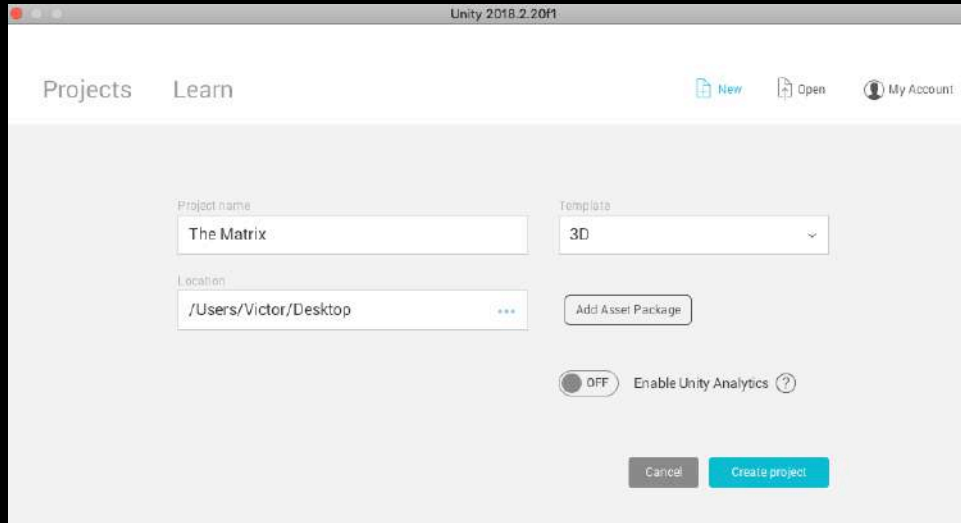# Hands on tutorial: The Matrix vs. Tom Cat



Supported by Udemy course: "Máster Realidad Aumentada con Unity 3D, Vuforia y C#"

