

Frequently Asked Questions

⚠ Please read the [Set up GPU Rendering](#) page prior to the FAQ.

Frequently Asked Questions

Which features are supported on the GPU?

The code base for V-Ray GPU differs from the code base for the CPU engine. The GPU code base is optimized for GPU hardware and supports a subset of the features of CPU rendering. For a complete list of supported features, please see the [V-Ray GPU Supported Features](#) page.

Can I choose which devices to be used for GPU rendering?

Yes. There are instructions how to do that in the [GPU Setup](#) section.

How can I check which devices on my system can be used for GPU rendering?

Why isn't my GPU device(s) listed?

Check the list of devices in the **GPU Devices** menu in the **Export** tab of the **V-Ray Renderer** node. If you don't see your device on the list there (the list is empty), try to deselect the **In process** checkbox. This will make V-Ray run as a separate process, working around the limitation.

Why does the UI of my OS become sluggish when I use GPU?

- If you have multiple GPUs, you can speed up screen refresh time by removing the GPU used for monitor/viewport redraw from the devices that GPU uses for rendering.
- If you have only one GPU on your system, you could try reducing the value for **Rays Per Pixel** and/or **Ray Bundle Size** in **V-Ray Renderer node > Renderer > IPR > Performance**. This will break up the data passed to the GPU into smaller chunks so that user interface requests can be processed faster. However, this might reduce rendering speed.
- The **Low Thread Priority** parameter was designed to help with this issue. Turn it on from **V-Ray Renderer > Renderer > Options > Rendering**.

V-Ray will display a warning if the GPU used for monitoring is participating in the rendering process.

How do I reduce the network bandwidth when using GPU with a heavy DR setup?

If you use many GPUs in DR nodes you may find that the network can become a bottleneck at some point, since the GPUs are producing data very fast. Increasing **Bundle Size** / **Rays Per Pixel** increases the size of the workload chunks that are being given to the DR nodes and helps reduce the communication between the client and the servers (you can try something like 192/32 or 256/64).

How do I check the memory usage on the GPU, and reduce the GPU memory needed for my scene?

You can use free 3rd party tools like [MSI Afterburner](#) and [EVGA Precision](#) to monitor both GPU memory usage and utilization. RT GPU also reports how much memory it uses for textures/geometry/light cache/etc in the V-Ray log.

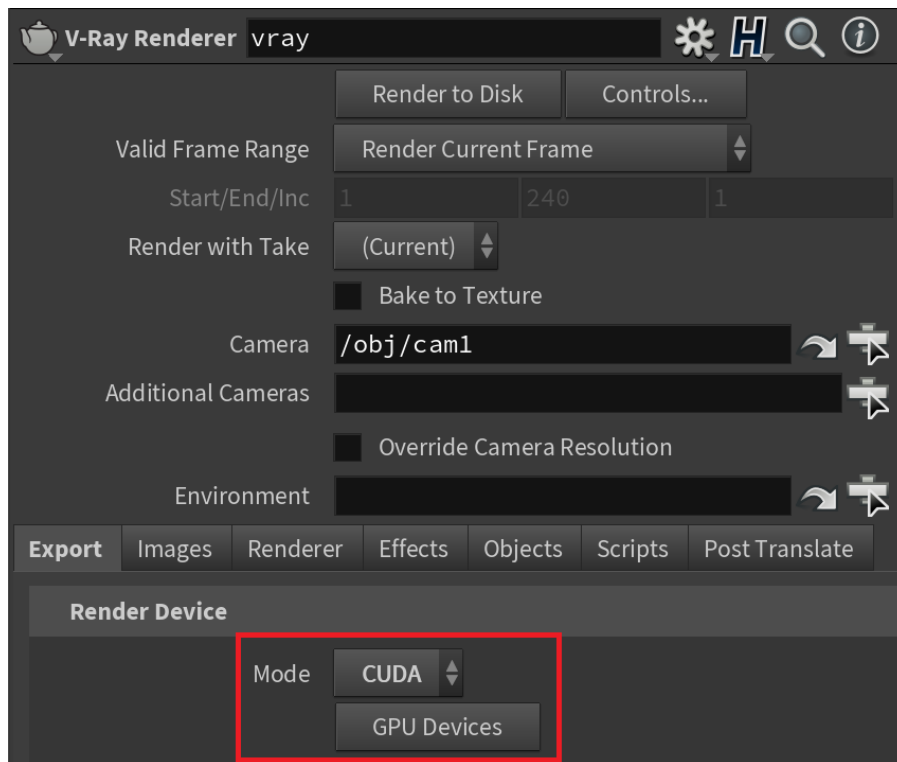
Can GPU be used as production render?

Yes, but within its [limitations](#).

