

10.3.10 Render - Cycles Render Engine - GPU Rendering

GPU Rendering.....	1
Introduction.....	1
Configuration.....	1
CUDA.....	1
OpenCL.....	2
Supported Features and Limitations.....	2
Frequently Asked Questions.....	2
Why is Blender unresponsive during rendering?.....	2
Why does a scene that renders on the CPU not render on the GPU?.....	2
Can I use multiple GPUs for rendering?.....	2
Would multiple GPUs increase available memory?.....	2
What renders faster, NVIDIA or AMD, CUDA or OpenCL?.....	2
Error Messages.....	3
Unsupported GNU version! gcc 4.7 and up are not supported!.....	3
CUDA Error: Invalid kernel image.....	3
CUDA Error: Out of memory.....	3
The NVIDIA OpenGL driver lost connection with the display driver.....	3
CUDA error: Unknown error in cuCtxSynchronize().....	3

GPU Rendering

Introduction

GPU rendering makes it possible to use your graphics card for rendering, instead of the CPU. This can speed up rendering, because modern GPUs are designed to do quite a lot of number crunching. On the other hand, they also have some limitations in rendering complex scenes, due to more limited memory, and issues with interactivity when using the same graphics card for display and rendering.

Cycles has two GPU rendering modes: **CUDA**, which is the preferred method for NVIDIA graphics cards; and **OpenCL**, which supports rendering on AMD graphics cards.

Configuration

To enable GPU rendering, go into the User Preferences, and under the System tab, select the Compute Device(s) to use. Next, for each scene, you can configure to use CPU or GPU rendering in the Render properties.

CUDA

NVIDIA CUDA is supported for GPU rendering with **NVIDIA graphics cards**. We support graphics cards starting from GTX 4xx (computing capability 2.0).

Cycles requires recent NVIDIA drivers to be installed, on all operating systems.

List of CUDA cards with shader model

OpenCL

OpenCL is supported for GPU rendering with **AMD graphics cards**. We only support graphics cards with GCN architecture (HD 7xxx and above). Not all HD 7xxx cards are GCN cards though, you can check if your card is [here](#).

Cycles requires recent AMD drivers to be installed, on all operating systems.

Supported Features and Limitations

For an overview of supported features, check the comparison in the *Features*.

CUDA:

The maximum amount of individual textures is limited to 95 byte-image textures (PNG, JPEG, ..) and 5 float-image textures (OpenEXR, 16 bit TIFF, ..) on GTX 4xx/5xx cards, and 145 byte-image textures and 5 float-image textures on GTX6xx cards and above.

OpenCL:

No support for HDR (float) textures at the moment.

Frequently Asked Questions

Why is Blender unresponsive during rendering?

While a graphics card is rendering, it can not redraw the user interface, which makes Blender unresponsive. We attempt to avoid this problem by giving back control over the GPU as often as possible, but a completely smooth interaction can't be guaranteed, especially on heavy scenes. This is a limitation of graphics cards for which no true solution exists, though we might be able to improve this somewhat in the future.

If possible, it is best to install more than one GPU, using one for display and the other(s) for rendering.

Why does a scene that renders on the CPU not render on the GPU?

There maybe be multiple causes, but the most common is that there is not enough memory on your graphics card. We can currently only render scenes that fit in graphics card memory, and this is usually smaller than that of the CPU. Note that, for example, 8k, 4k, 2k and 1k image textures take up respectively 256MB, 64MB, 16MB and 4MB of memory.

We do intend to add a system to support scenes bigger than GPU memory, but this will not be added soon.

Can I use multiple GPUs for rendering?

Yes, go to User Preferences > System > Compute Device Panel, and configure it as you desire.

Would multiple GPUs increase available memory?

No, each GPU can only access its own memory.

What renders faster, NVIDIA or AMD, CUDA or OpenCL?

Currently NVIDIA with CUDA is rendering faster. There is no fundamental reason why this should be so - we don't use any CUDA - specific features - but the compiler appears to be more mature, and can better support big kernels. OpenCL support is still in an early stage and has not been optimized as much.

Error Messages

Unsupported GNU version! gcc 4.7 and up are not supported!

On Linux, depending on your GCC version you might get this error.

If so, delete the following line in `/usr/local/cuda/include/host_config.h`

```
#error -- unsupported GNU version! gcc 4.7 and up are not supported!
```

CUDA Error: Invalid kernel image

If you get this error on MS-Windows 64-bit, be sure to use the 64-bit build of Blender, not the 32-bit version.

CUDA Error: Out of memory

This usually means there is not enough memory to store the scene on the GPU. We can currently only render scenes that fit in graphics card memory, and this is usually smaller than that of the CPU. See above for more details.

The NVIDIA OpenGL driver lost connection with the display driver

If a GPU is used for both display and rendering, MS-Windows has a limit on the time the GPU can do render computations. If you have a particularly heavy scene, Cycles can take up too much GPU time. Reducing Tile Size in the Performance panel may alleviate the issue, but the only real solution is to use separate graphics cards for display and rendering.

Another solution can be to increase the timeout, although this will make the user interface less responsive when rendering heavy scenes. [Learn More Here](#).

CUDA error: Unknown error in cuCtxSynchronize()

An unknown error can have many causes, but one possibility is that it's a timeout. See the above answer for solutions.