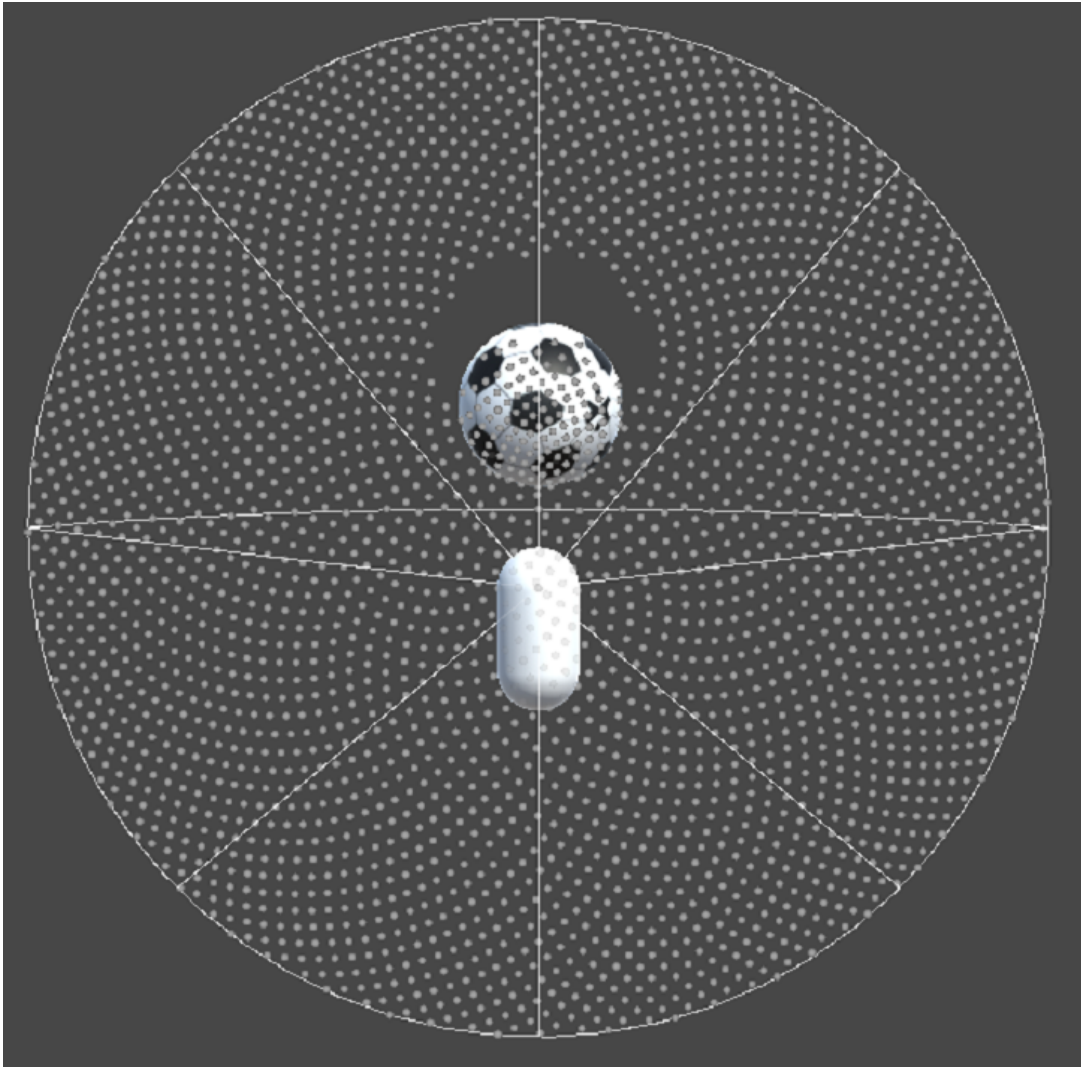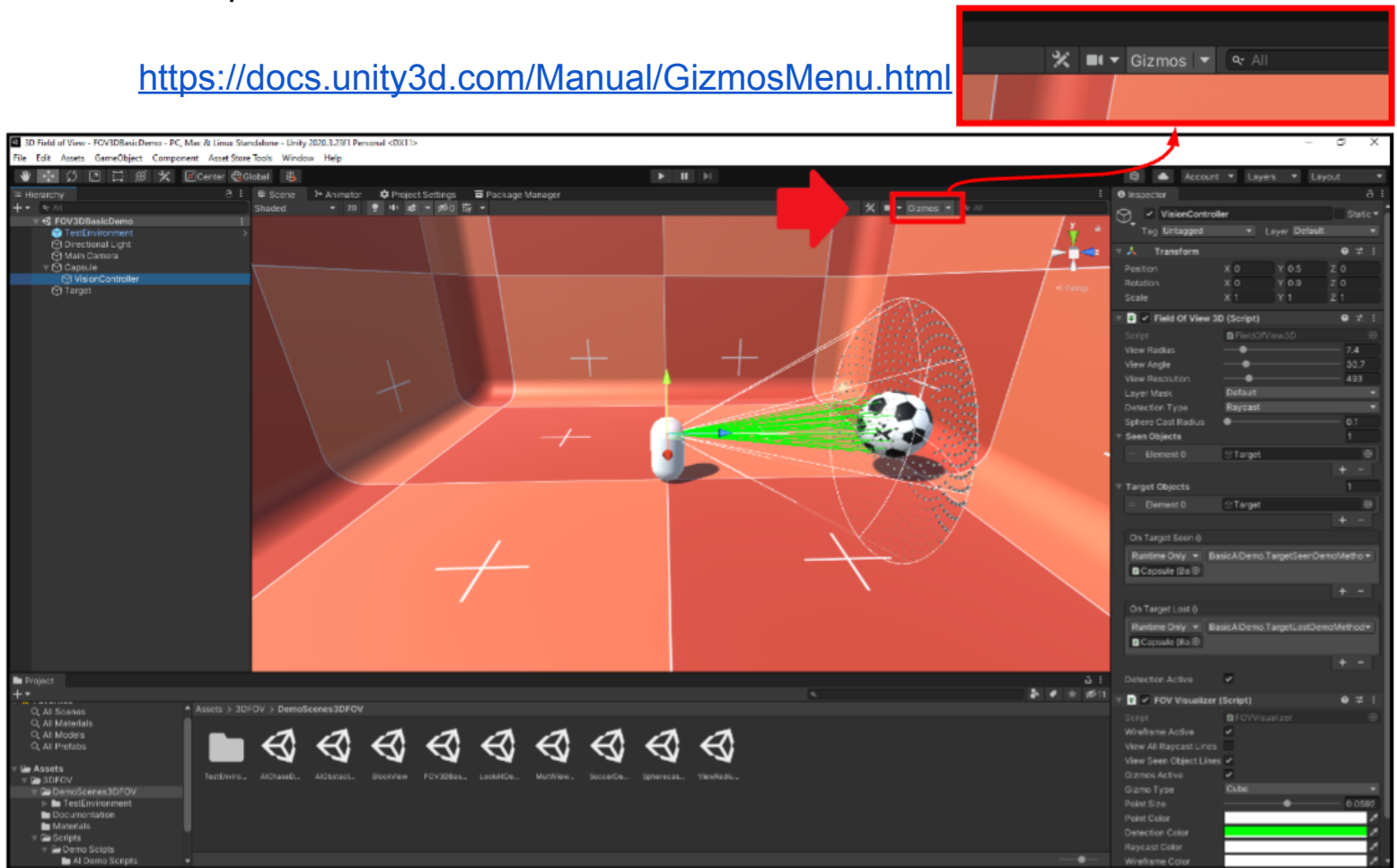