EZ HOVER DOCUMENTATION

Table of Contents

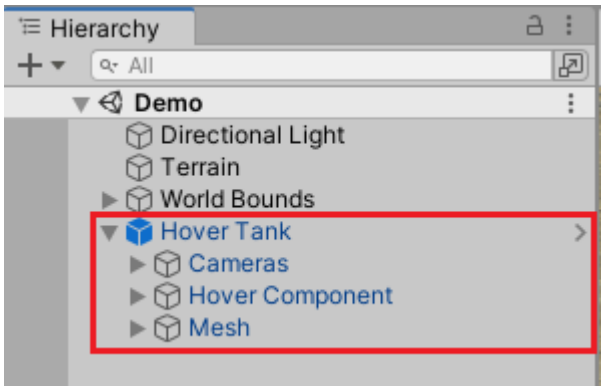
1. Hover Object Structure
2. Components
    a. Hover Grid
    b. Hover Movement
    c. Hover Look
    d. Hover Point
3. Troubleshooting
    a. Why does my hover object fall to the ground and not hover?
    b. Why does the movement of my hover object drift so much?
    c. Why does my hover object spin out of control so much?
4. Further help

Hover Object Structure

It is recommended that all game objects that need hover physics follow this structure:

- Parent object with Rigidbody component
    - Child object that holds cameras for 1st/3rd person views (optional)
    - Child object that contains hover scripts
      It is recommended to put the Hover Grid script on a separate child object so that it can be repositioned.
    - Child object that contains mesh and colliders

This structure is reflected in the demo scene hierarchy for the 'Hover Tank' prefab:



Tip: use the 'Hover Tank' prefab as a template. This is located in EZ Hover Studios/EZ Hover/Demo/Prefabs.

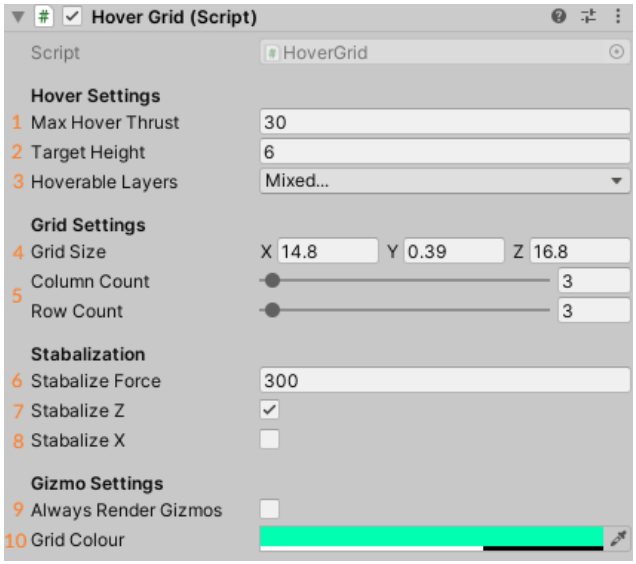
Components

This section covers all of the configurable settings on the Hover Grid, Hover Movement, and Hover Look components.

Hover Grid

The hover grid is responsible for applying hover forces to the object. Essentially what the hover grid does is it spreads a group of raycasts over a wide area in a grid-like pattern to determine the

distances between the grid and the ground below. The closest detected distance is then used to calculate how much hover force is applied to the object. For example, if the object is very close to the ground the distance will be small resulting in a larger hover force to be applied.



1. **Max hover thrust**
   Determines how much force can be used to keep the object hovering above the ground.
   A high value will result in a more sudden upward motion.
   If the value is too low then there may not be enough force to lift the object above the ground.

2. **Target height**
   Determines how far the object should hover above the detected ground below.

3. **Hoverable Layers**
   Determines what layers will be included when detecting obstacles/ground to hover over.

4. **Grid size**
   How much area should the grid cover?
   It is recommended that the grid covers the size of the mesh colliders.

5. **Column count & Row count**
   Both determine how many rays are cast from the grid.
   It is recommended that both are adjusted so that the size of each grid cell remains roughly the same.
   A high column and row count will make the grid more sensitive to small bumps and irregularities on the ground below.

6. **Stabilization Force**
   The amount of force applied to keep the object stabilized on the X/Z axis.

7. **Stabilize X**
   Should the game object stabilize itself to be at 0 degrees rotation on the X axis?

It is recommended that this remains switched off if the 'Look' script is in use so that the object can freely look up and down.

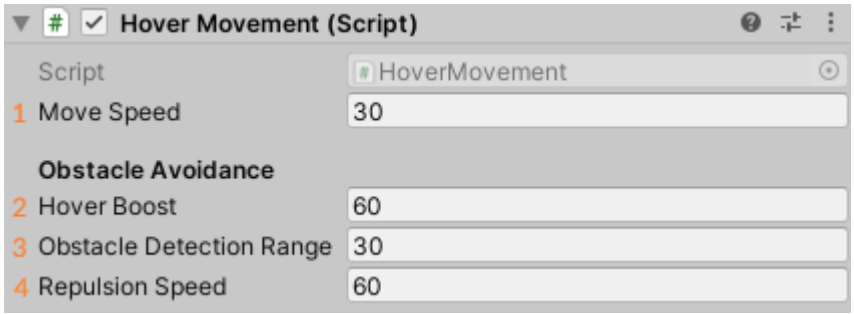
8. **Stabilize Z**
   Should the game object stabilize itself to be at 0 degrees rotation on the Z axis?