# New Unity + Vuforia Project

**Option 1: The Matrix**
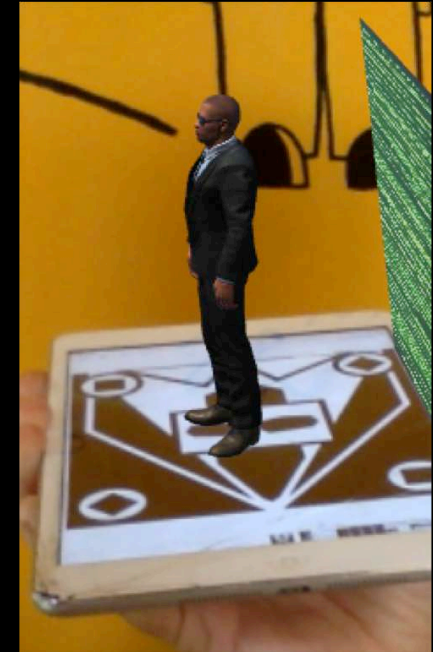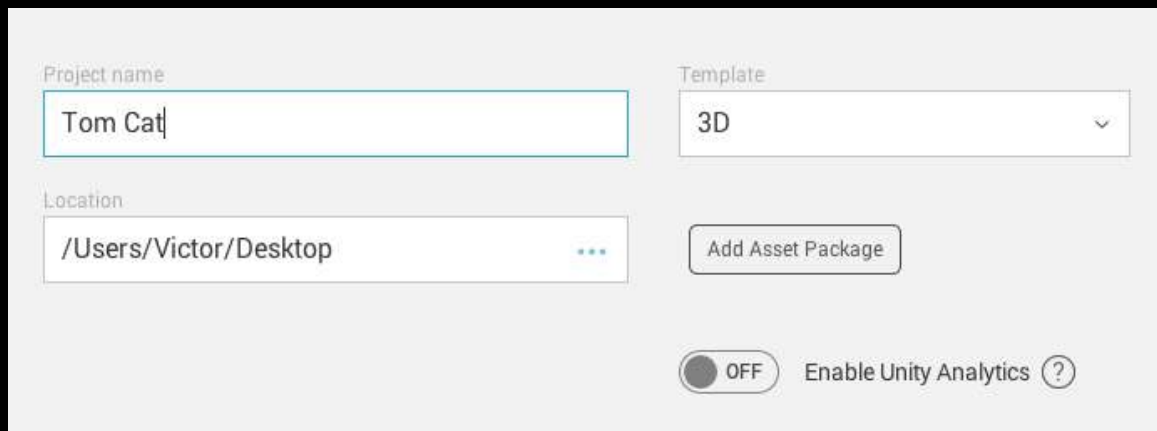


**Option 2: Tom Cat**

# Import Vuforia Assets
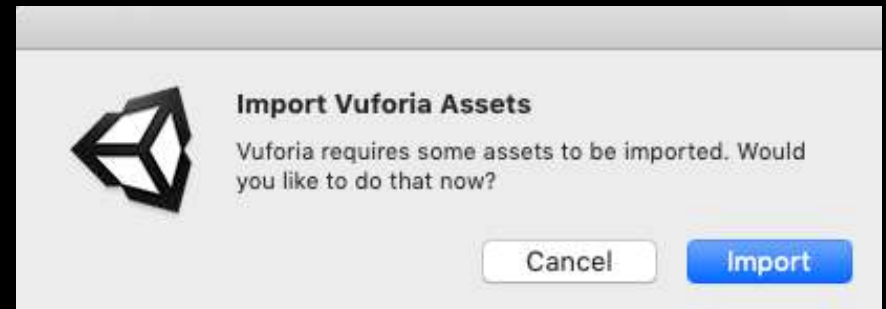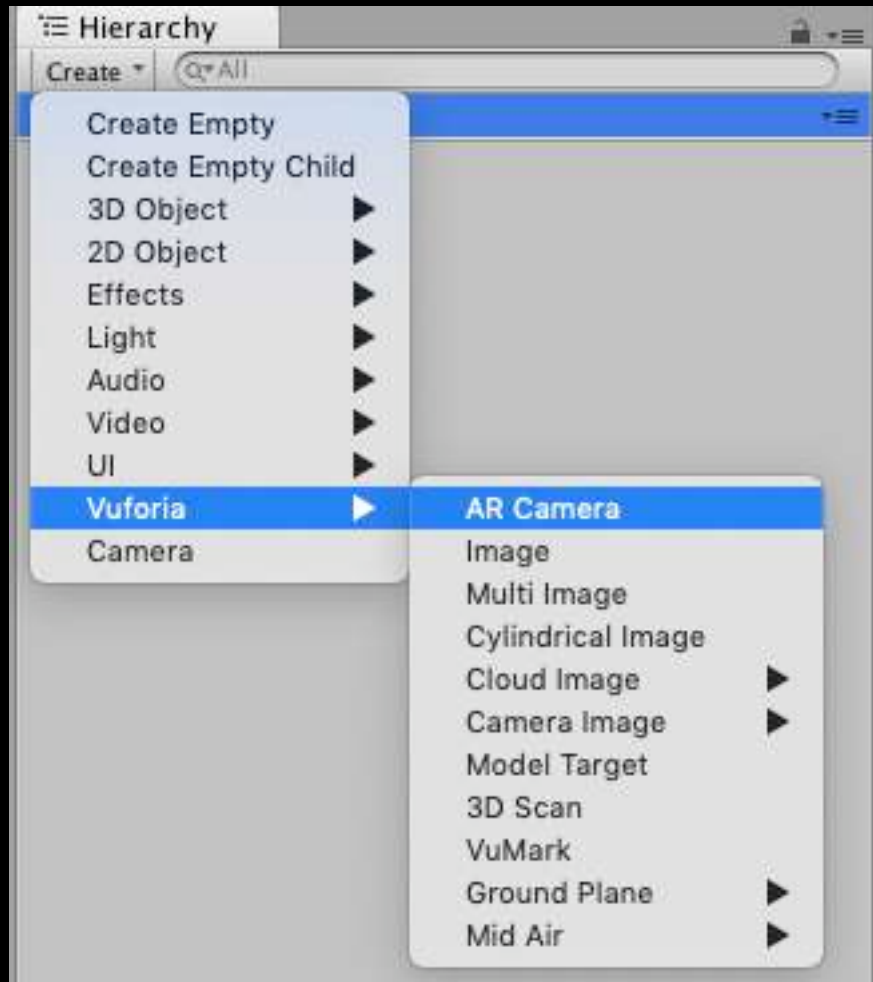
# Remove Main Camera and configure your build settings for Android/iPhone
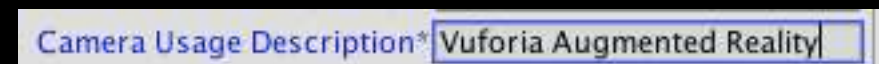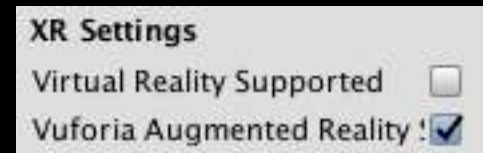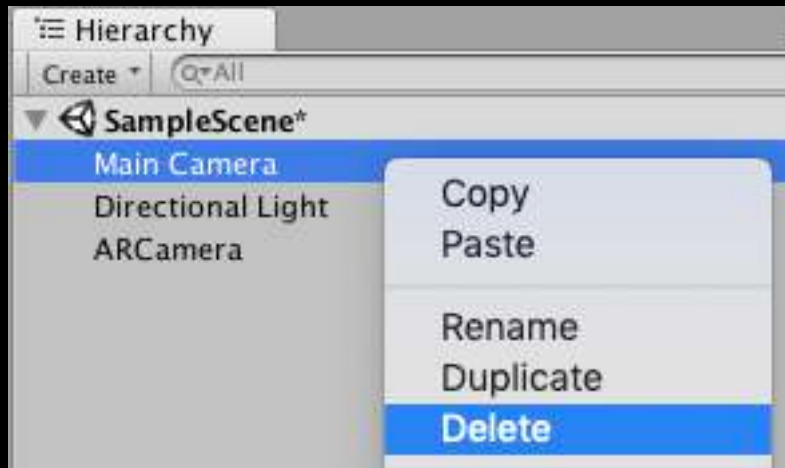
# Part 1
# Matrix Scene: Agent Smith (image target)
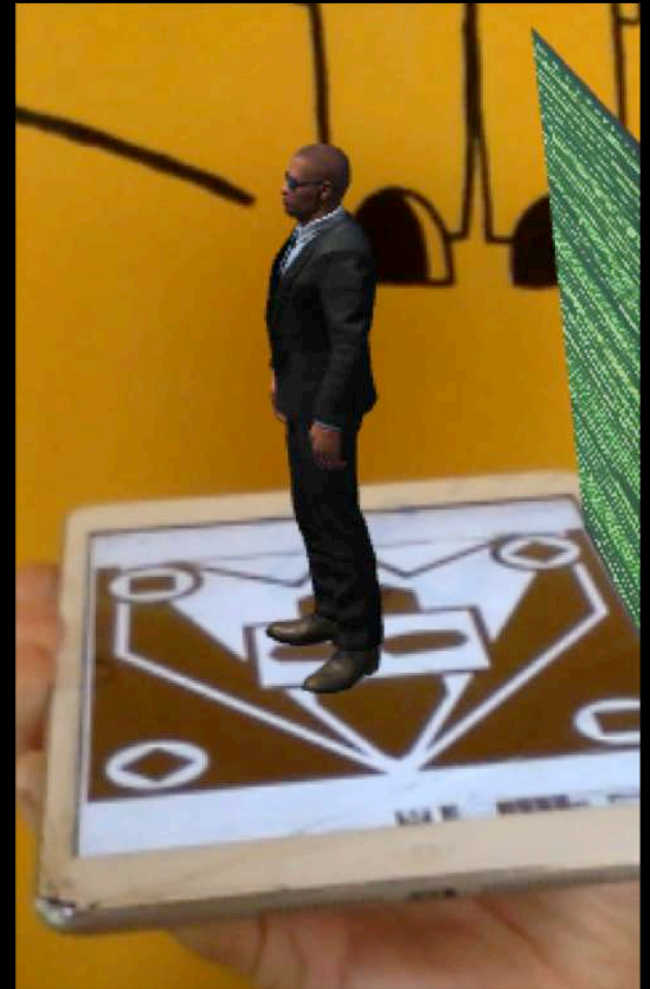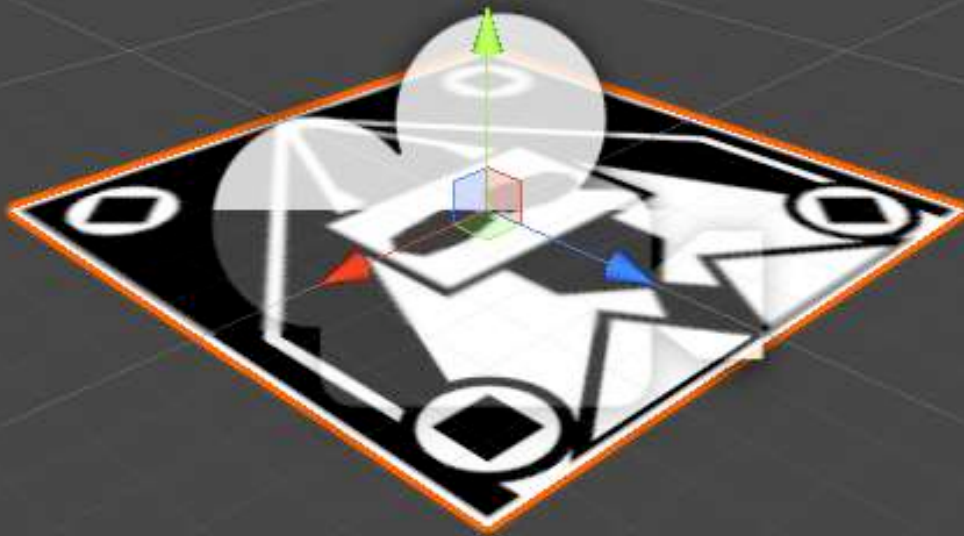
# Alternative Tom Cat Scene

# Add license – copy Vuforia license key

# Add & Import Image Target
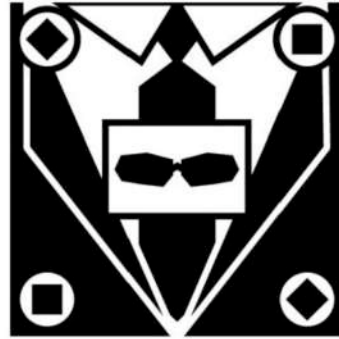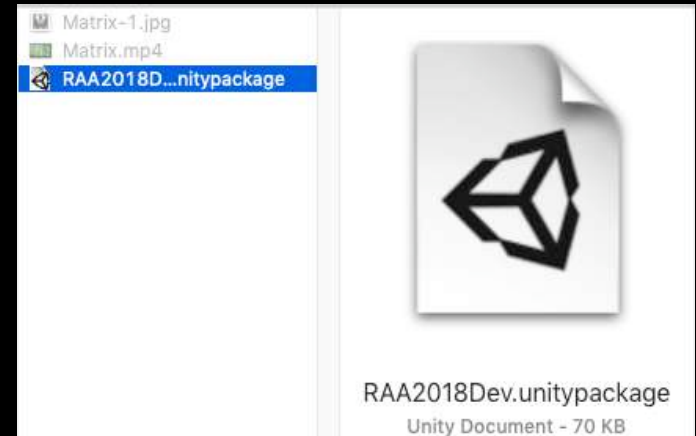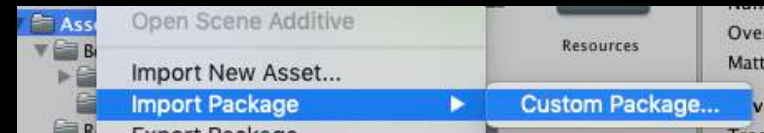
# Add Image Target to Scene