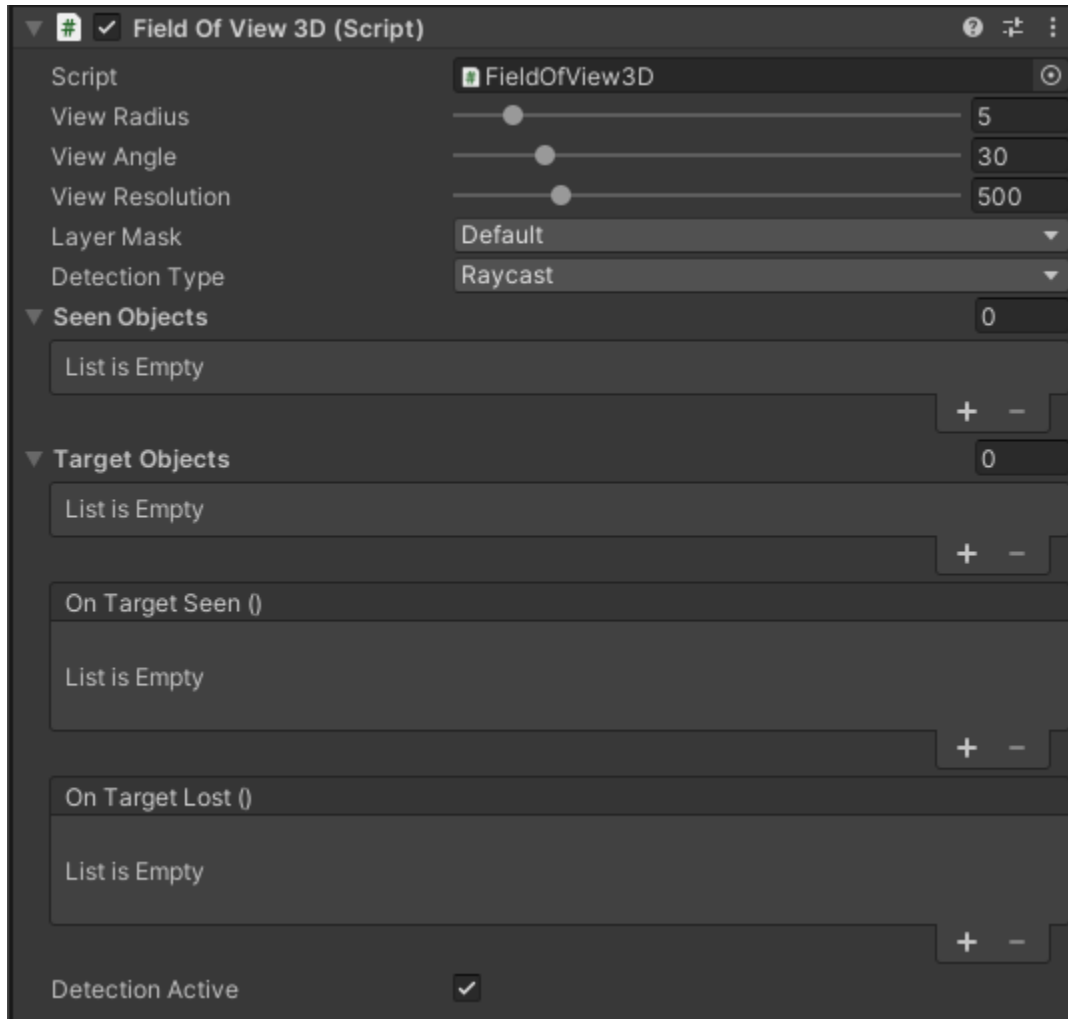# 3D Field of View

