Based Localization and artworks detection** in cultural sites.
The tool is made for Unity 3D 2018.2.14f1 or recent version, it has been tested with Windows 10.

1) GettingStarted

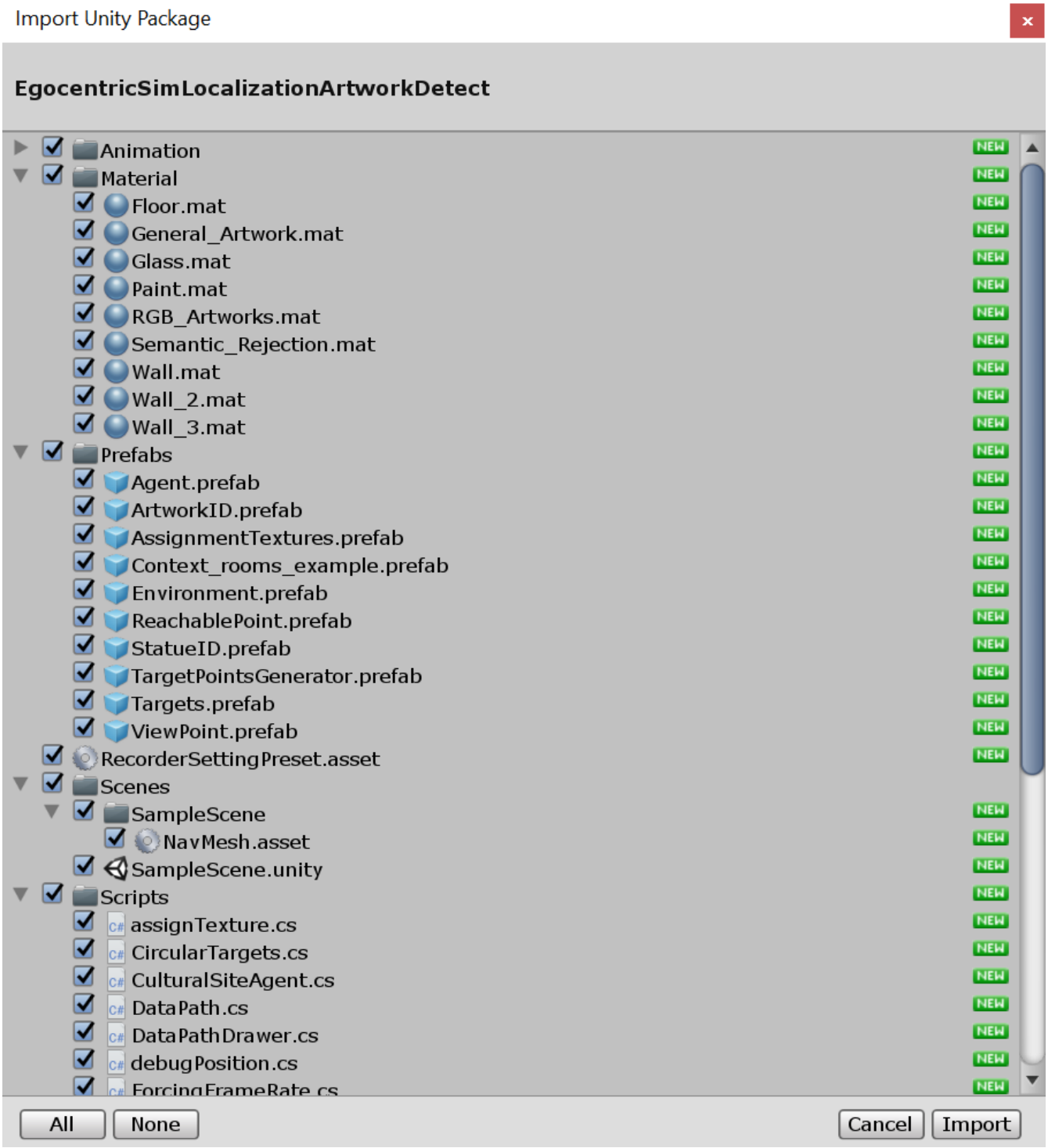
1) Create a New Project

To start to use this tool you need to create a “New Project” from Unity 3D as shown below:



2) Import Package

In the project Choose **Assets > Import Package > Custom Packages...** and select the package: "EgocentricSimLocalizationArtworkDetect.unitypackage". From dialog **box, Import Package** (See Figure below), click on **Import** button.