Older versions: Unity finds and adds the image target automatically

Newer versions: Fix Image Target Behaviour and Vuforia Configuration Databases

# Handling Target Events

```
ImageTarget
```

**Default Trackable Event Handler (Script)**

Script                    DefaultTrackableEventHandler

```
// Vuforia is starting,
// Call OnTrackingLost()
OnTrackingLost();
```

```csharp
if (newStatus == TrackableBehaviour.Status.DETECTED ||
    newStatus == TrackableBehaviour.Status.TRACKED ||
    newStatus == TrackableBehaviour.Status.EXTENDED_TRACKED)
{
    Debug.Log("Trackable " + mTrackableBehaviour.TrackableName + " found");
    OnTrackingFound();
}
```
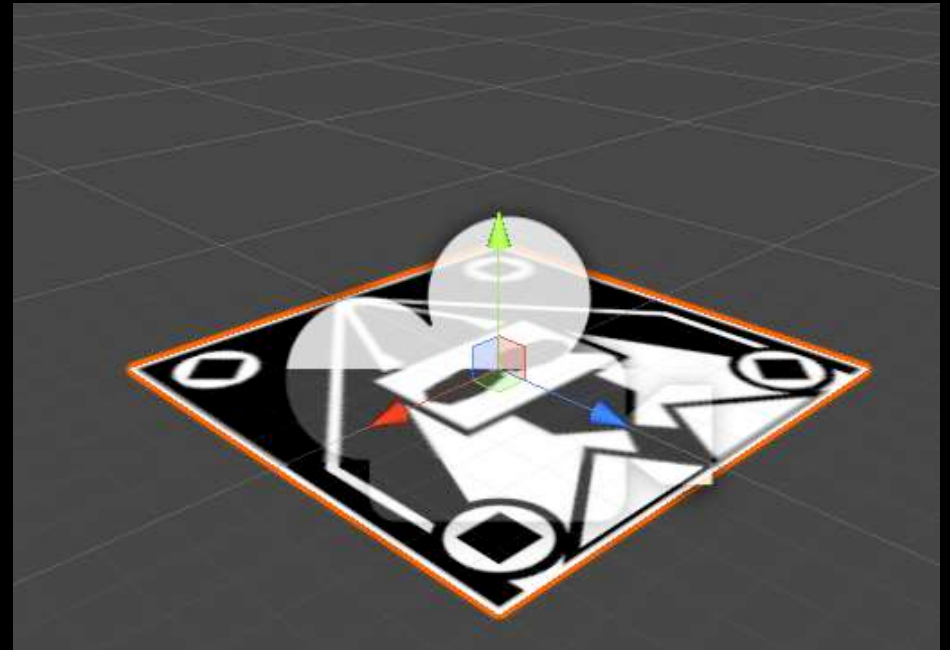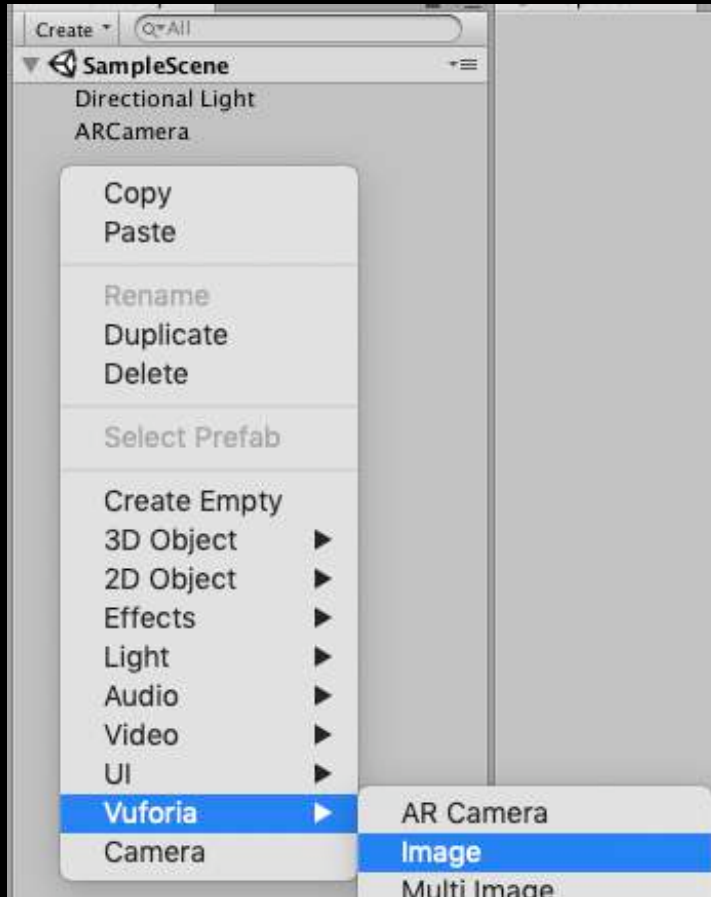
# Customised target handler (copy & paste, edit)