Setup

To use this asset to its full extent, make sure the "Gizmos" button is selected in both the Scene view and Game view. The Gizmos menu at the top of the Scene view and Game view window.

https://docs.unity3d.com/Manual/GizmosMenu.html

USAGE INSTRUCTIONS

Simply drag and drop the scripts "FieldOfView3D.cs" and "FOV Visualizer.cs" onto an empty game object.

FieldOfView3D.cs

View Radius:

How far the engine sees.

View Angle:

How far reaching the engine peripheral vision is, with a max of 180° for perfect 360° vision.

Important Note: Increasing the View Angle decreases the View Resolution because of the square cube law. Because of this, you may want to increase the View Resolution at greater View Angles for equal coverage. See the MultiView scene for a demonstration of this principle.

View Resolution:

The amount of iterations within the view cone to check. The Higher the value, the more accurate the simulation. Values around 1000 are usually fine, values above 5000 may cause lag.
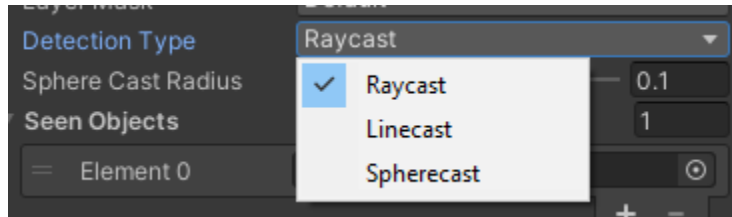
Layer Mask:

