Camera Tool

Version 1.0

http://www.forge3d.com
support@forge3d.com

Contents

Introduction

Thank you for purchasing Camera Tool!

This guide describes the features of the Camera Tool integration in Unity3D. A basic understanding of the Unity 3D engine, as well as C# programming language is assumed.

For more information please visit www.forge3d.com

If you have any questions, suggestions, comments or feature request please do not hesitate to contact us at support@forge3d.com

Brief overview

Camera Tool is a Unity extension aimed to accelerate workflow by enabling editor scene camera navigation within the game view and vice versa.

Key features:

- Navigate editor and game cameras directly inside game or scene view
- Works while the editor is playing or stopped
- Customizable activation hotkey
- Supports two game cameras with separate settings for position and rotation
- Manageable camera origins with lots of options
- Plays well with multiple scene views like “4 split” editor layout
- Supports orthographic camera mode
- Doesn’t require Unity Pro to use

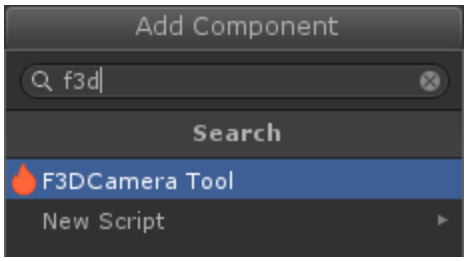
Known limitations when using Camera Tool within game view:

- Camera Tool won’t be able to process input events inside game view if the editor is paused
- Camera Tool requires at least one visible scene view to operate from inside game view window

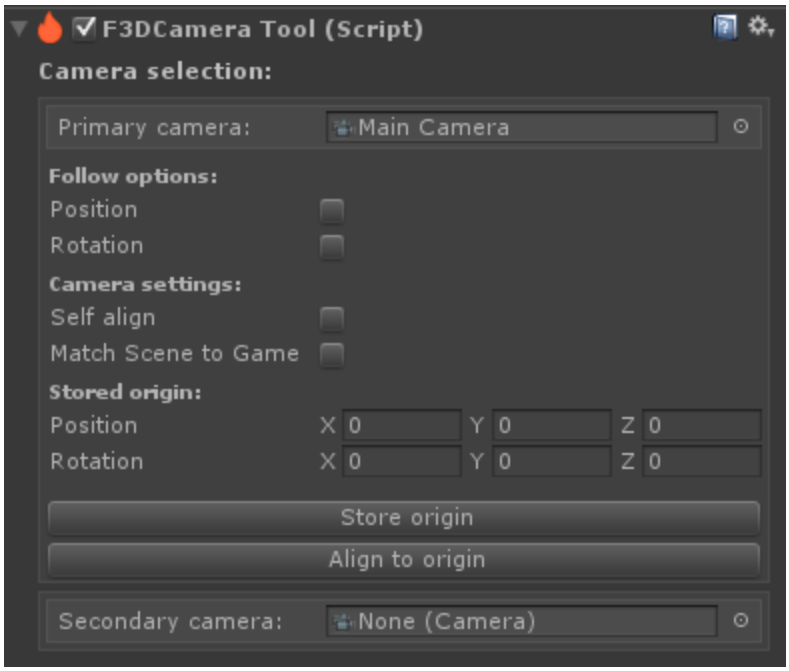
Getting started

Camera Tool ships with two required scripts: F3DCameraTool and F3DCamraToolEditor. To get started you need to add F3DCameraTool to any game object in your scene. Lets add it to the Main Camera:

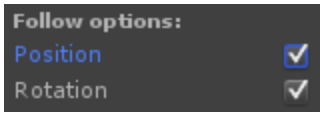
1. **Select Main Camera and click Add Component in the inspector window. Type “f3d” in the search field and select F3DCameraTool.**



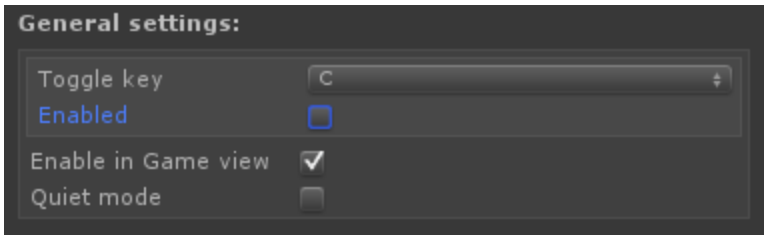
2. **Assign your game camera using the inspector selection window or do it by dragging one from the scene on top of the corresponding inspector field. If you have additional camera that is used to render you background scenery add it to the secondary slot as well.**