Replace Default Trackable
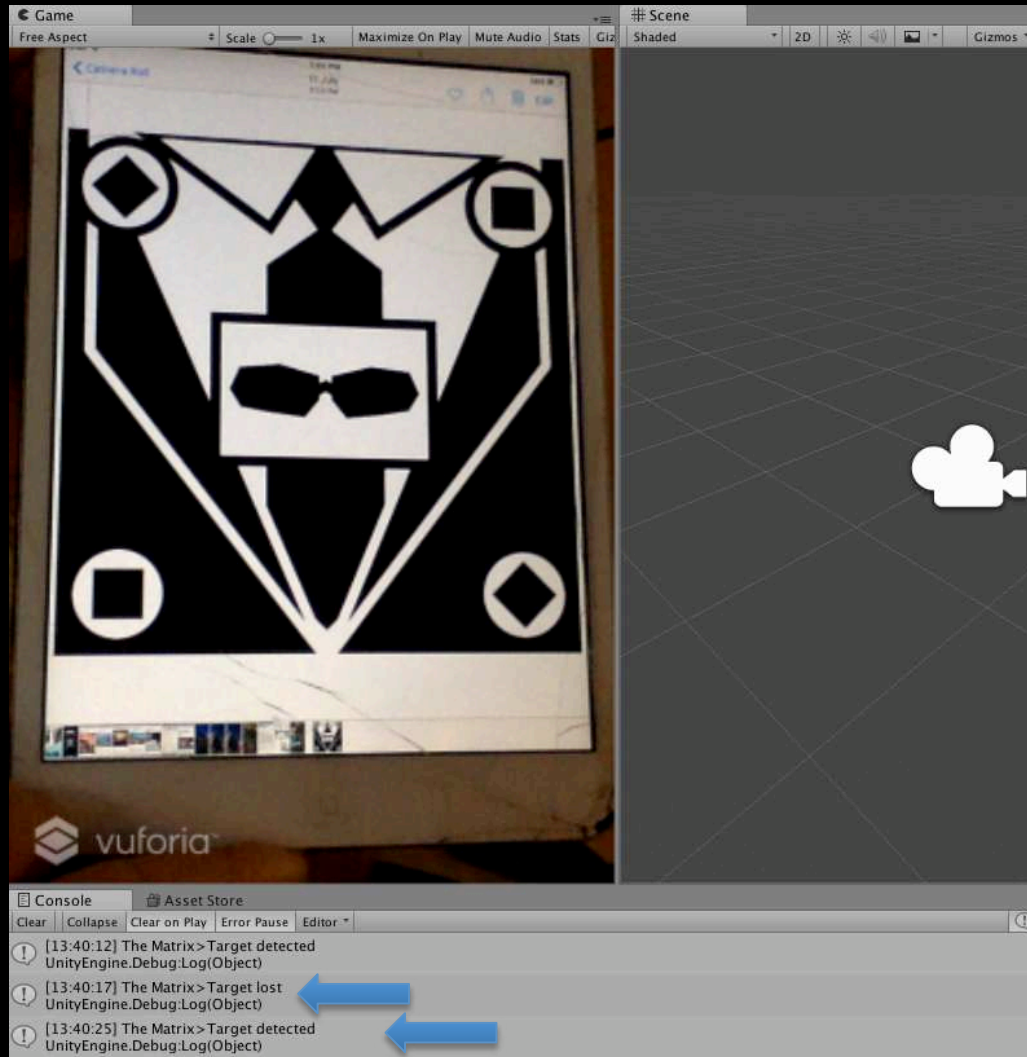Event Handler (Script) with
new customised
Event Handler

```csharp
using UnityEngine;
using Vuforia;

public class TheMatrix : MonoBehaviour, ITrackableEventHandler
{
    protected TrackableBehaviour mTrackableBehaviour;

    void Start () {
        mTrackableBehaviour = GetComponent<TrackableBehaviour>();
        if (mTrackableBehaviour)
            mTrackableBehaviour.RegisterTrackableEventHandler(this);
    }

    public void OnTrackableStateChanged(
        TrackableBehaviour.Status previousStatus,
        TrackableBehaviour.Status newStatus)
    {
        if (newStatus == TrackableBehaviour.Status.DETECTED ||
            newStatus == TrackableBehaviour.Status.TRACKED ||
            newStatus == TrackableBehaviour.Status.EXTENDED_TRACKED)
        {
            Debug.Log("The Matrix>" + "Target detected");
        }
        else if (previousStatus == TrackableBehaviour.Status.TRACKED &&
                 newStatus == TrackableBehaviour.Status.NO_POSE)
        {
            Debug.Log("The Matrix>" + "Target lost");
        }
        else
        {
            Debug.Log("The Matrix>" + "Tracking starts");
        }
    }
}
```
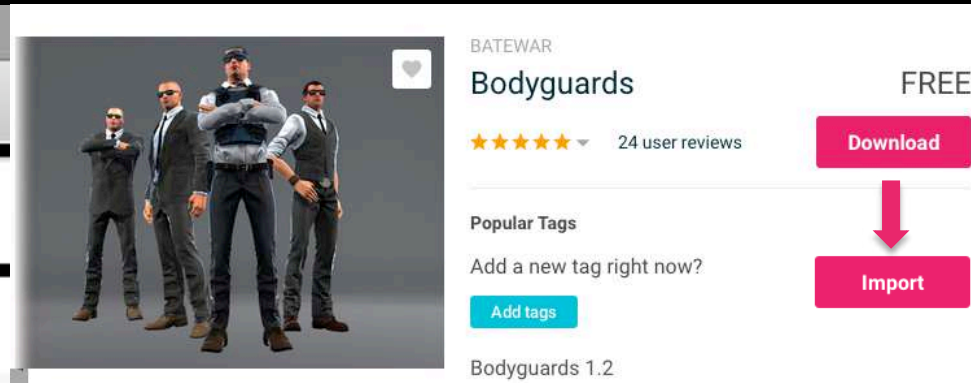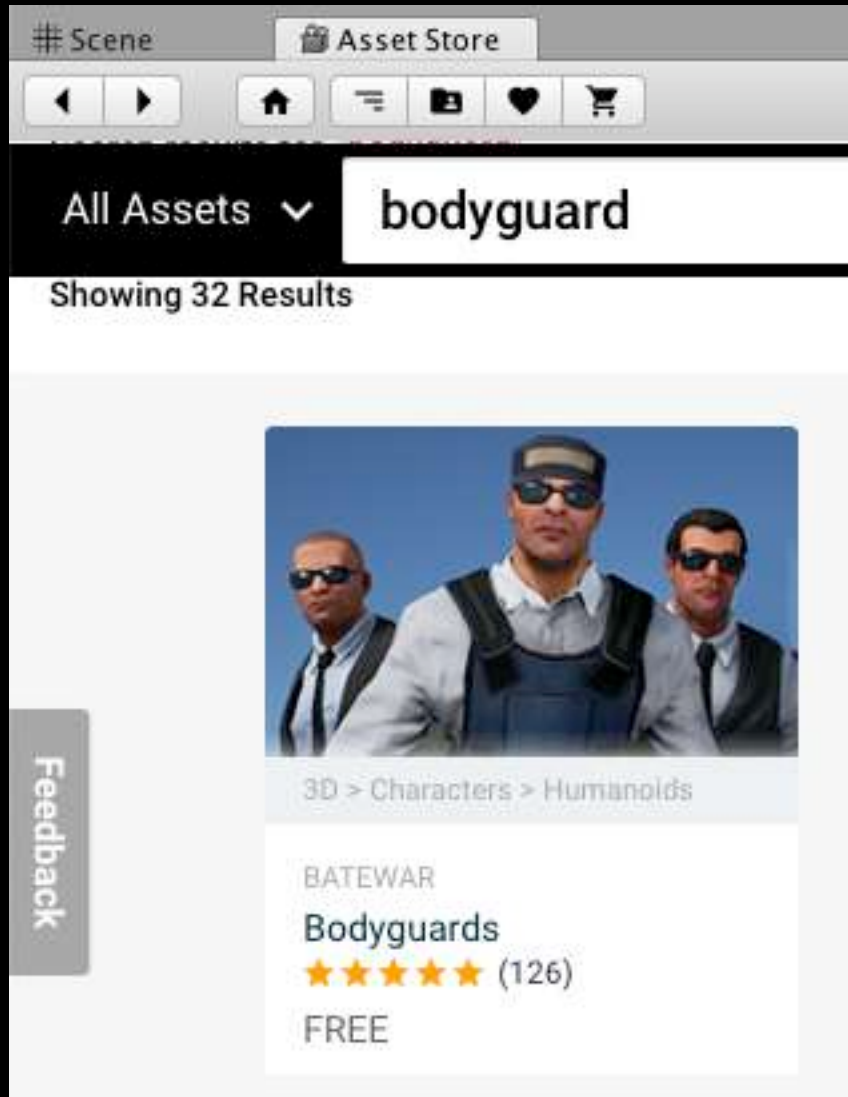
Edit   Assets   GameObject   Co

Undo Selection Change   ⌘Z
Redo   ⇧⌘Z

Cut   ⌘X
Copy   ⌘C
Paste   ⌘V

**Duplicate**   ⌘D

C#

UserDefinedEventH...

ImageTarget

☐ Default Trackable Event Handler (Scr
Script        DefaultTrackableEventHandler

☑ User Defined Event Handler (Script)
Script        UserDefinedEventHandler

# Handling Target Events

# Download and import a (free) 3D Model 😎
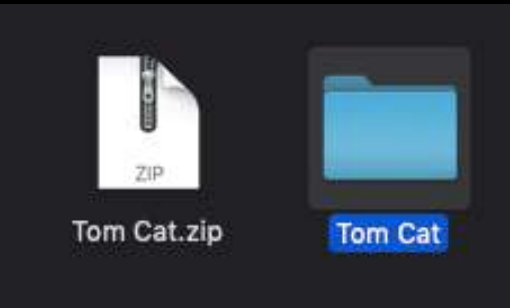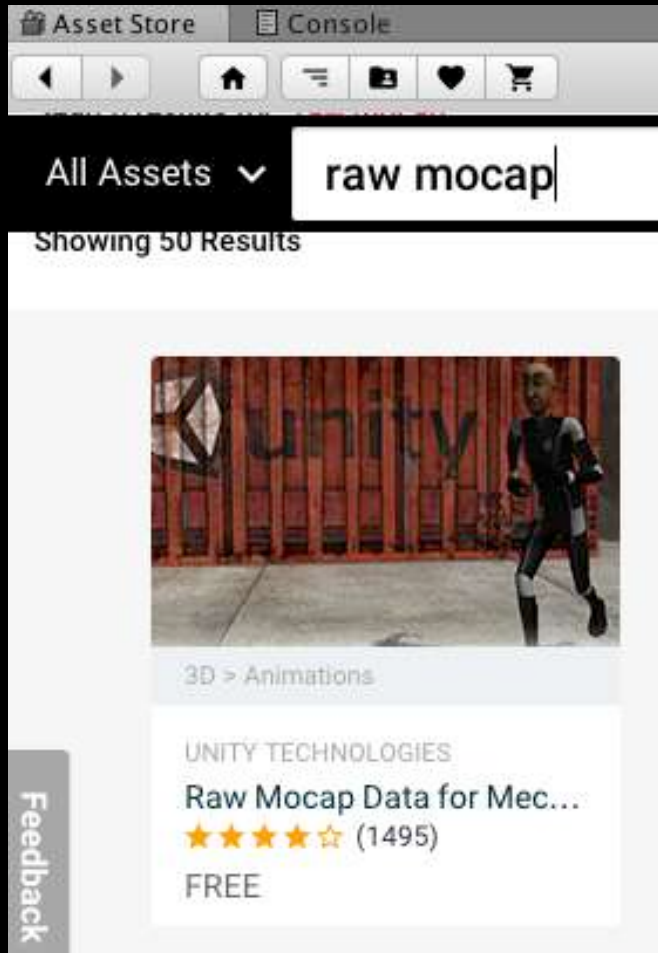
# Import Cat 3D Model 😼

