Camera Orbit Tool by Oluss Studio
Ver. 8.1 – Aug 2014

Table of Contents

1. Main Feature

The main function of this tool is to create a camera set up where a single object-of-interest is the focus of the scene. The implementation of this is potentially limitless. An object-of-interest in a 3D environment does not really mean that it has to be a single object in the real world – It can be a product, schematic, furniture, vehicle, layout of a room, building, or even a town – basically anything as far as they are made of polygons.

2. Setting It Up

2a. Pan and Tilt Orbit

Here are a few simple steps to get this tool up and running for your own scene:

1. Import the Camera Orbit Tool to your Unity project.
2. Open the setUpBasic.unity
3. Replace the objectOfInterest with your own object-of-interest.
4. Adjust the scale of your object.
5. Select the horizontalPivot from the Hierarchy tab to adjust the parameters of your camera to suit your need (See section 5b. for details).
6. Select the verticalPivot from the Hierarchy tab to adjust the parameters of your camera to suit your need (See section 5c. for details).

If you are intending to just use the scripts to set up your own scene, follow the logic of the following steps. Please note that different cases may allow different amount of steps:

1. Import the scripts from the Camera Orbit Tool to your Unity project.
2. Create an empty game object.
3. Name it horizontalPivot.
4. Position horizontalPivot at 0, 0, 0.
5. Create another empty game object.
6. Name it verticalPivot.
7. Nest it to be the child of horizontalPivot.

8. Position verticalPivot at 0, 0, 0.
9. Nest your scene camera to be the child of verticalPivot.
10. Get your object-of-interest to the scene and place it at 0, 0, 0. Note: It does not have to be at 0, 0, 0 all the time, but in principle, your object-of-interest should be located at the same spot as your two pivots.
11. Position and rotate your camera so that it looks toward your object-of-interest at a comfortable default angle. Note: you could also adjust the field of view parameter of your camera.
12. Attach HoriPivot.js to the horizontalPivot object. Recommended value for *Rotation Speed* is 5 and *Lerp Speed* is 2. Refer to part 5 of this pdf for details on these two variables.
13. Attach VertPivot.js to the verticalPivot object. Please mind that both HoriPivot.js and VertPivot.js have its own *Rotation Speed* variable. Refer to part 5 of this pdf for details on the variables of VertPivot.js
14. If you want your camera to have the zoom function, attach the PinchZoom.js to you camera, not to any other objects. Make sure that the projection of you camera is set to *Perspective*, not *Orthographic*.

2b. Spherical Orbit

If your object of interest has a proportion like a sphere or cube, you can quickly set up a spherical orbit camera by doing the following:

1. Import the Camera Orbit Tool to your Unity project.
2. Open the setUpSphereOrbit.unity
3. Replace the objectOfInterest with your own object-of-interest.
4. Adjust the scale of your object.
5. Select the horizontalPivot or verticalPivot object from the Hierarchy tab and adjust the parameters of your camera to suit your need (See section 5d. for details).

3. Outcome

By setting up a scene in accordance to this tool, you could easily get a scene where the object can be panned and rotated 360 degrees (although doing 360 degrees mostly gives the users a bad headache!). The scene is controllable via left mouse button and dragging on PC, Mac and Linux, as well as swiping on touch screens. The function and code used in this tool is the basic GetMouseButton and is therefore a generic function that can be applicable to various devices.

4. Use Case

This list is not limited to what will be mentioned, but here is a list of scenario where you will find this tool useful:

1. 3D character design portfolio.
2. Packaging design of a product.
3. Showcase of a material.
4. Furniture virtual showroom.
5. Engineering schematic or diagrams.
6. Trophy showcase scene that could be a part of a game.
7. A 3D storybook where a scene is shown via 3D scene.

5. Parameter

5a. Demo Scene

On the Demo Scene which is available online, there are a few parameters on display that could be adjusted. Below are the details of these variables.

1. Camera Top Limit
Determines the top limit (maximum) position of the “look at” target of the camera.

2. Camera Bottom Limit
Determines the lowest limit (minimum) position of the “look at” target of the camera.

3. Top Rotation Threshold
Determines the position where the camera will start tilting upward, creating an effect of looking down on the object-of-interest.

4. Bottom Rotation Threshold
Determines the position where the camera will start tilting downward, creating an effect of looking up on the object-of-interest.

5. Camera Movement Speed
Determines the speed of the camera panning up and down. The higher the value, the less you need drag/swipe to pan the camera up or down.

6. Camera Rotation Speed
Determines the speed of the camera rotation (vertical rotation only). The higher the value, the less you need to drag/swipe to tilt the camera upward or downward.

5b. HoriPivot.js
The script has the following variables:

1. Rotation Speed
Determines how fast the camera will turn left or right. This is such because it is only affecting the horizontal axis of the pivot.

2. Lerp Speed
Determines how slow the deceleration will be. The higher the value the longer it will take for the rotation to go to a complete stop. Recommended value for this is 2.