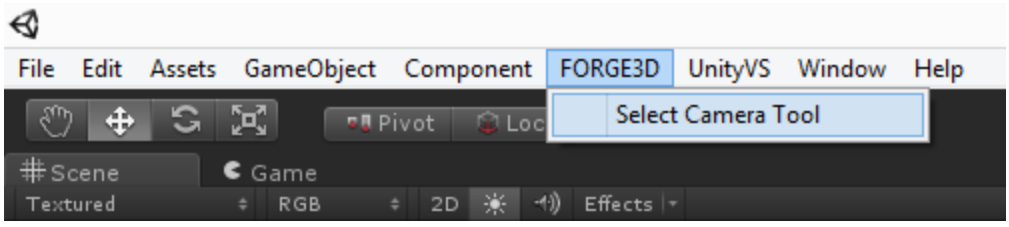
3. Lets set Follow Options. We will use position and rotation both. This means that position and rotation of the specified camera will be synchronized with the editor's scene camera when the tool is active. Note that Follow Options does not apply to the editor's scene camera and you still be able to move it inside of the game view when the tool is active.



4. Go to General Settings section and assign Toggle key using drop down list. For example if you want to assign "C" hotkey, click the list and use your "C" keyboard key to quickly cycle through all entries starting with this letter.



5. Now you can toggle Camera Tool using the hotkey you have set either do it manually using Enabled checkbox. If you like to quickly access Camera Tool there is a menu included which enables you to gain fast access to the game object the F3DCameraTool resides on.



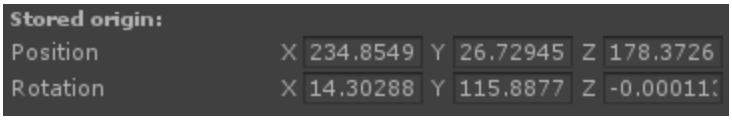
** Note that hotkey can be used inside scene or game view only.*

Using camera origin and per camera settings

Camera Tool allows you to quickly set and restore your camera position and rotation using Store origin and Align to origin button in the inspector panel.

Storing camera origin

When you click Store origin button you will instantly see the changes reflected in the Stored origin fields. You can use manual input to adjust these values at any time if needed.

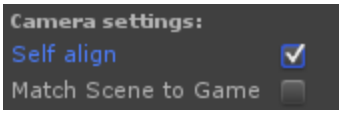


Aligning camera back to origin

To align your camera back to its origin click the Align to origin button in the inspector panel.
* *Note that this automatically disables Camera Tool.*

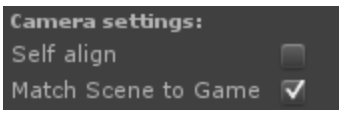
Self align

Self align option makes it possible to align your camera back to origin automatically when the Camera Tool becomes disabled.



Match Scene to Game

Setting this option will make your scene camera match the origin of your game camera each time you enable Camera Tool.
* *Note that this setting is only present in Primary camera section.*



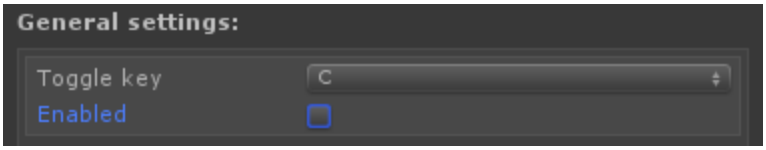
General settings

Toggle key

Allows toggling Camera Tool by using using selected hotkey.
** Note that hotkey can be used inside scene or game view only.*

Enabled

Allows manual activation / deactivation of Camera Tool.



Enable in Game View

Allows controlling Camera Tool using editor's game view.