**Tom Cat**



Fix textures    Hierachy: Cat: Cat
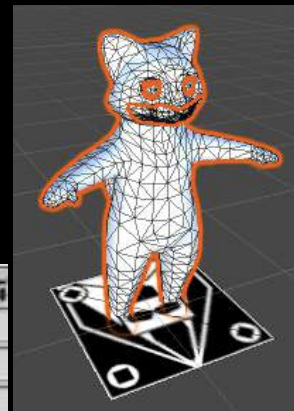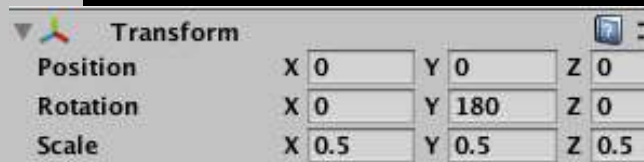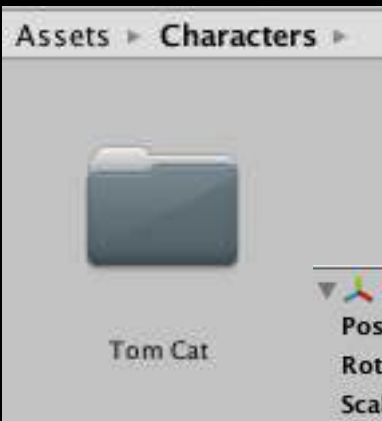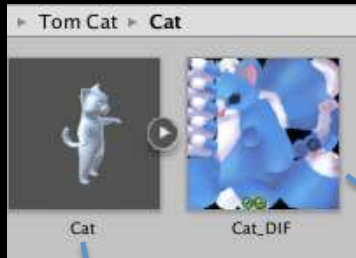
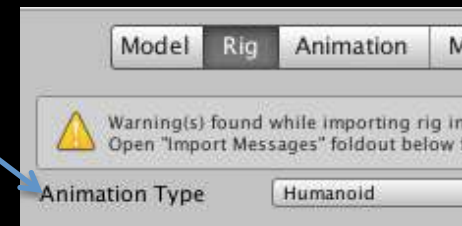# Download and import animation



Character Animation Type: Humanoid 👆🏼

1 month ago: Free Unity animations 👆🏼
Today:

Unity humanoid animations

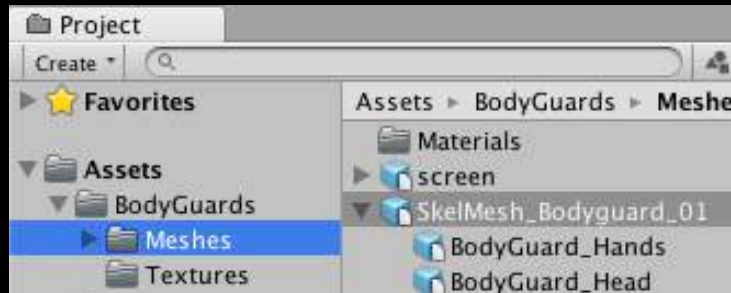Default animation for the character 👎🏼

# Setting up our 3D model
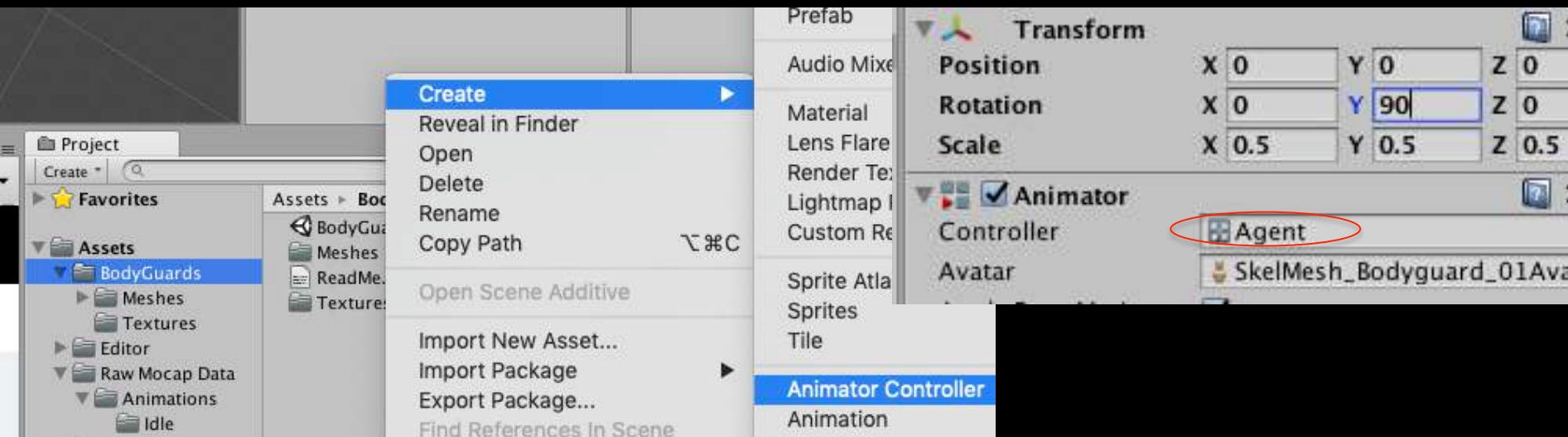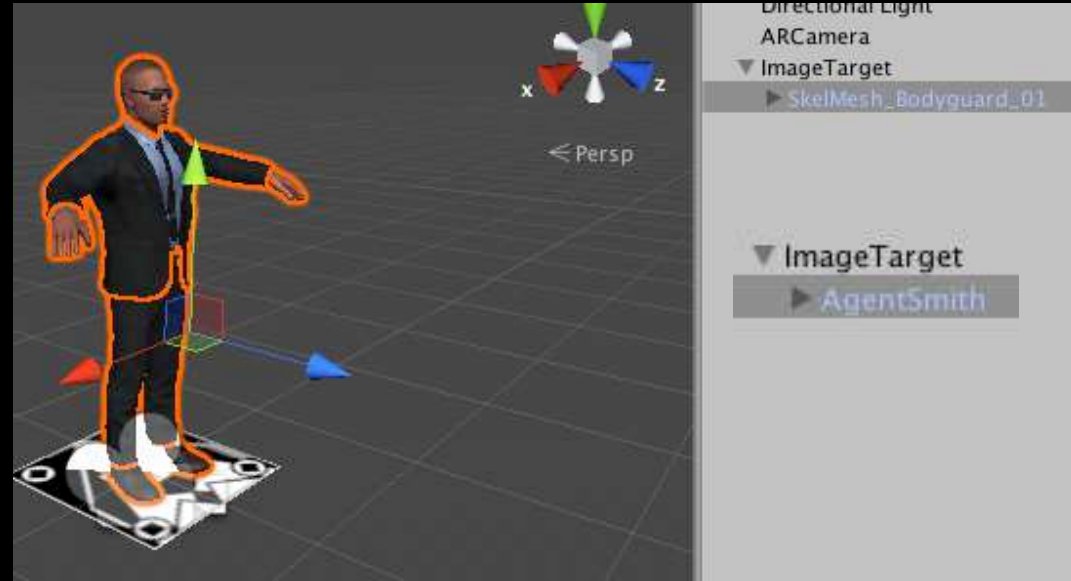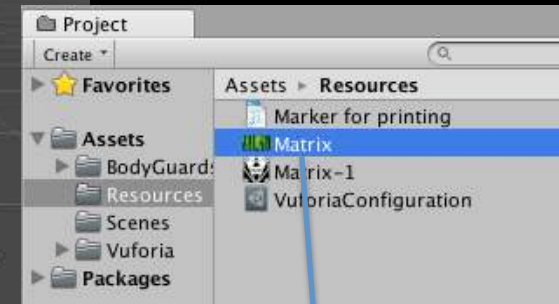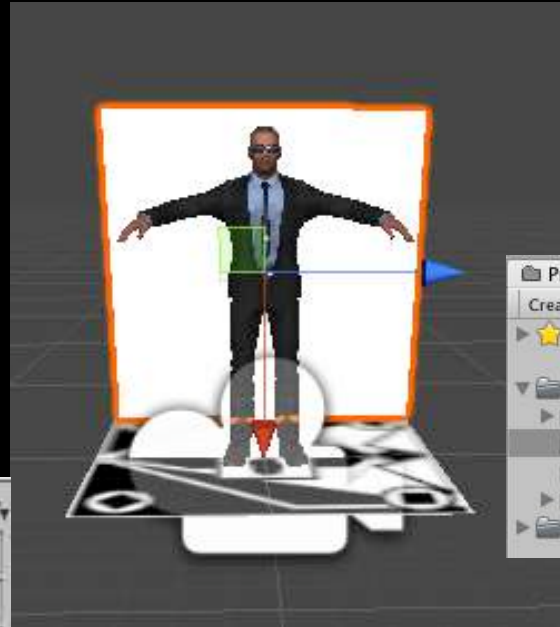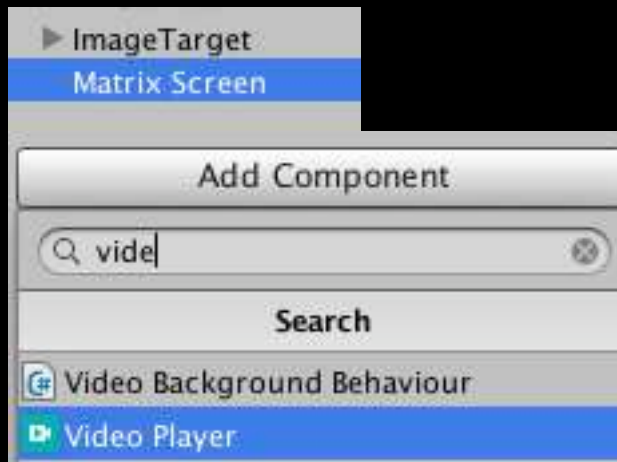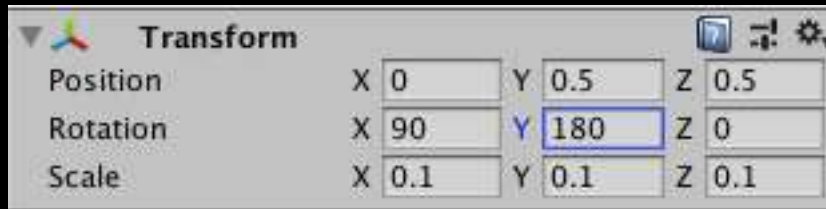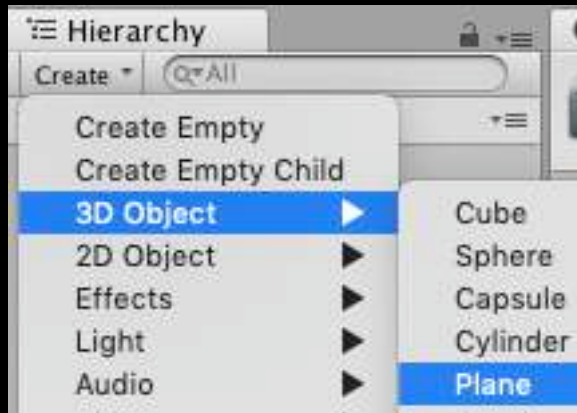
# Setting up our animation