5c. VertPivot.js
The script has the following variables:

1. Camera Top Limit
Determines the top limit (maximum) position of the “look at” target of the camera. On normal cases, this variable will affect the Z position of the object that this script is attached to.

2. Camera Bottom Limit
Determines the lowest limit (minimum) position of the “look at” target of the camera. On normal cases, this variable will affect the Z position of the object that this script is attached to.

3. Threshold Top
Determines the position where the camera will start tilting upward, creating an effect of looking down on the object-of-interest.

4. Threshold Bottom
Determines the position where the camera will start tilting downward, creating an effect of looking up on the object-of-interest.

5. Move Speed
Determines the speed of the camera panning up and down. It is affecting the Z position. The higher the value, the less you need drag/swipe to pan the camera up or down.

6. Rotation Speed
Determines the speed of the camera rotation (vertical rotation only). The higher the value, the less you need to drag/swipe to tilt the camera upward or downward.

7. Lerp Speed
Determines how slow the deceleration will be. The higher the value the longer it will take for the rotation to go to a complete stop. Recommended value for this is 2.

8. Minimum Tilt
Determines the lowest degree of the camera rotation (minimum downward tilting). To disable tilting downward altogether, simply put this value as -1. Do NOT enter zero to this value as it will cause the camera to be jerking.

9. Maximum Tilt
Determines the highest degree of the camera rotation (maximum upward tilting). To disable tilting upward altogether, simply put this value as 1. Do NOT enter zero to this value as it will cause the camera to be jerking.

5d. VertPivot_Sphere.js
The script has the following variables:

1. Rotation Speed
Determines the speed of the camera rotation both horizontally and vertically. The higher the value, the less you need to drag/swipe to tilt the camera.

2. Lerp Speed
Determines how slow the deceleration will be. The higher the value the longer it will take for the rotation to go to a complete stop. Recommended value for this is 2.

3. Minimum Tilt
Determines the lowest degree of the camera rotation. This variable only affects the vertical axis as the horizontal axis will always be rotate-able 360 degrees. To prevent erratic behaviour, this value should ideally be lower than the Maximum Tilt and it should never be of same value as the Maximum Tilt. Recommended Minimum Tilt is down to -90. To enable a full orbital movement, set this variable to -360 and the Maximum Tilt to 360.

4. Maximum Tilt
Determines the highest degree of the camera rotation. This variable only affects the vertical axis as the horizontal axis will always be rotate-able 360 degrees. To prevent erratic behaviour, this value should ideally be higher than the Minimum Tilt and it should never be of same value as the Minimum Tilt. Recommended Maximum Tilt is up to 90. To enable a full orbital movement, set this variable to 360 and the Minimum Tilt to -360.

5e. PinchZoom.js
This code zooms in and out by changing the field of view of the camera rather than actually moving the camera within the 3d space. This is why the camera's projection in Unity has to be set to perspective and not *orthographic*. With this in mind, the recommended value of the Max Out and Max In variable should be between 10 – 120. Do have experiments on these number to achieve desired results.

The script has the following variables:

1.Speed
Determines how fast your camera will zoom in or out. When using a mouse, the greater this variable, the lesser you have to scroll to zoom in or out. On touch devices, this will affect the sensitivity of pinch input. In principle, the larger the physical size of the device, the greater this variable should be.

To push it further: You can create functions within the PinchZoom.js to calculate the physical size of the device and adjust this variable accordingly.

2. Max Out
Determines the furthest distance of you camera to the object of interest. The greater this variable, the further the camera from object of interest and vice versa.

3. Max In

Determines the nearest distance of you camera to the object of interest. The greater this variable, the further the camera from object of interest and vice versa.

6. Future and Previous Updates

By purchasing this tool, you are entitled for free future updates. The current version is 8 and below you will find records and notes of the updates.

6a. Ver 1 to 5 Updates

- Minor bugs and performance issues.

6b. Ver 6 Updates

- After vertical rotation at the top or bottom limit, the camera will now turn back to its original position, creating a better and smoother user experience.
- There is a function to make the camera auto-rotate, as well as adjusting the speed of auto rotation.

6c. Ver 7 Updates

- Spherical Orbit is added.
- seUpSphereOrbit.unity scene is added for easy set-up.

6d. Ver 8 Updates

- Zoom function is now available.
- All scenes now have the PinchZoom.js attached to the cameras.
-

6e. Ver 8.1 Updates

- setUpSphereOrbit.unity has been modified to enable 360 degree rotation vertically. The set-up is now similar to that of setUpBasic.unity scene.
- SphereOrbit.js has been removed and replaced with vertPivot_Sphere.js, which is explored in chapter 5d.

7. Support

We are really thankful for your support by purchasing this tool, please continue supporting us by sending us feedbacks, complaints (or even curses!) to support@oluss.com and we will get back to you in 48 working hours. We do not have phone support right now.