How do I setup V-Ray GPU as my production renderer?

What are the differences between IPR and Production mode?

You can set V-Ray GPU as production renderer by the **V-Ray Renderer** node's parameters (**Export** tab > **Mode**).



By default, the **V-Ray IPR** node will use the settings from the **V-Ray Renderer** node. The main differences are that the IPR mode is interactive and the Production Rendering Mode is not.

How do I use Light Cache with GPU?

In most situations, using Brute Force for your primary engine and Light Cache for your secondary will give you the best ratio between quality and speed in terms of not having to tweak a lot of GI settings to get a nice result. The render will have clean light bounces from accurate GI calculations, with minimal blotchiness or flickering for animated sequences

Why do some bump maps look different with GPU?

Procedural textures are supported for GPU bump maps, but because V-Ray GPU is a very different engine, the bump map itself can look a bit different compared to V-Ray.

Larger scenes cause a "Failed to allocate memory for buffer" error with V-Ray IPR GPU. Why?

If you get the following error message, this means that some or all of your video cards do not have sufficient memory to load the scene.

```
C:\Program Files\Autodesk\Maya2015\vray/bin/vray.exe
[2014/Dec/21 13:58:11] Number of tree leafs: 2084673
[2014/Dec/21 13:58:11] Average faces/leaf: 14.1919
[2014/Dec/21 13:58:11] Memory usage: 1161.42 MB
Failed to allocate memory for buffer
[2014/Dec/21 13:58:17] error: Error 2 transferring vertex normals to
Failed to allocate memory for buffer
[2014/Dec/21 13:58:17] error: Error 2 transferring nodes and meshes to
Failed to allocate memory for buffer
Failed to allocate memory for buffer
Failed to allocate memory for buffer
[2014/Dec/21 13:58:17] Building optimized CUDA kernel for comp cap 2.
[2014/Dec/21 13:58:17] Optimized CUDA kernel assembled successfully
[2014/Dec/21 13:58:17] Total memory for device 0 : 1024 MB, currently
RT Mem usage : Mtls 0.26 MB, Geom 736.57 MB, Total 866.77 MB (new da
_tracedevice.cpp(3069) : CUDA error 2 : unable to allocate enough me
[2014/Dec/21 13:58:18] Device 0 statistics:
[2014/Dec/21 13:58:18] GPU time: 0.086180 %
[2014/Dec/21 13:58:18] GPU ray trace time 0.000000 %
```

To solve this problem, you can enable texture resizing for GPU rendering and set a texture size limit. This will cause IPR GPU to use smaller textures for rendering to conserve memory. See more information on [GPU texture resizing](#).

Why is the GPU rendering not much faster than the CPU rendering?

The difference in render speeds depends on the video cards and the CPUs that are compared, as well as on the scenes used as benchmarks. It is normal to achieve a big speed boost with the GPUs compared to the CPU with some scenes and to have equal performance with others.

Can I use multiple systems with GPU devices to speed up my rendering?

Yes, you can speed up your rendering using DR on multiple systems with CUDA enabled devices. Keep in mind that some drivers may have limitations and may not be able to use GPUs if there is no monitor attached to the machine or you are logged in via Remote Desktop (for example). Check the GPU vendor documentation for more information.

Can GPU be used for scenes with Motion Blur?

Yes. Note that GPU supports motion blurred frames that are integer numbers only. Make sure the starting frame of your render is an integer number for motion blur to work correctly. For example, starting the render from frame 0 will produce correct motion blur, but starting from frame 1.25 will produce incorrect motion blur.

How to start IPR - with or without the VFB?

There are several ways to start rendering with [V-Ray IPR](#). First, create both the V-Ray Renderer and V-Ray IPR nodes in your output network. Make sure you have added a camera into your scene.

1. Go to **Render View** > **Render** button. V-Ray IPR will start rendering directly in the Render View.
2. To render in the VFB, go to **V-Ray Renderer** > **Export** > **Use V-Ray Frame Buffer**. Once enabled, go to **Render View** and press the **Render** button. Alternatively, use the V-Ray Shelf **IPR (VFB)** button or go to **V-Ray** menu > **Render VFB IPR**. V-Ray IPR will start rendering in the VFB.
3. You can use the **Render Region** option, if you go to **Scene View** (viewport).

Can I use V-Ray GPU on macOS? How?

Though GPU rendering with CUDA on macOS devices is no longer supported, the advanced users can bring back the old behavior by using the following [environment variable](#): `VRAY_APPSDK_UNHIDE_DEVICES=1` written in `houdini.env`.

How can I get access to the nightlies section?

If you have registered for V-Ray for Houdini, then you automatically have access to the nightlies section. You can access them [here](#). Note that you have to provide your Chaos Group credentials.