The layers that the engine can detect. For the engine to detect an object, the object must have a collider, and the object's layer must be selected in "Layer Mask".
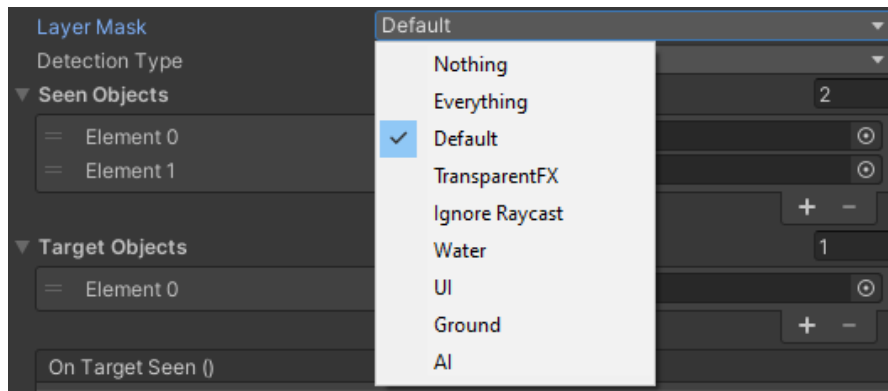
FieldOfView3D.cs cont.

Detection Type:

Raycast/ Linecast:

In practice, these are virtually indistinguishable from one another. Raycast is slightly faster, while Linecast is slightly more accurate. However, these differences are negligible, and can be widely ignored.

Spherecast:

Spherecast is essentially a "fat raycast". This allows you to increase the resolution of your detection without having to add more points to check, which in some cases improves performance. This is also useful to make sure no small objects slips through your FOV undetected. *See the SpherecastDemo for a demonstration.*

Spherecasting will affect fine point accuracy, and in rare cases, it can detect objects slightly outside its range. In most cases, Raycast is fine.
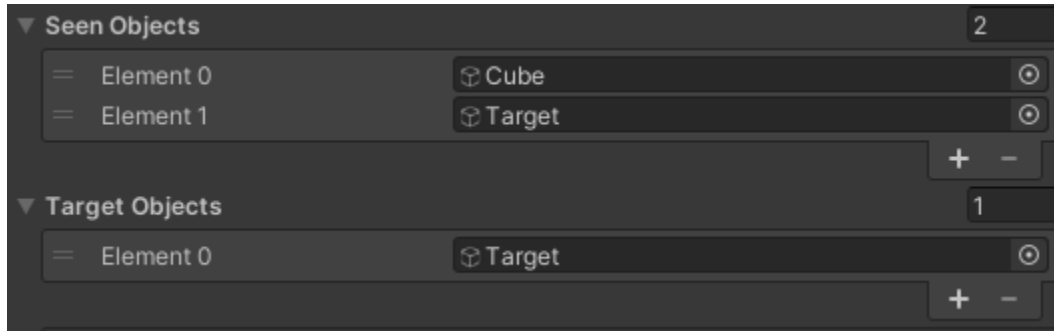
Spherecast Radius:

The radius of the sphere to cast. This value is only applicable if "Detection Type" is set to "Spherecast."
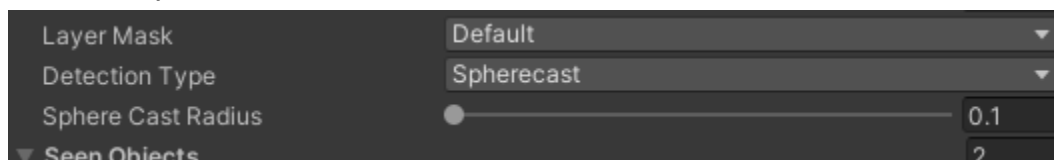
Seen Objects: A list of what the engine currently sees. This can be referenced by other scripts.

Target Objects: Add objects to this list for use of OnTargetSeen() and OnTargetLost().

OnTargetSeen() / OnTargetLost():

 Unity Events for developers to assign predefined methods.

 OnTargetSeen() is called first when the target is among the "Seen Objects" list.
 OnTargetLost() is called once when the target is no longer among the seen objects.