Create new Animator Controller
for the character and choose animation
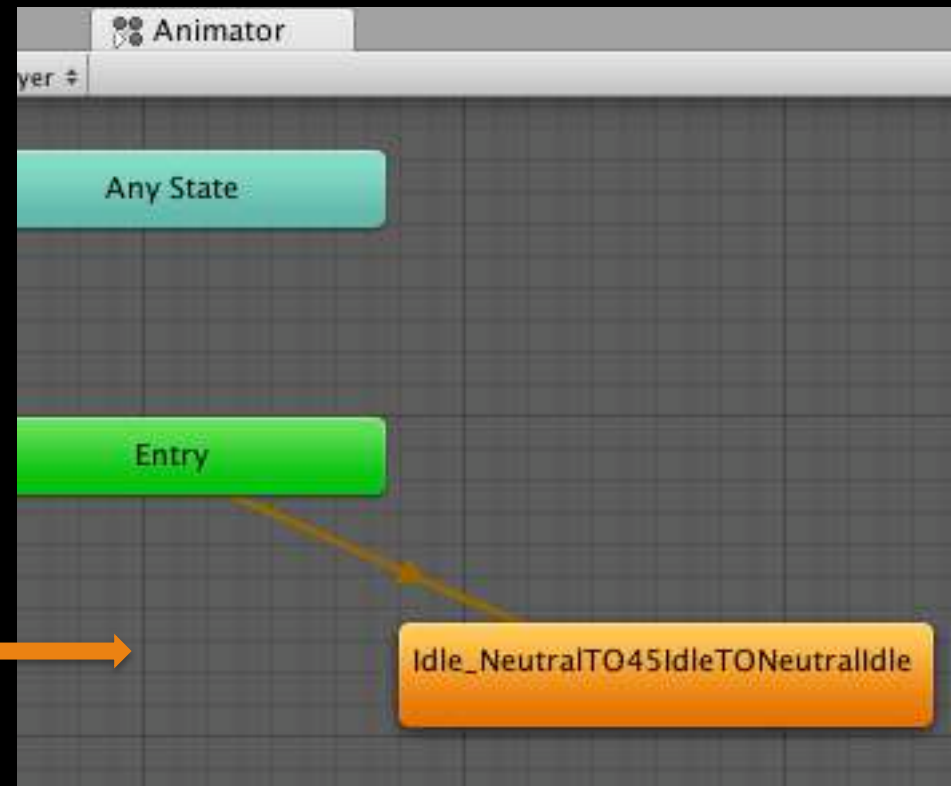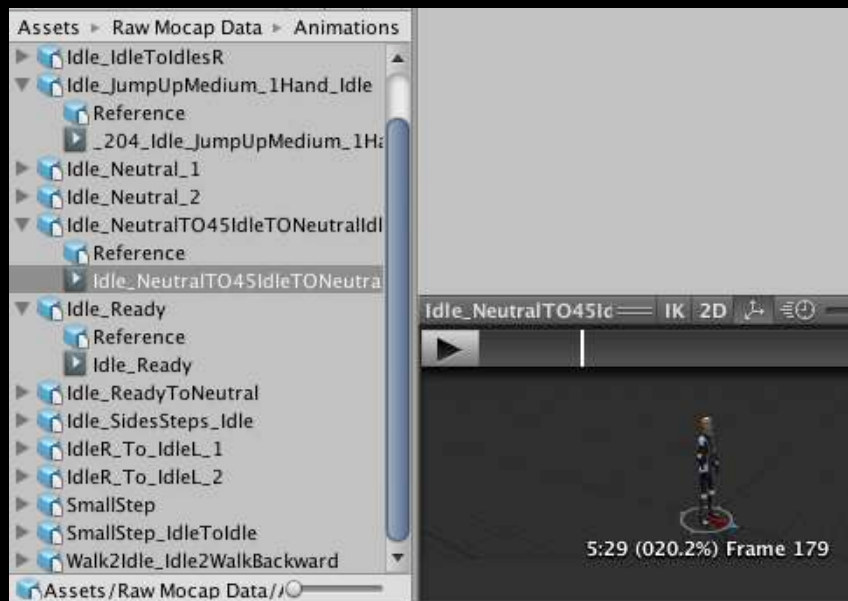
# Adding a background Video Player
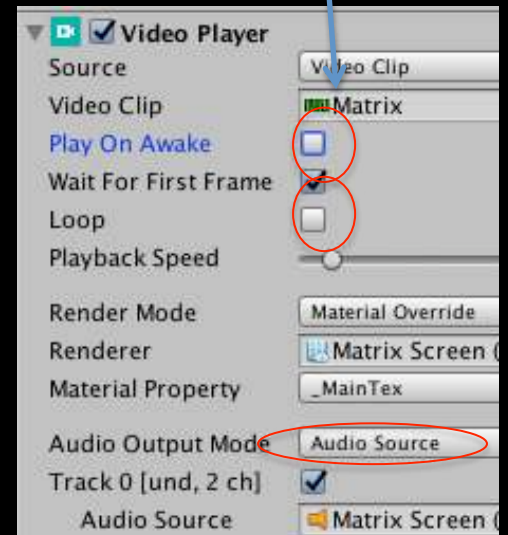
# Video plays when target is detected

```
3   using UnityEngine;
4   using Vuforia;
5   using UnityEngine.Video;
6
7   public class TheMatrix : MonoBehaviour, ITrackableEventHandler
8   {
9       protected TrackableBehaviour mTrackableBehaviour;
10      public VideoPlayer matrixScreen;
11
12      void Start () {
13          mTrackableBehaviour = GetComponent<TrackableBehaviour>();
14          if (mTrackableBehaviour)
15              mTrackableBehaviour.RegisterTrackableEventHandler(this);
16      }
17
18      public void OnTrackableStateChanged(
19          TrackableBehaviour.Status previousStatus,
20          TrackableBehaviour.Status newStatus)
21      {
22          if (newStatus == TrackableBehaviour.Status.DETECTED ||
23              newStatus == TrackableBehaviour.Status.TRACKED ||
24              newStatus == TrackableBehaviour.Status.EXTENDED_TRACKED)
25          {
26              //    Debug.Log("The Matrix>" + "Target detected");
27              matrixScreen.Play();
28          }
29          else if (previousStatus == TrackableBehaviour.Status.TRACKED &&
30                  newStatus == TrackableBehaviour.Status.NO_POSE)
31          {
32              //    Debug.Log("The Matrix>" + "Target lost");
33              matrixScreen.Pause();
34          }
35          else
36          {
37              // Debug.Log("The Matrix>" + "Tracking starts");
38              matrixScreen.Stop();
39          }
40      }
41  }
```