Wait the importing process…

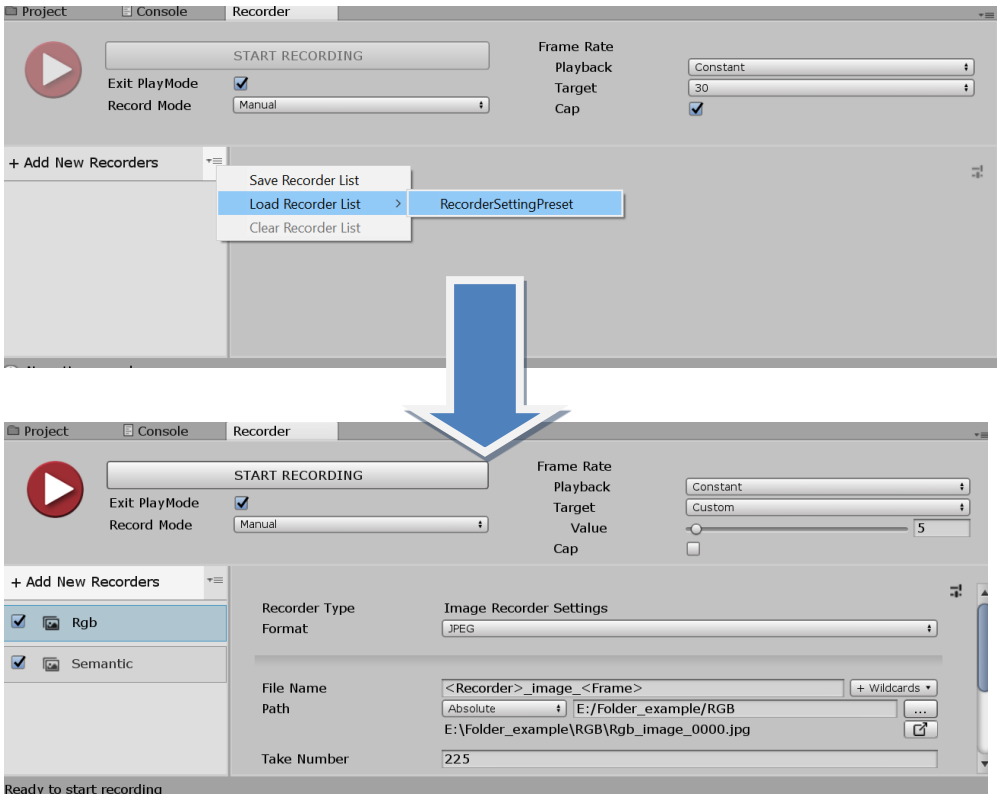
3) Run the Example navigations

From **Project** window open “Scene” folder and open the scene “SampleScene”. By clicking on play button, you can test the Simulation of the Egocentric Navigations.

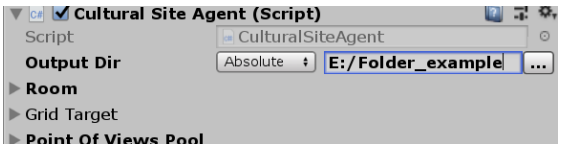
To acquire the frames to generate the dataset open Recorder window from:

- Window>General>Recorder>Recorder Window

From **Recorder** window load the preset Recording Settings:



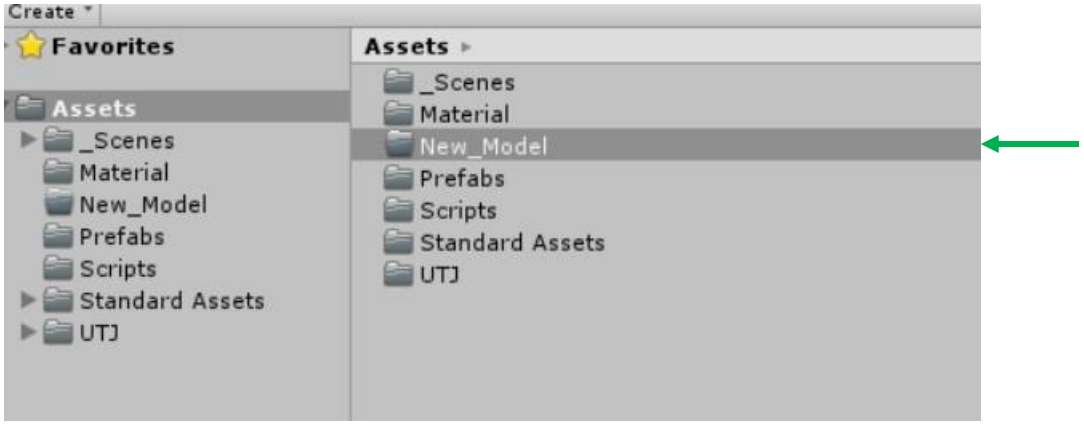
PS: Make sure to create inside the desired output folder, the folder “RGB” and “Semantic” to capture Rgb frames coupled with “Semantic” mask.
You have to change the Path on Recorder window (Both for Rgb and semantic Recorder) and please change also Output Dir on component “Cultural Site Agent” of GameObject Agent.



