

IPR Tab

This page provides details on the settings found on the IPR tab of the V-Ray Renderer parameters.

Overview

The IPR tab in the Render Settings window allows you to control the settings of the V-Ray IPR Renderer.

To use the GPU engine, a video card that supports the following is required:

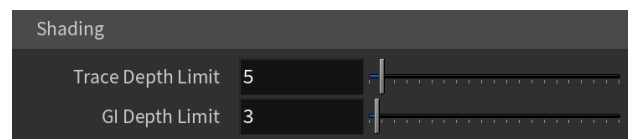
- CUDA 2.0 or higher.
- The CUDA engine for GPU works on NVIDIA cards with the Maxwell, Pascal, Volta, Turing, or Ampere architectures.

We have received reports of geometry occasionally disappearing during IPR. After further investigation, it appears a specific environment variable might be causing this unexpected behavior. If you experience IPR issues related to geometry suddenly disappearing during rendering, please try setting the "HOUDINI_ACCESS_METHOD" environment variable to "3".

Shading

Trace Depth Limit – Represents the maximum number of bounces that are computed for reflections and refractions. The individual material reflection/refraction depth settings are still considered, as long as they don't exceed the value specified here.

GI Depth Limit – The number of bounces for indirect illumination. Other GI settings (e.g. whether GI is enabled or disabled) are taken from the production V-Ray renderer.



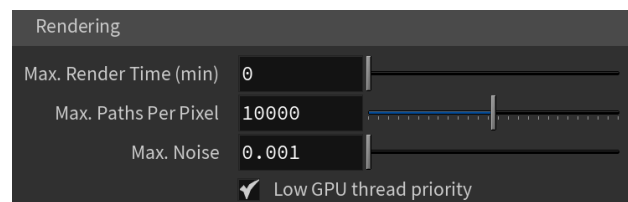
Rendering

Max. Render Time (min) – Since V-Ray IPR uses progressive path tracing, it would never stop rendering unless specified otherwise. This option specifies a maximum render time in minutes.

Max. Paths Per Pixel – Limits the maximum quality achieved by V-Ray IPR, by specifying a maximum number of rays traced for each pixel. Once the specified number has been reached, V-Ray tops rendering.

Max. Noise – Specifies the noise threshold for V-Ray IPR. Once the threshold is met for a certain part of the image, V-Ray stops sampling it and reassigns rendering power to part of the image that requires more sampling.

Low GPU thread priority – (Usable only if the renderer is set to **CUDA**.) When enabled, V-Ray tries to lessen the use of GPUs working on displaying graphics to the monitor(s) in order to give them more resources to complete other processes and tasks for the OS. This is done by using a lower value for **Ray bundle size** for those GPUs with attached displays (*Rays per pixel is still the same for all GPUs*). This can affect overall performance, and the render speed might be reduced. It is recommended to utilize a separate GPU for the display, if possible.



Performance

When CUDA/RTX is chosen as a rendering engine, all unsupported options are disabled from the parameters.

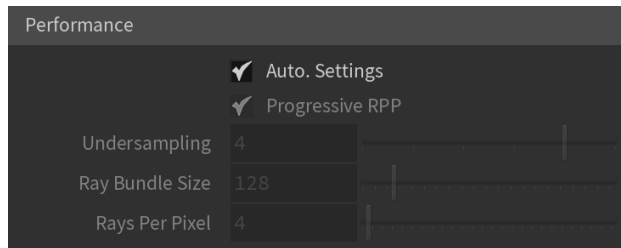
Auto. Settings – Sets predefined optimal settings.

Progressive RPP – When enabled, IPR starts rendering the image with a lower Rays Per Pixel value and then progressively increases it. This speeds up the initial preview of the image.

Undersampling – When enabled, V-Ray IPR starts rendering the image at a lower resolution in order to speed up the initial preview. Later the image is rendered at its final resolution.

Ray Bundle Size – These control the number of rays that are sent to the IPR render servers for processing when using V-Ray IPR, running on CPUs or GPUs respectively. When using distributed rendering, the smaller sizes cause more frequent client/server communication with smaller network packets thus decreasing the speed of the renderer but increasing the interactivity and vice versa. Note that this number is not the exact amount of rays, but is proportional to it. It is not recommended to increase this value beyond 512.

Rays Per Pixel – The number of rays that are traced for each pixel during one image pass. The greater the value, the smoother the picture from the very beginning of the rendering with GI, but interactivity may be significantly diminished. Increasing this value also reduces amount of data transferred from the render servers back to client machine.



Options

Min. Draw Interval – Specifies minimum time, in milliseconds, between image updates. If set to 0, shows all frames.

Max. Draw Interval – Specifies maximum time, in milliseconds, between (partial) image updates. If set to 0, disables the partial image updates.