Set Video Screen object on the Event Handler Script
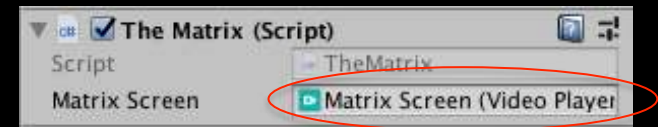
# Ta Daaa!
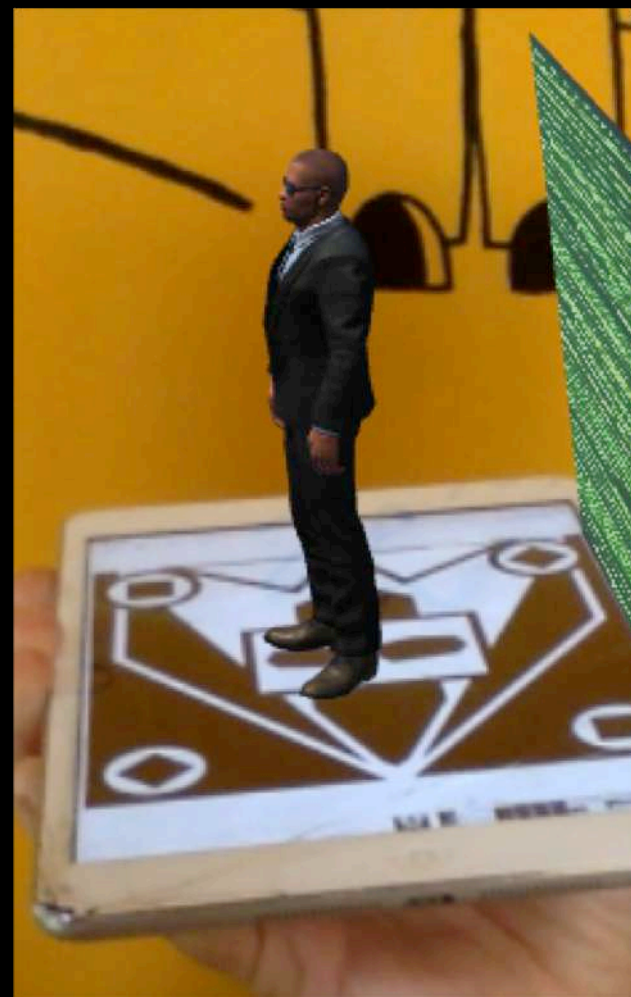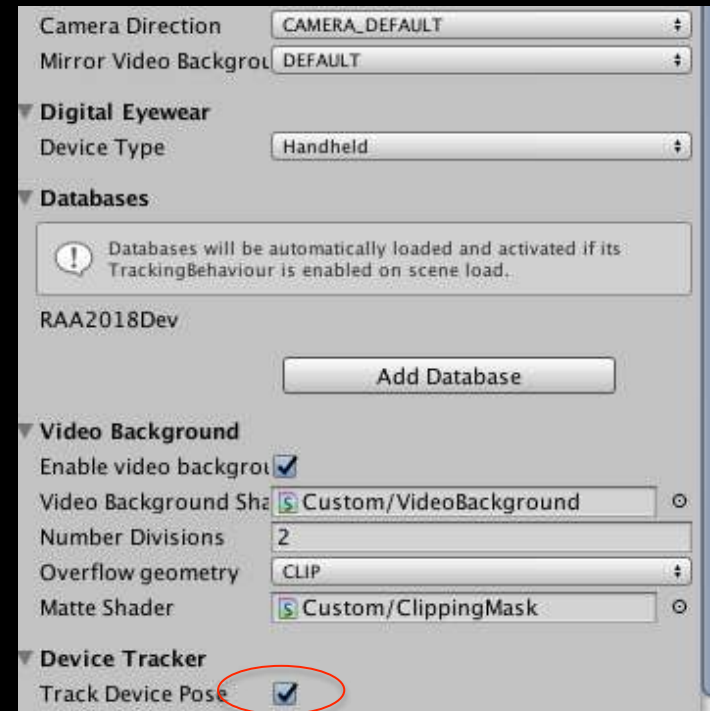
# Scene 2: Interacting with the character
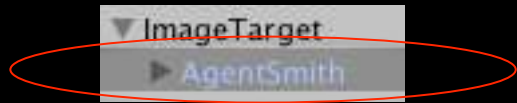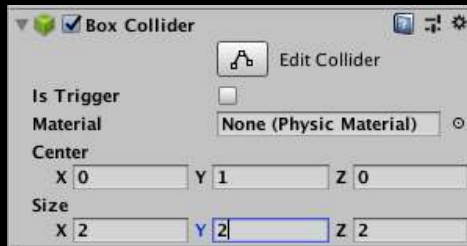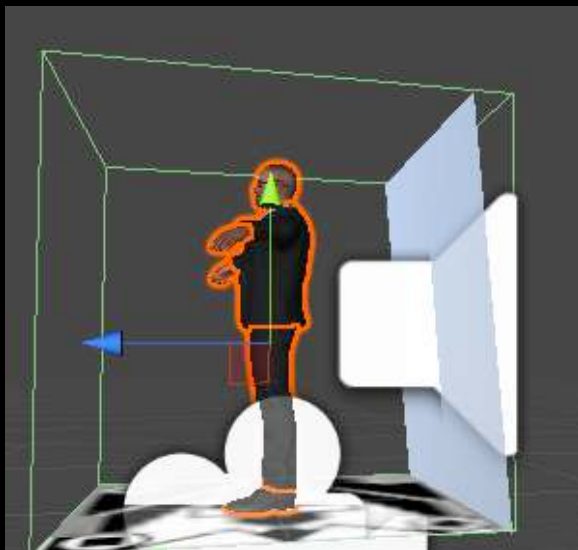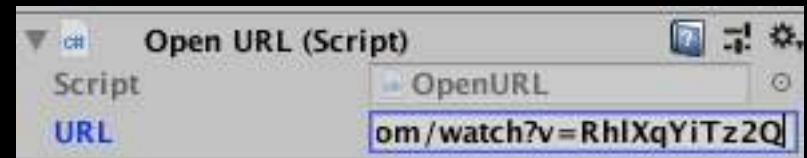
# Scene 2: Position tracking
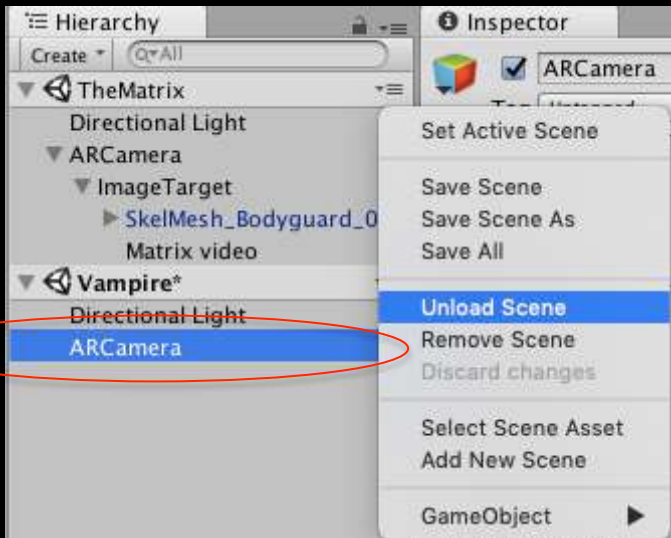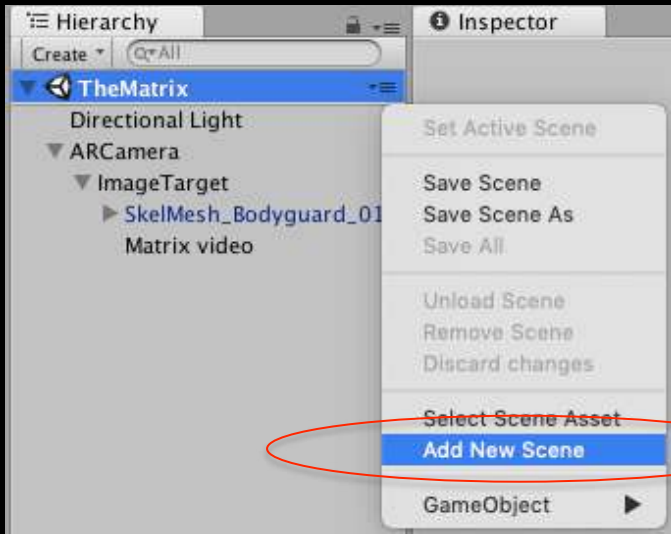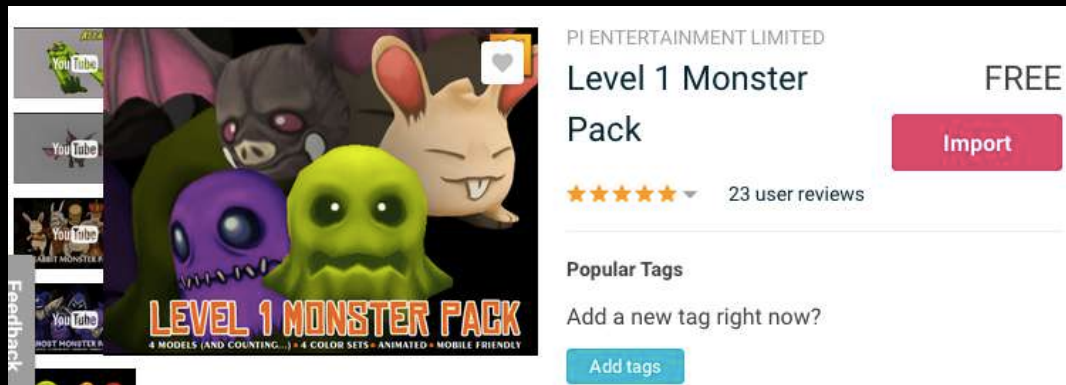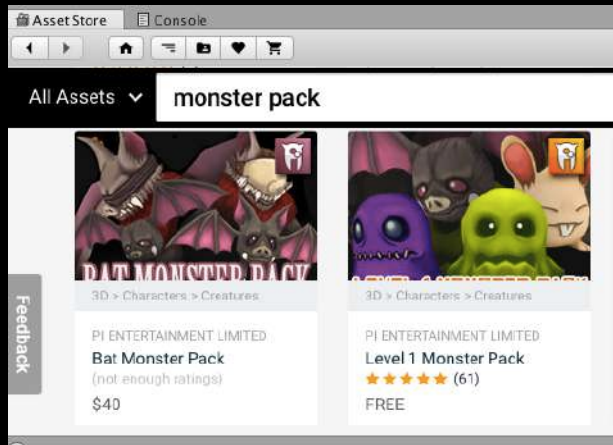


