

Raytracing

Reference

Panel:

Render ▸ Raytracing

The ray-tracing pipeline goal is to increase the accuracy of surface indirect lighting. This is done by generating ray from each [BSDF](#) and finding their intersection with the scene individually.

When disabled, it is replaced by a faster pipeline that uses pre-filtered light-probes. This fallback mode offers a more visually stable and optimized alternative when visual fidelity is not the primary goal.

See also

[Limitations](#).

Method

Determine the tracing method used to find scene-ray intersections and indirect lighting.

Light Probe:

Use light-probe spheres and planes to find scene intersection. This option has the lowest tracing cost but relies on manually placed light-probes.

Screen-Trace:

Trace ray against the screen depth buffer. Fallback to light-probes if ray exits the view.

Resolution

Resolution at which the ray-tracing is performed. Lower options will be faster and use less memory but will produce blurrier results.

Screen Tracing

These settings control the behavior of the screen space ray-tracing. They are only visible if *Screen-Trace* is the active tracing *Method*.

Precision

Higher values increase precision of the screen space ray-tracing but lower the maximum trace distance. Increased precision also increases performance cost.

Thickness

How thick to consider the pixels of the depth buffer during the tracing. Higher values will stretch the reflections and add flickering. Lower values may make the ray miss surfaces.

Denoising

Denoising can be enabled to reduce the amount of noise from the raw ray-traced output. This can help image stability but will also over-blur the final ray-traced output.

Spatial Reuse

Reuse the rays from neighbor pixels. Can introduce some light leaks across surfaces.

Temporal Accumulation

Accumulate samples by re-projecting the last ray tracing results. This removes [Fireflies](#) but also introduces color bias. Useful for viewport temporal stability or making renders converge faster.

Bilateral Filter

Blur the resolved ray-traced output using a bilateral filter.

Fast GI Approximation

Fast GI Approximation is a fallback to the ray-tracing pipeline for [BSDF](#) with high roughness. It produces a less noisy output and captures bounce light more efficiently than individually traced rays.

This is currently implemented as a screen space effect and will inherit all associated [limitations](#).

Threshold

Maximum roughness a [BSDF](#) can have to use ray-tracing. BSDFs with higher roughness will progressively use the *Fast GI Approximation*. A value of 1 will raytrace every surfaces and disable the Fast GI.

Method

Determine the method used to compute the fast GI approximation.

Ambient Occlusion:

Use scene intersections to shadow the distant lighting from light-probes. This is the fastest option.

Global Illumination:

Compute global illumination taking into account light bouncing off surrounding objects.

Resolution

Resolution at which the fast GI is computed. Lower options will be faster and use less memory but will produce blurrier results.

Rays

Number of GI rays per pixel at the specified *Resolution*. Higher values will reduce noise.

Steps

Number of screen samples per GI ray. Higher values will reduce the noise amount and increase the quality.

Tip

With a higher step count, there is less chance to miss other surfaces that could reflect or block the light. This means that the Fast GI *Thickness* parameters can be tweaked to lower values without losing too much light bounce energy.

Precision

Higher values increase the precision of the scene intersections with the GI rays. Increased precision also increases performance cost.

Distance

If non-zero, the maximum distance at which other surfaces will contribute to the fast GI approximation.

Thickness Near

Geometric thickness of the surfaces when computing fast GI and ambient occlusion. Reduces light leaking and missing contact occlusion. The effectiveness decreases proportionally to the distance from the shading point, following the inverse square law.

Far

Angular thickness of the surfaces when computing fast GI and ambient occlusion. Reduces energy loss and missing occlusion of far geometry. Higher values will make the very thin objects block or reflect too much light.

Bias

Bias the shading normal to reduce self intersection artifacts.