** Note that in play mode mouse and keyboard input is rewired into the scene view while the Camera Tool is active. However it is still possible to regain focus while holding left mouse button over the game view. For example this might be essential if you'd like to use Camera Tool while moving your character using keyboard keys inside editor's game view.*

Quiet mode

Setting to quiet mode disables all on screen GUI notifications for both scene and game views.

Play mode camera fighting

Camera fighting might occur in play mode when Camera Tool and your camera script are trying to take control of the same camera. To get around this issue put a check in your code for static boolean ***F3DCameraTool.Enabled***

Lets assume that you do all your camera manipulations in LateUpdate:

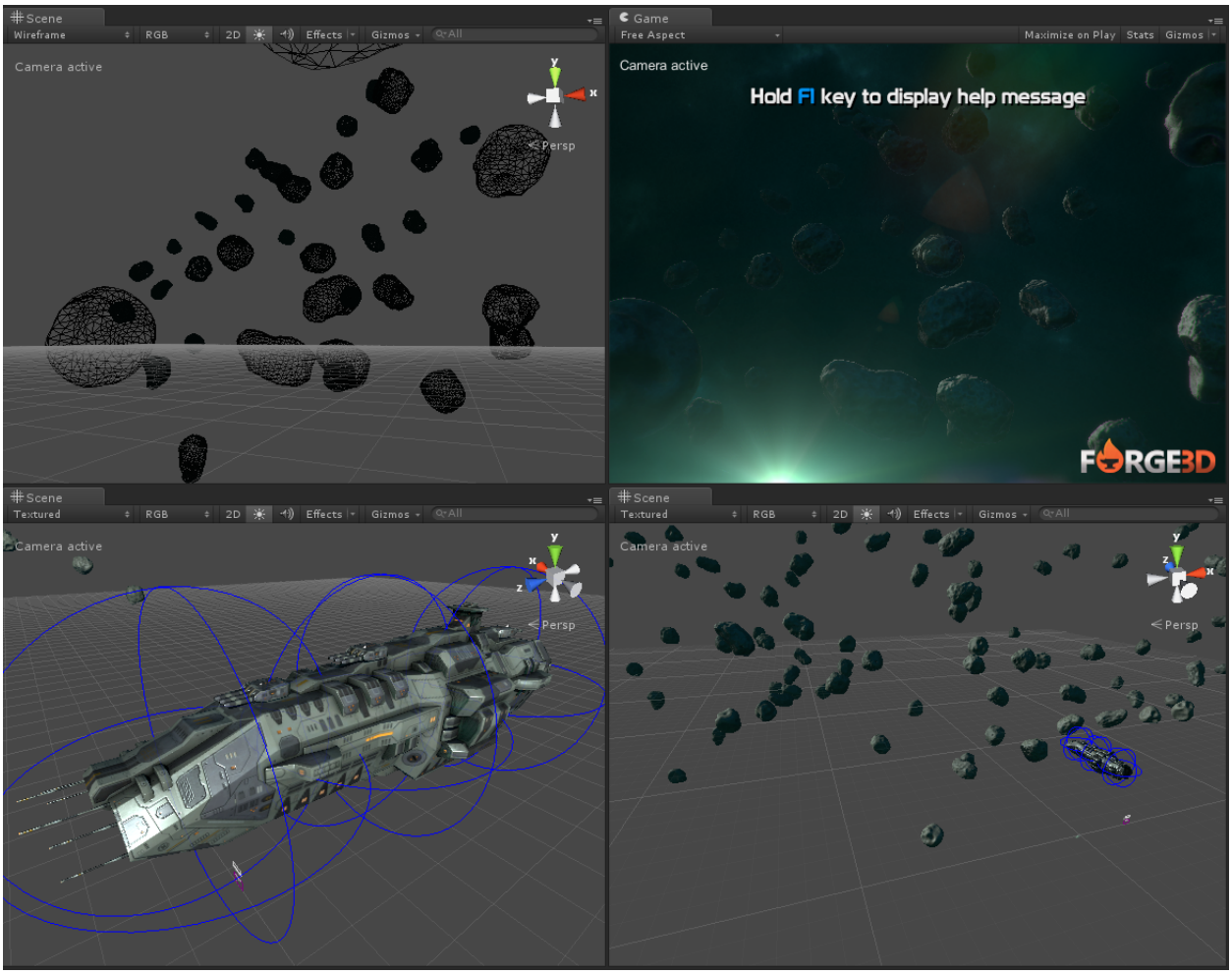
```
0 references
void LateUpdate()
{
    if (isActive)
    {
        if (Input.GetMouseButton(1))
            cameraRotation += new Vector3(FF.Input.GetMouse().x, FF.Input.GetMouse().y, 0f) * mouseSensitivity;

        cameraRotation.y = Mathf.Clamp(cameraRotation.y, -85f, 85f);

        zoomLevel += Input.GetAxis("Mouse ScrollWheel") * -zoomSpeed;
        zoomLevel = Mathf.Clamp(zoomLevel, 50f, 150f);
        offsetVector = -Vector3.forward * zoomLevel;

        offsetVector = Quaternion.AngleAxis(cameraRotation.x, Vector3.up) * offsetVector;

        Vector3 offsetSide = Vector3.Cross(offsetVector, Vector3.up).normalized;

        offsetVector = Quaternion.AngleAxis(cameraRotation.y, offsetSide) * offsetVector;

        camera.position = Vector3.Lerp(camera.position, Player.position + offsetVector, Time.deltaTime * PositionTime);
        camera.rotation = Quaternion.Slerp(camera.rotation, Quaternion.LookRotation(Player.transform.position -
                                           transform.position, Vector3.up), Time.deltaTime * LookTime);

        bgCamera.rotation = camera.rotation;
    }
}
```