You can click on **Start Recording** to start to generate the Dataset, the generation will stop automatically when the simulation will be done. To stop it manually you can click the play button on top (but you will lose the label of camera position).

2) How to import a Matterport 3D Model

Create a new folder. On **Project** window right click on **Assets > Create > Folder.**



Follow these steps:

1) Create a folder **Textures** in the new folder;
2) **Drag** in it the textures of the model;



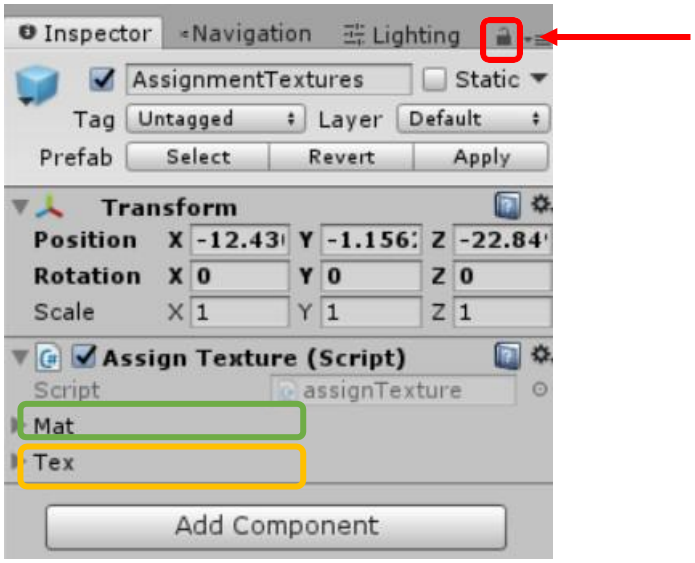
3) **Drag** in the folder of the model the files **.obj** and **.mtl** respectively the 3d model and its materials;
4) You should have the model already textured as below:



5) if you don't have the same results of the figure above follow the next optionally procedure:

Optionally

1) Create a new empty scene **File > New Scene**
2) Drag in it the prefabs **"AssignmentTextures"** (Project window > Assets > Prefebs)
3) Click on the GameObject in **Hierarchy** window and in **Inspector** window lock it by clicking in:



4) Select all the materials of the new model and drag it on **"Mat"** parameter
5) Select all the textures of the new model and drag it on **"Tex"** parameter
6) Click play to attach the textures to the materials

3) Crete a Navigation system by your own

To start to simulate the Navigations you need to follow this steps:

1) Create the Walkable Area
2) Create the reachable target points
3) Setting the Agent

3.1 Create the Walkable Area

1) Create a new empty scene **File > New Scene**

2) Delete the **Main Camera** from the **Hierarchy,** and set the **Direction Light** rotation to (90, 0, 0) with a white colour

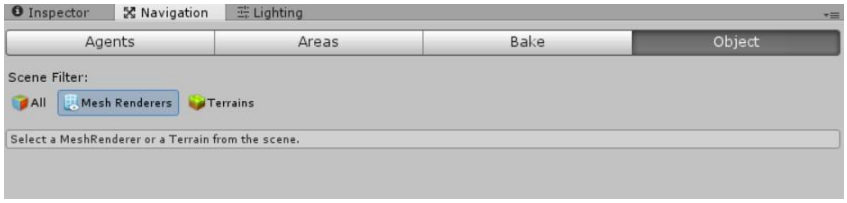
3) Drag the 3D model on the **Hierarchy**