9. **Always Render Gizmos**
   Should the Hover Grid gizmos render in the scene even if the gameobject is deselected?

10. **Grid colour**
    It is used to customize the colour of the grid gizmo rendered in the scene.

Hover Movement

The hover movement component will enable the hovering object to move horizontally using keyboard WASD input. In addition to this, an extra upward force will be applied when incoming obstacles or rises in the terrain are detected to help avoid collisions.



1. **Move Speed**
   Determines how fast the object will move horizontally.

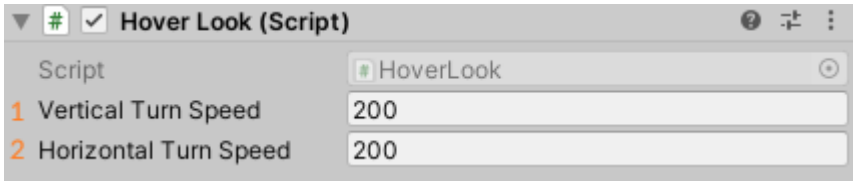
2. **Hover Boost**
   Determines how much extra hover force will be applied to help avoid collisions with incoming obstacles or rises in the terrain.

3. **Obstacle Detection Range**
   Determines how far incoming obstacles or rises in the terrain will be detected.
   A higher detection range means that the Hover Movement component will anticipate collisions sooner.

4. **Repulsion Speed**
   Determines the magnitude of push back from a nearby obstacle or rise in terrain which is in the way of movement.
   This gives more time for the object to rise with the hover boost to help avoid collision.

Hover Look

And finally, the hover look component enables the hover object to look around left/right/up/down by moving the mouse. This works best for a player character with a 1st/3rd person camera.



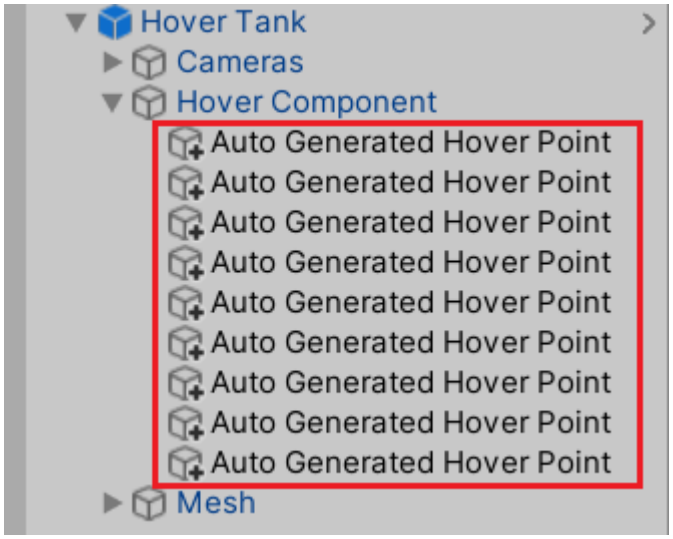
1. **Vertical Turn Speed**
   How fast should the hover object turn up and down?

2. **Horizontal Turn Speed**
   How fast should the hover object turn left and right?

Hover Point

The hover point component is placed on the automatically generated hover point game objects. In the image below, each 'Auto Generated Hover Point' is used at as reference point on the grid so that a ray is cast from the centre of each grid cell.

Note: You don't need to worry about using the Hover Point scripts since they are only useful within the context of the Hover Grid component.

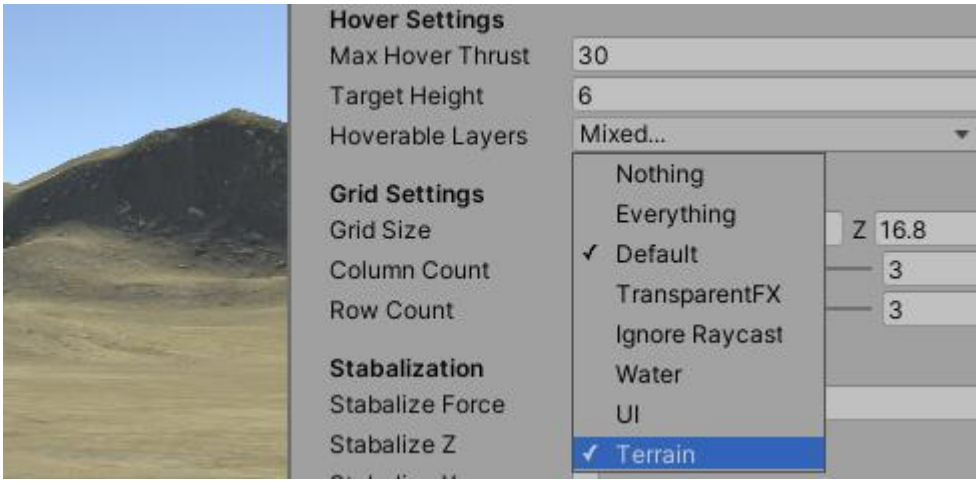


Troubleshooting

This section will help you solve common problems you may have when trying to use the asset.

