RealityScan to Unreal Engine

Workflow Tutorial

Introduction

In this workflow, we show how real-world photos are turned into a usable 3D model and imported into Unreal Engine. The process uses RealityScan for scanning and Unreal Engine for real-time use.

1. Importing and Aligning Images

We start by importing our photos into RealityScan.

- Set the workspace to a 1 by 2 view for better overview
- Drag the folder containing the images or video into RealityScan
- Click Align Images or press F6
- After alignment, a point cloud and camera positions should appear

At this point, the project should be saved to prevent data loss.

2. Scaling the Model to Real-World Size

To make sure the model has the correct size for Unreal Engine, we define a real-world scale.

- Enable Control Points in the 3D Scene Tools or press F3
- Place two control points on locations with a known distance
- Assign each control point to several clear images
- Use Define Distance and select both points in the 3D view
- Enter the real distance in meters
- Update the alignment to apply the scale

Errors can be ignored here. If needed, restarting RealityScan resolves update issues.

3. Setting Orientation and Ground Plane

Next, we align the model correctly in the scene.

- Activate Set Ground Plane
- Use top, side, or front views to align the model
- Rotate and move the scan so it sits flat on the grid
- Center the model before exiting the tool

This ensures the model is not tilted or floating later.

4. Defining the Reconstruction Region

To control what part of the scan becomes geometry:

- Set the reconstruction region automatically
- Resize the box tightly around the object

Only the content inside this box will be reconstructed, improving quality and performance.

5. Reconstructing the 3D Model

Now we generate the mesh.

- Go to the Mesh Model tab
- Choose Normal Detail
- Wait for the reconstruction to complete

This creates the actual 3D geometry.

6. Cleaning the Mesh (Optional)

Unwanted floating geometry can be removed.

- Select the largest connected component
- Invert the selection
- Delete unnecessary triangles

RealityScan keeps previous versions, so changes are reversible.

7. Texturing the Model

Textures are generated from the original photos.

- Open mesh color and texture settings
- Set texture resolution based on project needs
- For best quality, use Fixed Texel Size with optimal settings
- Unwrap and generate textures

This step defines the visual quality of the model.

8. Optimizing for Real-Time Use (Optional)

To reduce performance cost:

- Use the **Simplify** tool
- Set a target triangle count
- Enable color and normal reprojection
- Generate a simplified version

This is recommended for game engines.

9. Exporting to Unreal Engine

Finally, the model is exported.

- Select the correct model version
- Export as FBX
- Enable vertex normals
- Disable vertex colors if textures are used
- Use the Unreal Engine export preset

The model is now ready to be imported into Unreal Engine.

10. Unreal Engine Check

Inside Unreal Engine:

- Import the FBX file
- Check scale and orientation
- Place the model on the grid
- Verify textures and materials

If everything looks correct, the workflow is complete.

Closing

This workflow shows how real-world objects can be transformed into clean, game-ready 3D assets using RealityScan and Unreal Engine.