4) Drag the prefab “Agent” on the **Hierarchy**

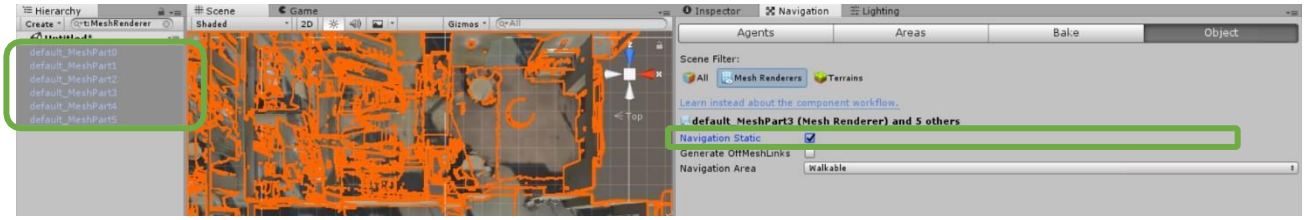
5) Select the model from the **Hierarchy**

6) Go to **Windows > Navigation** to open the navigation window

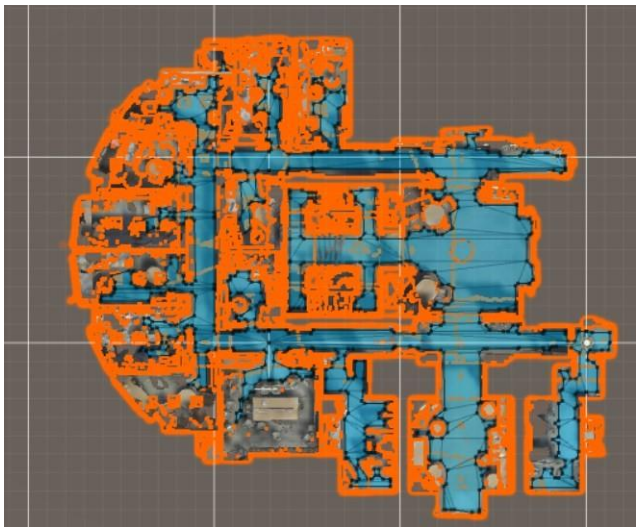
7) On the **Navigations** window select **Object** tab and select Mesh Renders Icon



8) In **Hierarchy** will be shown just the part of the 3D Mesh, Select it all and check **Navigation Static**



9) On the **Navigations** window select **Bake**, set the **Agent Radius** to **0.2** and click on **Bake Button.** You can see in blue the walkable area



3.2 Label of Room Context

After that you created the Walkable Area, you have to label the context for each room of your environment. To do this you have to use the prefab “Context_rooms_example”. This prefab is made of a box collider aims to detect when the virtual agent is inside the room.
The component **Room Information** must contain:

- **Target In Grid:** the reachable target generated inside this room;
- **Pov Target:** the points of view for each artwork present inside this room;

After having labelled each context go to step 3.3.

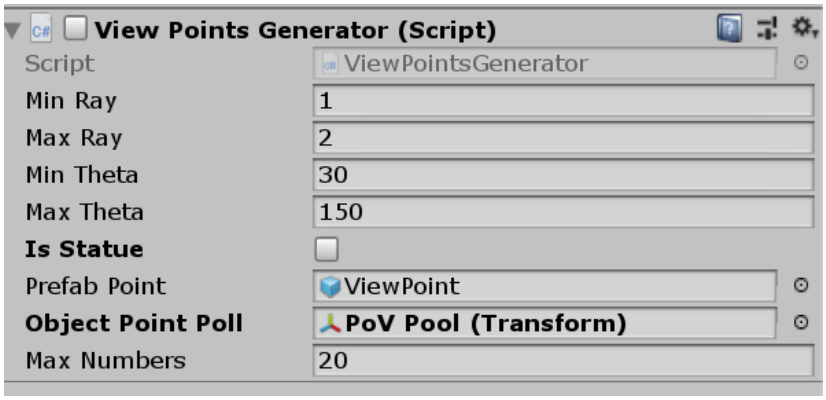
Ps: if you need to use for room with particular shape, more than one “Context_rooms_example” prefabe, make sure to assign it the same name as in “Context_rooms_example_2”.

