



**Users Guide Version 1.1**

**Tutorials Available**

**[Click Here!](#)**

**Or**

**Go to :**

**[http://www.youtube.com/playlist?list=PLK4BWycdGcCX0d6bDSlyCIDykQx2h0Qog&feature=view\\_all](http://www.youtube.com/playlist?list=PLK4BWycdGcCX0d6bDSlyCIDykQx2h0Qog&feature=view_all)**



# PathCam v1.1 Users Guide

## Types of Path Controllers:

- Standard Path Controller
  - o Use this controller to move the camera along waypoints throughout your scene.
  - o You can make the camera look directly at your waypoint or you can setup a "Look At Object"
- Tracked Path Controller
  - o Use this controller to move the camera along a linear path
  - o You can setup "Look At Objects", or simply just make the camera move along the path given.

## Terms to know:

- Path Controller
  - o This is a game object that has the PathController.cs or PathControllerTracked.cs script attached to it.
- Waypoint
  - o This is a point in the scene at which you want the camera to move to
- Look At Object (LAO)
  - o If you USE Look at Object, the camera will rotate to look at the object in the scene
  - o If you DO NOT USE Look At Object, the camera will default to looking at the next waypoint
  - o Setup the LAO, by adding an object to the "Look At Object" property in the inspector
  - o NOTE : If you check the box to use the LAO and do not set the LAO, the camera will default to look at the next waypoint
  - o NOTE : You can set the LAO to the "LookAtObject" prefab OR anything else in the scene you want the camera to look at.
- Constant Speed
  - o This bool will allow you to set which type of movement you want to use
  - o If you USE Constant Speed, the camera will continuously move to the next waypoint at the set speed.
  - o If you DO NOT USE constant speed, the camera will slow down as it gets closer and closer to the next waypoint
- Approach Speed (Value of 0 to 25)
  - o This is the speed at which the camera will move along the path
- Rotation Speed (Value of 0 to 25)
  - o This is the speed at which the camera will rotate to look at the LAO or the next waypoint
- Transition Distance (Value of 0 to 100)
  - o While the camera is moving along the path, it will be decrementing down to 0 based on the speed to the next waypoint

- Transition Distance is the distance at which you want the camera to move on to the next waypoint
- EX : The camera is moving towards a waypoint (W1). While it is moving it is going from 15 – 0 in distance. The transition distance is set to 5. When the Camera reaches a distance of 5 from the W1, it will move on to the next waypoint, W2.
- Inbetween Distance
  - This is the distance between the previous and next waypoint.
  - This should only show in the normal Path Controller
- Travel Distance
  - This is the distance between a starting and ending waypoint in the Tracked Path Controller
  - We need to know this distance so we can setup the Starting Distance of our first look at object
  - The first LAO's Starting Distance > Travel Distance, to insure we are looking at that object from start to the ending distance

## Setting up a Normal Path Controller

- 1) From the prefabs folder, add the object called "PathController"
- 2) Click on the PathController and in the Inspector, set the "Main Camera" Camera object to the camera you want to use
- 3) Then expand the Waypoints and set the size of the waypoints array
- 4) Now, go to the Prefabs folder again, and bring in "WaypointStart"
  - a. The starting waypoint will always be the 1 waypoint in the waypoint array in the path controller
  - b. This will be the spot the camera start, therefore you do not need any settings for speed or distance
- 5) Place the starting waypoint in an area of your discretion.
- 6) Go to the prefabs folder again and drag in the "Waypoint" object.
  - a. It should be labeled as a "Middle Position"
  - b. NOTE: If you want to setup the "Look At Object", please read the above "Terms to Know"
  - c. NOTE :See "Terms to Know" (Above) for the rest of the properties descriptions
- 7) Place the middle waypoints in an area of your discretion
- 8) Go to the prefabs folder again and drag in the "WaypointEnd" object
  - a. It should be labeled as "Ending Position"
  - b. This will be the final waypoint on your path.
  - c. NOTE : The final waypoint can be any waypoint. I made this so it would show as a RED ball in the scene view, so you know where the path ends.
- 9) Place the ending waypoint in an area of your discretion
- 10) Click on the Path Controller object in the scene
- 11) Drag and drop the Waypoints to the corresponding Waypoint positions in the array.