Positional only for latest phones (ARcore/ARkit)
* using the camera to calculate distance
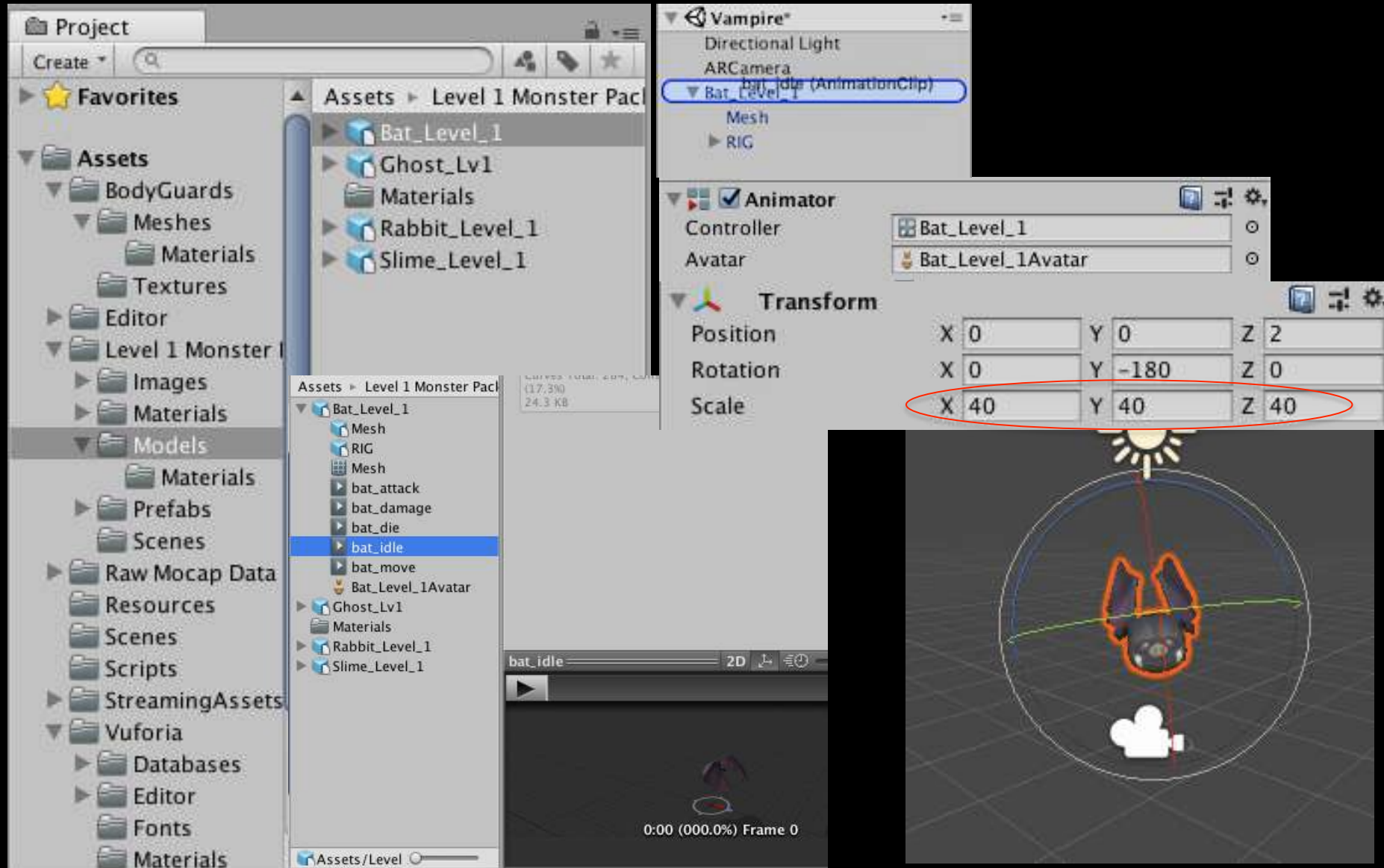Rotational for older phones using sensors
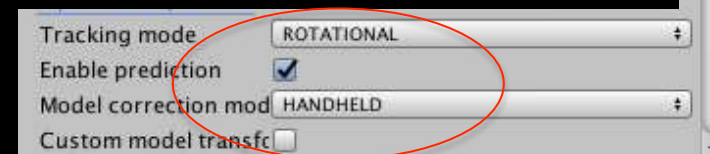
# New character: Dracul-bat
# (or Cat Bat)

# Add character+animation to the scene

# Monster Cat Bat

Extract and replace material

# Scor🦇 + Box Collid🦇r

## Text: **Catch Draculbat / Catbat**

### Box Collider:



**Canvas:**

# Scripting

# Build Android App

# Build iOS App



Open this folder with Xcode

Catch Draculbat!!

Catch Catbat!