3.3 Label artworks and generate observation point (Point of view)

To label each artwork you can use the prefab “ArtworkID”. This prefab has to be Renamed for each artwork and positioned in the right position in the 3D mesh.

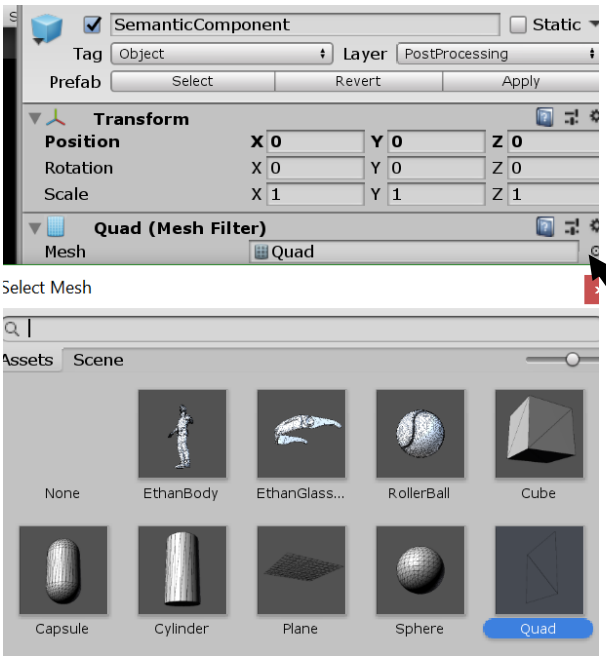
Enable the component **View Points Generator** to generate the observation point for each artwork. You have to set the following parameters:

- **Min Ray** and **Max ray:** delimite the area where the target will be generated;
- **Is statue:** is a Boolean to say that the area of the observation point is 360° around the artwork;
- **Object Point Pool:** is an Empty object that collects all the observation points generated for each artwork;

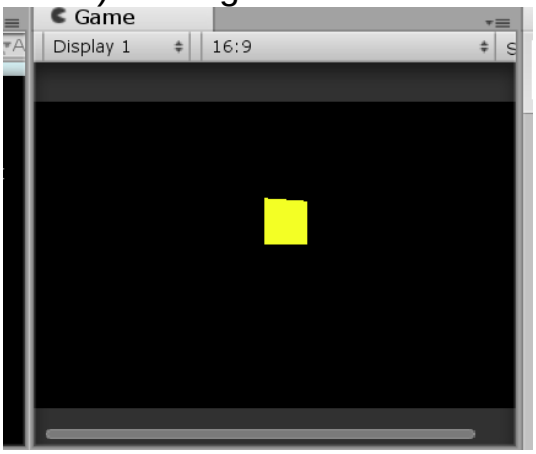


1) Click on play button to generate the points
2) During play mode, inside the created "Object Point Pool" will be instantiated a set of game objects called: "ViewOf_PaintID_000; ….; ViewOf_PaintID_xxx", where PaintID will be the name of your artwork. While you are in play mode select from Hierarchy the **current** game object instantiated inside your "Object Point Pool", and copy it (Ctrl+C)
3) Stop the scene clicking again on play button
4) Paste it into "Object Point Pool" in Hierarchy.
5) Assign with a drag and drop, the generated observation points to the correct "Context_room", where the artwork is located, in the "**Room Information**" component, in "**PoV Target**" parameter.
6) Repeat from 1) for each artwork to label.

PS: You can modify the shape of the "SemanticComponent" (child of "ArtworkID") by changing the (Mesh Filter) component as shown this figure:



Create a material for each artwork using material “Paint” as example (Materials folder). Change the colour to distinguish it from other artworks.