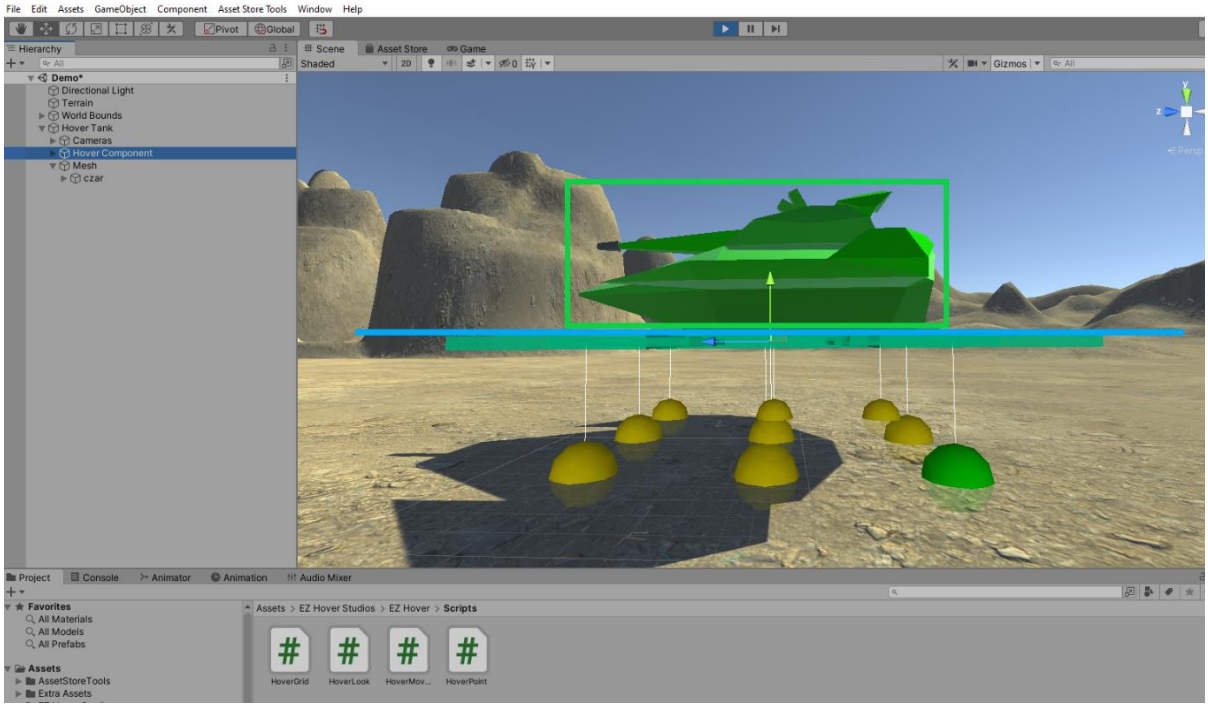
Why does my hover object fall to the ground and not hover?

1. Make sure the object has a Rigidbody and Hover Grid component
2. Increase the max hover thrust on the Hover Grid – it may be too low to support the weight of the object.
3. Make sure the target height on the hover grid is set to a value above 0.
4. Double check that the ground or obstacles below have layers that are included in the 'Hoverable Layers' dropdown of the Hover Grid.



In the demo scene, the hover grid on the hover tank specifies both default and terrain as hoverable layers. The terrain layer needs to be included because the terrain game object (the ground below) has this layer.

5. The Hover Grid needs to be positioned roughly at the bottom of the hover object's collider. In the image below, the blue hover grid is roughly aligned with the bottom of the hover tank. The hover grid may not be detecting the ground because it is positioned too low, try moving it up a little.
   Don't worry if the hover grid cuts into any of the object's colliders because the raycasts will ignore them.



Note: It is recommended to put the Hover Grid script on a separate child object so that it can be repositioned.

Why does the movement of my hover object drift so much?

This is likely caused by the Drag value on the object's Rigidbody. By default, the drag is set to 0 which means the hover object will continue to drift horizontally.

All you need to do is increase the drag value until you get the desired results.

The Hover Tank in the demo scene has a drag value of 1 which means it slows down at a reasonable rate and accelerates quickly. A high drag and high move speed will result in more responsive movement controls.

If you do increase the drag value, just make sure to increase the move speed on the Hover Movement component otherwise the hover object will move very slowly.

Why does my hover object spin out of control so much?

The angular drag value on the Rigidbody component is too low. By default, it is set to a very low value which means that any torque force applied to the hover object will dissipate very slowly.

Simply increase the angular drag until the turn forces die down quick enough. If you are using the Hover Look component, be sure to increase the vertical and horizontal turn speeds.

Further help

Still can't solve the problem?

Don't hesitate to join the Discord server to ask for help.

Post any questions or troubles you're having with the asset, and we'll get back to you!