Glider Flying System Documentation

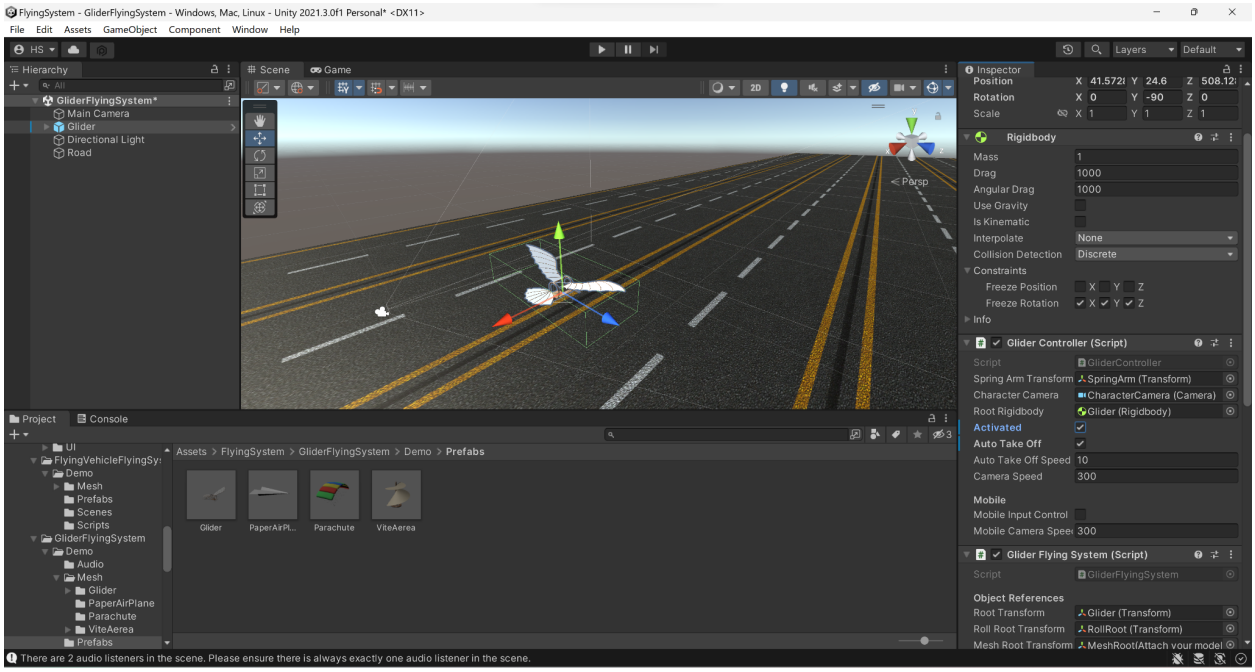
Description

This system is used for simulating flying behaviors of parachute and glider. This type of aircraft doesn't have power. Diving and gliding are supported.

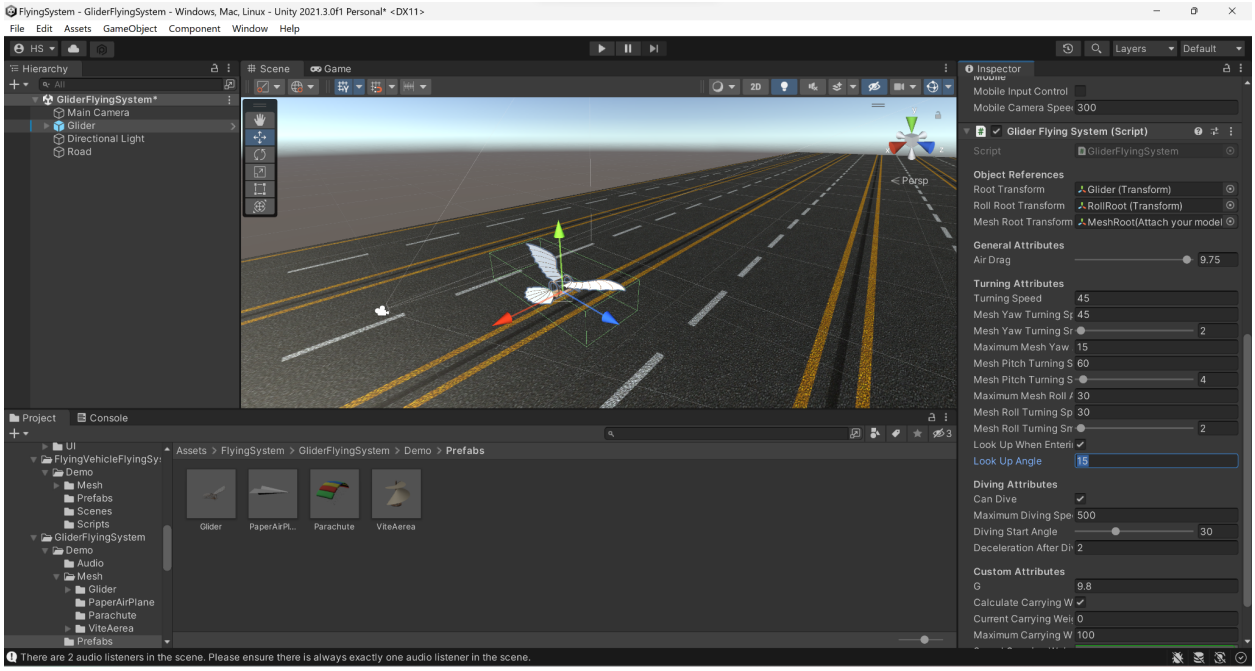
How to dive and glide: When controlling the aircraft, press E to point down until reaching some degree, then it will start accelerating and diving. Press Q to point up before the aircraft touches the ground, it will gain horizontal speed and glide for a certain distance.