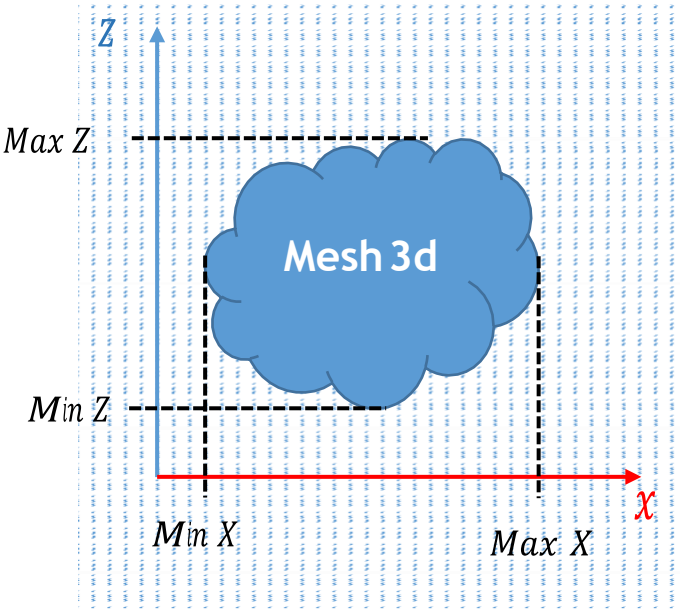


3.4 Create the reachable target points

To create the reachable target points you have to use the prefab "TargetPointsGenerator".

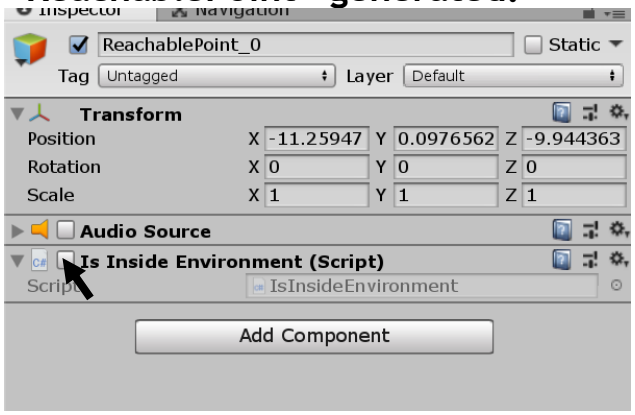
1) Put into the **Scene** the prefab "TargetPointsGenerator" and sets its Position to X **= 0;** Y **= 1;** Z **= 0**
2) In its **Inspector** window go to the component **Target Generator (Script)** and set the parameters:
    a. Shift X (position shift between points along x-axes) in meters
    b. Shift Z (position shift between points along z-axes) in meters
    c. Min X, Max X
    d. Min Z, Max Z

    as shown in figure below.



3) Click on button play
4) During play mode, inside "TargetPointsGenerator" will be instantiated a set of game objects called: "ReachablePoint_0; ….; ReachablePoint _n", while you are in play mode select from Hierarchy the game object TargetPointsGenerator, and copy it (Ctrl+C)
5) Stop the scene clicking again on play button
6) Paste in Hierarchy with Ctrl+V the new one TargetPointsGenerator
7) Now you can remove all the target points that aren't inside the **Walkable Area (if you don't label the context of room as illustrated in section 3.2)**
8) **Put in the correct "Room Information" GameObject the created reachable target by drag and drop it in "Target in Grid" parameter.**

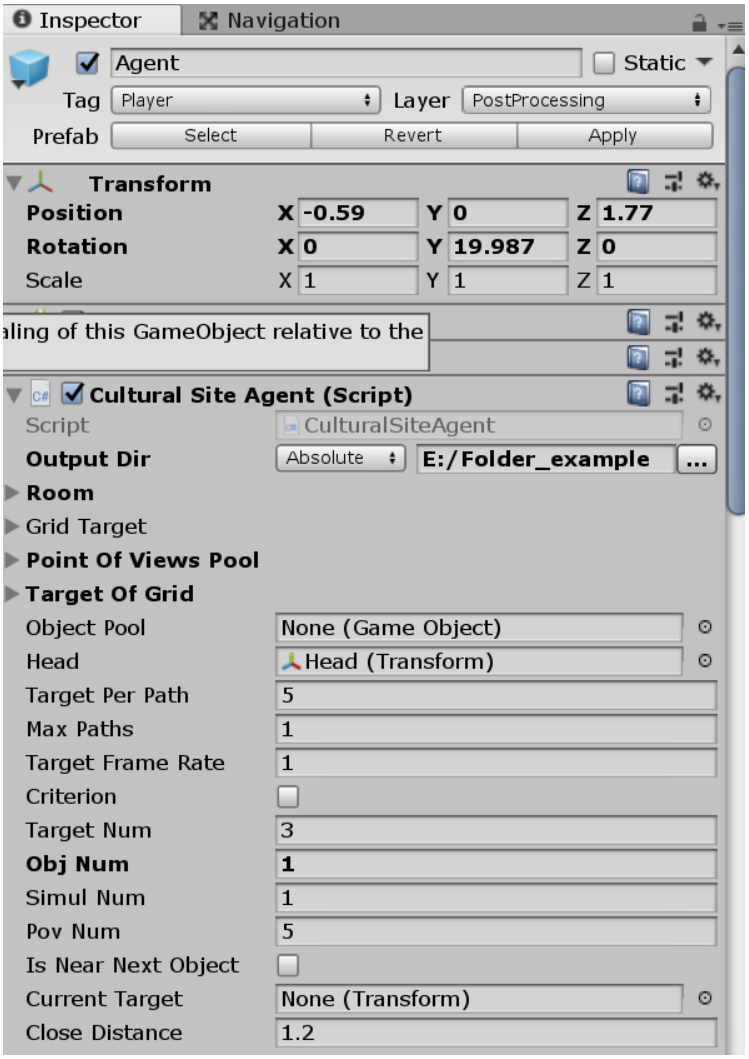
PS: Please make sure to disable Is Inside Environment component of each "ReachablePoint" generated.