Check if the Camera Tool is active and bypass any other camera code execution:

```
0 references
void LateUpdate()
{
    if (isActive && !F3DCameraTool.Enabled)
    {
```

Using Unity 4 Split interface layout

Camera Tool plays nicely with 4 split editor layout or multiple scene views.



There are few tips on using Camera Tool in this mode:

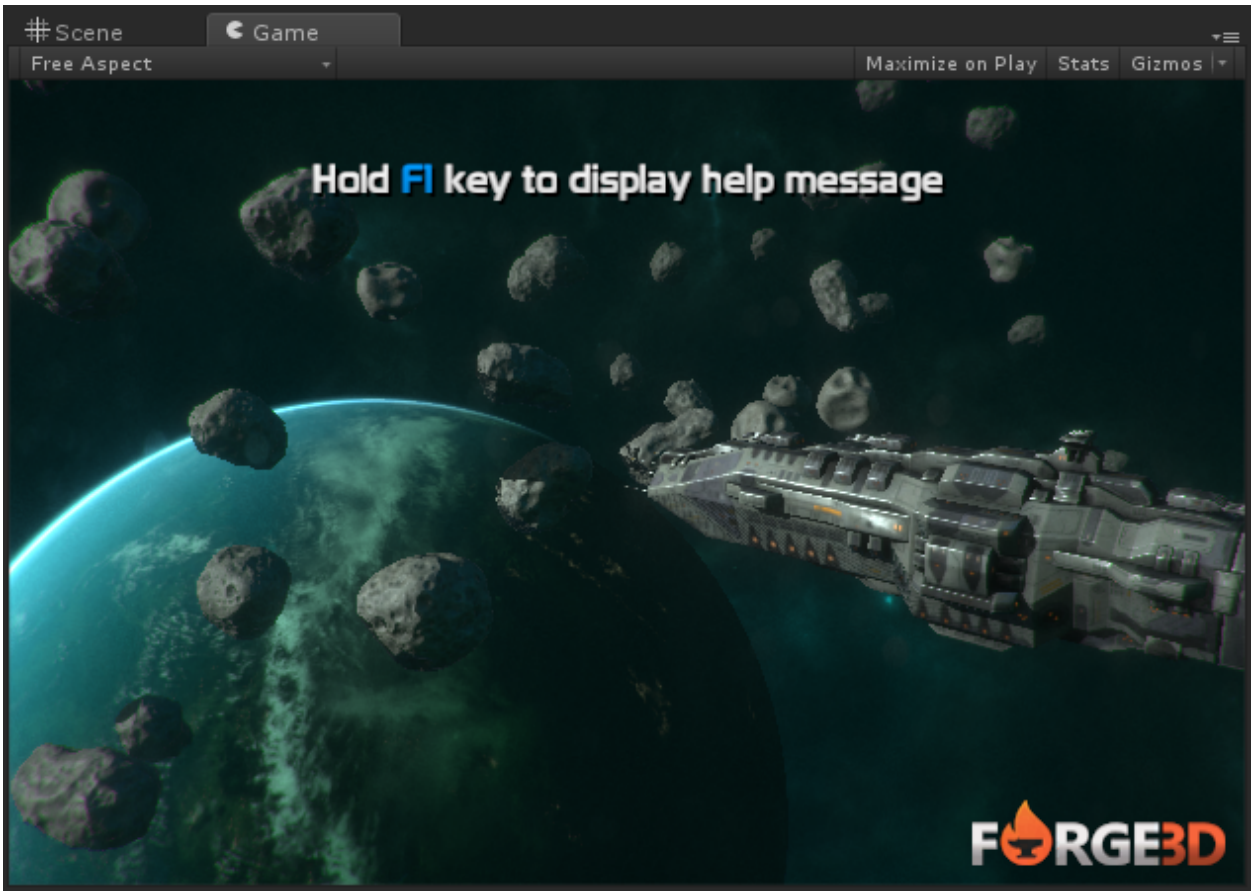
- Remove one of the scene views and place your game view in its place like shown in the image above. Or if you have a second monitor you might just skip that and use it for the maximized game view.
- Select one of the scene views to make it active. This will also reposition your game camera to match the active scene view.
- Moving the camera using game view will synchronize your last active scene view to it.
- You might use the Hierarchy window to quickly move your scene/game camera by double clicking one of the objects.