 For proper functionality, methods called must be designated with the modifier "public".
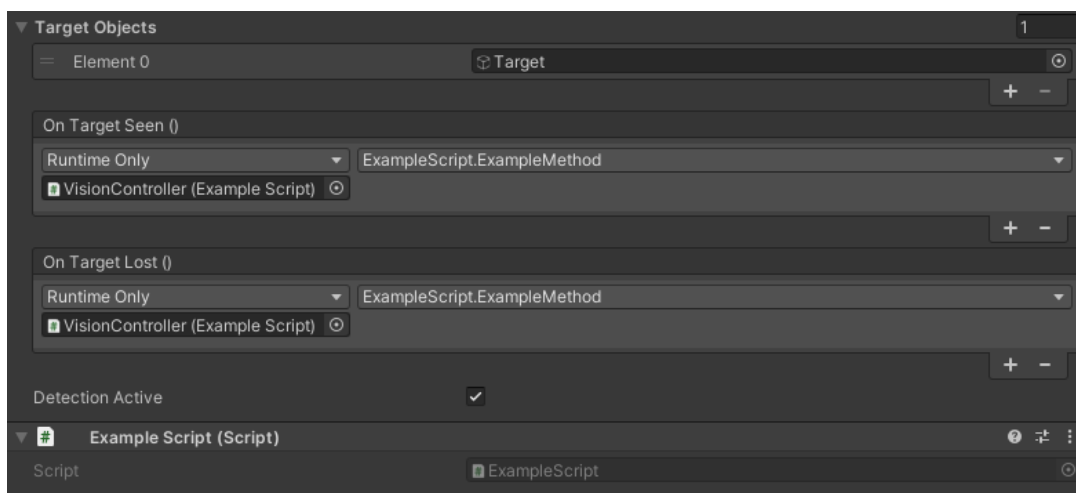
```
public void ExampleMethod()
{
    ///Do Stuff
}
```

These are assigned and called very similarly to the Unity Button "On Click Event".
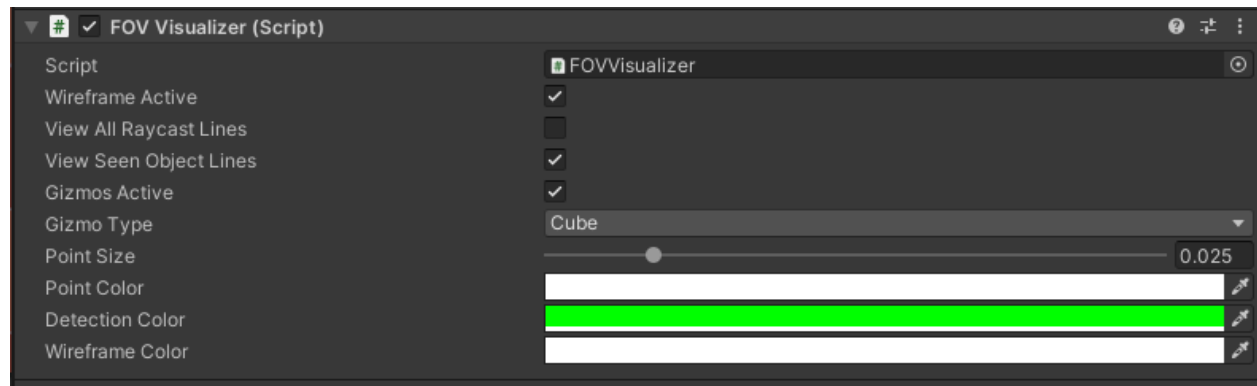*For more information, see*

  https://docs.unity3d.com/Packages/com.unity.ugui@1.0/manual/script-Button.html
  https://docs.unity3d.com/Manual/UnityEvents.html

Detection Active: Engine is currently running.

FOVVisualizer.cs

FOVVisualizer:

FOVVisualizer is used to help the developer visualize what the engine can see. This is just an editor tool, however, and will not be included in the final build.

Wireframe Active: Cosmetic bool to allow developer to see outer edges of the vision cone.

View All Raycast Lines: Cosmetic bool to allow developer to see internal resolution of the vision cone.

View Seen Object Lines: Cosmetic bool to allow developer to see what the engine sees, and where the engine sees it.

Gizmos Active: Cosmetic bool to allow developer to see outer boundary of the vision cone, using Gizmo DrawSphere.

Gizmos:
> Point: Basic setting
> Ray: Useful for visualizing the raycast without obscuring the screen.
> Cube: Recommended Setting for combination of lag and prominence.
> Sphere: May cause substantial lag at higher view resolutions.
> Disc: May cause substantial lag at higher view resolutions.

Point Size: Cosmetic float to define size of Gizmos.