Setup

1. Import the plugin.
2. Install "Post Processing" using Package Manager if it is not installed.
3. Create a new scene and make a simple terrain, navigate to Assets\FlyingSystem\GliderFlyingSystem\Demo\Prefabs folder, drag the Glider prefab into the scene, select "Activated" and "AutoTakeOff" for the GliderController script on the Inspector panel, then adjust the "AutoTakeOffSpeed".



4. Depending on the need, the attributes and methods in Glider Flying System component can be adjusted/executed.



Explanation

Setting the pitch, yaw, roll together for the same GameObject will cause gimbal lock problem(https://en.wikipedia.org/wiki/Gimbal_lock). To solve this problem, two GameObjects(RollRoot, MeshRoot) are used to separate them.

The prefab has the following hierarchy:

- Glider
    - SpringArm
        - CharacterCamera
    - RollRoot
        - MeshRoot
            - Glider(3d model)

1. Glider: It only acts as the root, and doesn't represent the actual collision for the flyer. Rigidbody, C# scripts and audio source are attached to this GameObject.
2. SpringArm: For controlling the third-person camera.
3. CharacterCamera: The actual camera.
4. RollRoot: The roll value of the flyer will be set here.
5. MeshRoot: The pitch and yaw values will be set here.
6. Glider(3d model): The actual flyer mesh, can be customized. The mesh should be attached to MeshRoot, since the flying rotation will be applied to its parent, not directly to the mesh.

Essential Object References

rootTransform	The Transform component of the root object.
rollRootTransform	The Transform component of the roll root object. There must be a RollRoot object above the MeshRoot object in the Hierarchy. See the examples of “Parachute.prefab” and “Glider.prefab” in Assets\FlyingSystem\Demo\Prefabs\Controllers\AirTransportations folder.
meshRootTransform	The Transform component of the mesh root object.

Adjustable Attributes

General Attributes	
airDrag	It only affects the vertical direction of gliding, the greater the value, the smaller vertical acceleration will be.
Turning Attributes	
turningSpeed	It defines how fast the aircraft turns.
meshYawTurningSpeed	The speed of yaw(horizontal) turning.
meshYawTurningSmoothingFactor	The interpolated parameter for meshYawTurningSpeed. The smaller the value, the smoother and slower the turning will be.
maximumMeshYawAngle	The maximum yaw angle of the mesh.
meshPitchTurningSpeed	The speed of pitch(vertical) turning.
meshPitchTurningSmoothingFactor	The interpolated parameter for meshPitchTurningSpeed. The smaller the value, the smoother and slower the turning will be.
maximumMeshRollAngle	The maximum roll angle of the mesh.
meshRollTurningSpeed	The speed of roll turning.
meshRollTurningSmoothingFactor	The interpolated parameter for meshRollTurningSpeed. The smaller the value, the smoother and slower the turning will be.
lookUpWhenEnteringAirflow	Whether the aircraft will point up when entering the airflow.
lookUpAngle	The additional angle for pointing up.

Diving Attributes	
canDive	Whether the aircraft can dive or not.
maximumDivingSpeed	The aircraft can not dive faster than this value.
divingStartAngle	The minimum angle to start the diving. If it is 30, it will start diving when the camera looks down 30 degrees below the horizon.
decelerationAfterDiving	It defines how fast the speed decreases when finishing diving. The smaller the value, the longer distance it can glide.
Custom Attributes	
g	The value of gravity acceleration, only affects the diving.
calculateCarryingWeight	Whether calculate the carrying weight of the aircraft or not.
currentCarryingWeight	The current carrying weight of the aircraft.
maximumCarryingWeight	The maximum carrying weight of the aircraft.
speedCarryingWeightRatioAnimationCurve	An editable curve of the relation between speed and carrying weight. Normally the higher carrying weight, the slower speed.
Other Public Attributes	
enabledFlyingLogic	Whether enable/disable the flying logic.
inAir	Whether it is in the air or not.
inAirflow	Whether it is inside the airflow or not.
flyingDirection	The flying direction, in Vector3.

originalFlyingSpeed	The speed when taking off.
flyingSpeed	The flying speed.
flyingVelocity	The flying Velocity, in Vector3.
weightPercentage	The percentage of carrying weight, equals 1 when fully carried.

Methods

TakeOff(float takeOffSpeed)	Make the aircraft take off.
Land()	Make the aircraft land.
AddYawInput(float value)	Make the aircraft turn left/right.
StopYawInput	Make the aircraft stop turning left/right.
AddPitchInput(float value)	Make the aircraft point up/down.
AddRollInput(float value)	Make the aircraft roll left/right.
AddAirflowForce(float intensity, float acceleration, float fadeOutAcceleration)	Add an airflow force to the aircraft.
EndAirflowForce()	Stop the airflow force to the aircraft.
AddWeight(float increaseValue)	Increase the current carrying weight of the aircraft, the increaseValue can be positive or negative.

Universal Render Pipeline (URP) & High Definition Render Pipeline (HDRP)

The materials can be converted by following this tutorial:
https://www.youtube.com/watch?v=aJ1OpirisGM

References

The 3D models(Parachute.fbx, Glider.fbx, ViteAerea.fbx, PaperAirPlane.fbx) are the original creation by the developer of this project.

Wood texture: https://www.pexels.com/photo/brown-wooden-surface-129733/

Cloth & white cloth textures: https://www.publicdomainpictures.net/en/view-image.php?image=115236&picture=white-texture-background

Metal texture: https://www.behance.net/gallery/60064327/20-Seamless-Brushed-Metal-Background-Textures-DOWNLOAD