Troubleshooting

This section describes known limitations you might encounter while using Camera Tool and possible suggested workarounds.

Game view losing focus when Camera Tool is active

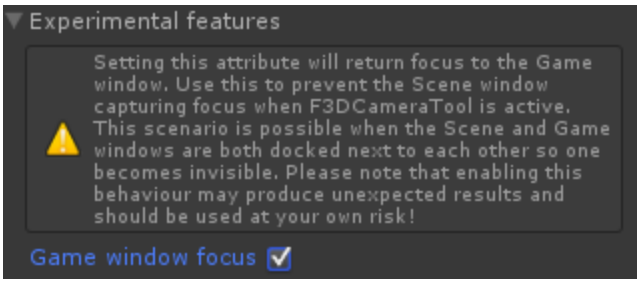
Lets describe the nature of this issue at first. Camera Tool is using active game view to capture OnGUI events that are rewired into the last active scene view. This goes well up until the point when you hide the scene view by another window docked on the same tab:



If it were still sending events to the hidden scene view the editor would have become unresponsive. However there is a safety guard in the code that returns focus to a scene view before sending any events and it is the main reason why this issue occurs.

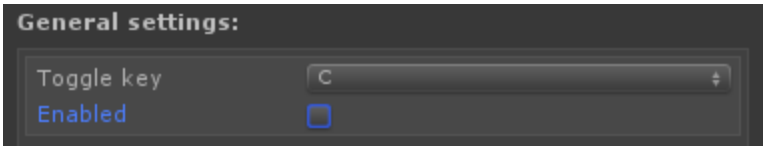
There are two ways to deal with this:

- If you're comfortable keeping scene view open all the time and that's completely fine.
- Try using an experimental option that allows it to restore focus back to the game view.



Camera Tool stops working when the scene view is closed

In order to work the Camera Tool requires at least one scene view window present. If you closed the scene view and get message saying "No active Scene window present or visible" make sure you select Camera Tool instance and manually disable it by unchecking "Enabled" checkbox in the General settings section in the inspector. This might occur if your scene view were hidden or closed at the time you pressed editor's play button. Try reopening scene window and hitting toggle hotkey again.



Game view does not receive input when editor is paused

The reason for this is mainly because Unity doesn't send input events into OnGUI when its paused. You can still use editor's scene view to control your camera.

Game camera flickering, becomes unsteady or behaves unnaturally in play mode

Make sure nothing else is modifying game camera position/rotation while the Camera Tool is active by checking for static boolean ***F3DCameraTool.Enabled.*** See Play mode camera fighting section of this manual for detailed instructions. Also check if the Game window focus checkbox is disabled as this might introduce similar issues as well.