3.5 Setting the Agent

After that the target points have been created and deleted the unreachable outside the Walkable Ares, we need to set the Agent that will be perform the Navigations.

1) Put into the **Scene** the prefab "Agent"
2) in **Inspector** window lock it as shown in this figure



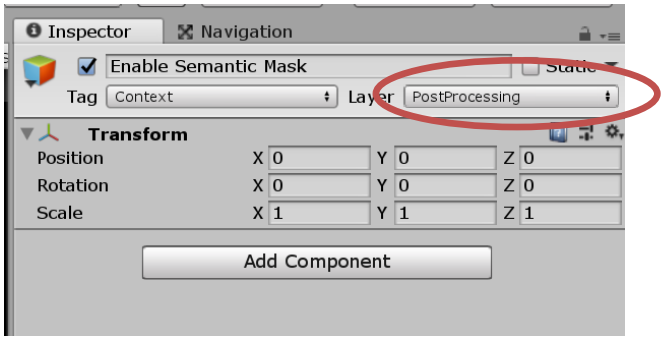
3) Select all the created target points (Target_0, …, Target_n), inside the GameObject "TargetPointsGenerator" and drag and drop it into **Target of Grid** parameter of **"Cultural Site Agent (Script)"** component
4) To start to simulate the navigations you have to set the parameters of **"Simple Agent (Script)"** :
    a. Room
    b. Point of views Pool
    c. Target of Grid
    d. Obj Num
    e. Point of view num

As son of the GameObject Agent you can find the GameObject Head with respective camera RgbCam and SemanticCam. The Height of the Agent has to be set changing the position Y of the GameObject “Head”(Default 1.7m).

Before to start the simulation you have to duplicate your 3d model and change each material of each mesh chunk with the material called:
“Semantic_Rejection” :



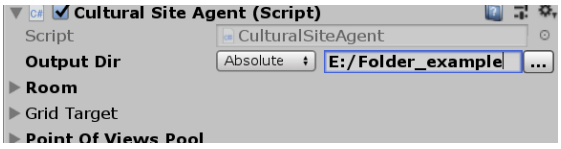
After make sure that the layer of the semantic 3d model and each artwork that you added in the scene have as Layer “PostProcessing”:



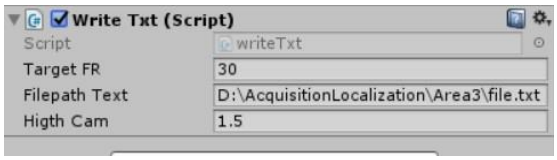
In this way the SemanticCam, already set with PostProcessing layer in Culling mask parameters, can render only the semantic.

To start to simulate the navigations inside your virtual environment you can click “Start Recording” button on Recorder Window, the generation will stop automatically when the Simulation will be done. To stop it manually you can click the play button on top (but you loose the label of camera position).

PS: Make sure to create inside the desired output folder, the folder “RGB” and “Semantic” to capture Rgb frames coupled with “Semantic” mask.
You have to change the Path on Recorder window (Both for Rgb and semantic Recorder) and please change also Output Dir on component “Cultural Site Agent” of GameObject Agent.



Optionally



Enabling the **Write TxT (Script)** and setting an output directory you can save the sequences of the target points reached for each generated Path.