- a. While you do this, you should notice 2 things:
  - i. A line will be drawn in the scene view from waypoint to waypoint
  - ii. The in-between distance will be calculated and shown in the inspector of the path controller.
- 12) In the PathController.cs script, I have added a button in the OnGUI() function to call the public Animate() function.
  - a. If you would like to, you can remove this part of the script and call the Animate() function from another script in the scene.
  - b. Use the `gameObject.GetComponent<PathController>().Animate();` line of code to call this function

### Setting up a Tracked Path Controller

- 1) From the prefabs folder, select and drag in the "PathControllerTracked" prefab object
- 2) Click on the PathControllerTracked object and you will see the Main Camera, Starting Waypoint, Ending Waypoint and the Look at Object Array
- 3) Drag the camera you wish to use into the Main Camera setting on the Tracked Path Controller
- 4) Go to the prefabs folder again and drag in the "WaypointStart" prefab
- 5) Place the starting waypoint in an area of your discretion in the scene
- 6) Go to the prefabs folder again and drag in the "WaypointEnd" prefab
- 7) Place the ending waypoint in an area of your discretion in the scene
- 8) Click on the Tracked Path Controller object
- 9) Drag the Starting and End Waypoints into their slots on the Tracked Path Controller
- 10) Now you should see 2 things happen:
  - a. A line will now be drawn in the scene view from the start to the ending waypoints
  - b. The travel distance parameter is now filled.
  - c. See "Terms to know" for Travel Distance information
- 11) Select to "Use Constant Speed" or not
  - a. See "Terms to Know for Constant Speed information"
- 12) Select to "Use Look At Objects" or not
  - a. NOTE : If you DO NOT use LAOs or your LAOs array size = 0
    - i. The camera will move from waypoint to waypoint without rotating to look at something
- 13) Go to the prefabs folder, and drag in the "LookAtObjectTracked" prefab
  - a. Add as many as you want
- 14) Click on the LAO to view its properties in the inspector
  - a. Set the Starting Distance
    - i. NOTE : Make sure the first LAOs starting distance is GREATER THAN the Travel Distance shown in the TrackedPathControllers Inspector Properties. This is to insure you are looking at this object right from the beginning

- b. Set the Ending Distance
    - i. This will be the distance along the path at which the camera will stop looking at the LAO
  - c. Set the Rotation Speed
  - d. Things to note when setting up multiple LAOs
    - i. Make sure the FIRST LAOs starting distance is GREATER THAN the Travel Distance between the 2 waypoints
    - ii. Make sure the Ending Distance is EQUAL TO the Starting Distance of the next LAO
- 15) Click on the Tracked Path Controller object in the hierarchy
- 16) Set the size of the LAO array in the inspector equal to the amount of LAOs you have in your scene
- 17) Drag and drop the LAOs into their corresponding positions in the LAOs array on the Tracked Path Controller
- 18) Now you will notice 2 things happen:
- a. A new checkbox will show between 2 LAOs in the array
    - i. This check box is to insure that the previous LAOs Ending Distance is EQUAL to the next LAOs Starting Distance
    - ii. If they are a match, the box will be checked
  - b. If the Transition Match is checked, a box will be shown on the path line in the scene view.
    - i. This is done so you know exactly where the camera will switch off the one look at object and move to the next.
    - ii. This can be very useful in the scene view
- 19) Once you have finished setting up the LAOs, you can run the game and click the button to move along the path
- 20) In the TrackedPathController.cs script, I have added a button in the OnGUI() function to call the public Animate() function.
- a. If you would like to, you can remove this part of the script and call the Animate() function from another script in the scene.
  - b. Use the `gameObject.GetComponent<PathController>().Animate();` line of code to call this function