

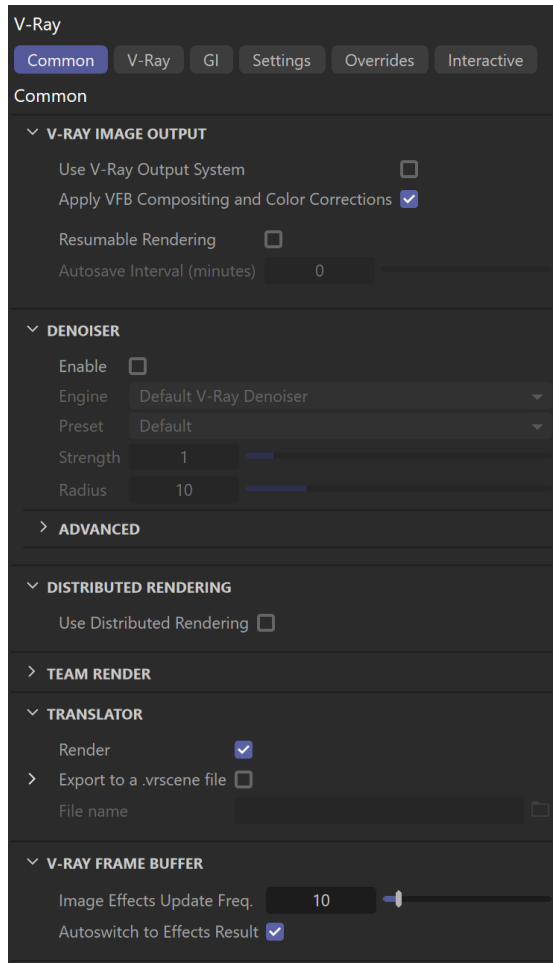
Common Tab

This page provides information about the Common tab in V-Ray Render Settings in Cinema 4D.

Overview

The Common tab of the V-Ray Render Settings provides options for distributed rendering and exporting a .vrscene file.

UI Path: ||Render Settings|| > **V-Ray** > **Common** tab



V-Ray Image Output

Use V-Ray Output System – When enabled, you can use V-Ray to set the output image format of the render. Unlike Cinema 4D's output options, V-Ray can output deep .exr and .vring files.

V-Ray Output System is not compatible with Cinema 4D Team Render.

Apply VFB Compositing and Color Corrections – Enable this option to save all applied VFB color corrections to the RGB channel of the output image. This does not apply to the *sRGB*, *Gamma 2.2* and *ICC display* corrections. OCIO, LUT and Background Image corrections are saved if this is explicitly specified in their VFB rollouts.

File – Specifies the name of the file to be saved.

Format – Specifies the file format in which the image is saved. *Expand to see more options.*

Alpha Channel to Separate File – When enabled, the alpha channel of the rendered image is saved to a separate file.

Don't Save Alpha Channel – When enabled, the Alpha channels is not saved (ignored with multichannel exr and vrimg).

RGB Channel Output – Determines how the color channel outputs are treated.

Output All – Outputs all color channels.

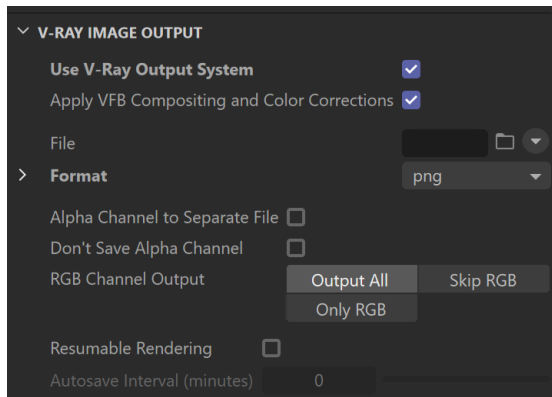
Skip RGB – The RGB channel is not saved (ignored with multichannel exr and vrimg file formats).

Only RGB – Only outputs the RGB color channel. Use this option when you need only a single final image, without any of the elements.

Resumable Rendering – If there is a resumable file existing for the current frame, the rendering is resumed from it. Otherwise, a new render is started and resumable files are saved. If the rendering is already complete, the frame is skipped and the next one (if any) is started. See the [Resumable Rendering tutorial for a basic workflow](#).

Note: Resumable information is stored in additional `.vrimg` or `.vrprog` files, if the output file location has not changed. Resuming a render with bucket sampling uses and creates additional `.vrimg` files. Resuming a render with Progressive Sampling uses and creates separate `.vrprog` (progressive resumable) files. For instance, if resumable rendering is enabled when rendering to a PNG file format using the Progressive Sampler, resumable information is saved as a `.vrprog` file along with the output PNG.

Autosave Interval (minutes) – Specifies an interval in minutes for saving resumable files during rendering. This parameter applies only to [Progressive sampling](#). A value of 0 disables autosaving during rendering and resumable files are saved only at the end of the render.



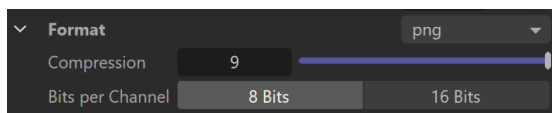
Format

`.hdr` and `.bmp` file formats have no controllable parameters.

PNG

Compression – Allows you to choose compression level for the resulting file. Lower values mean less compression and larger file size. Higher values mean more compression and smaller file sizes. Note that PNG files use a lossless compression, so the resulting image quality is not affected.

Bits per channel – The number of bits for each of the PNG channels (red, green, blue, alpha).



JPG

Quality – Allows you to choose the desired JPEG output quality. Lower values produce a smaller file size and worse quality; higher values produce a larger file size and better image quality.



VRIMG

.vrimg is a Chaos proprietary file format used by V-Ray Frame Buffer to store the rendered image incrementally (bucket by bucket) while rendering in full floating-point format, with all available render elements. It is used as a render output but it is not recommended to use it as a texture file format. VRIMG contains metadata, i.e. render statistics. It supports dynamic bucket size.



EXR

Type – Specifies the exr type to be saved: **Multichannel**, **Deep** or **Normal**.

Compression – Allows you to choose the compression type for the output file:

Default

No compression – Disables all compression.

Run-length – Basic form of compression that is comparable to the one used by Targa/BMP files.

Scanline ZIP – Zip style compression applied to individual scanlines.

Block ZIP – Zip style compression applied to blocks of 16 scanlines at a time. More effective than **Scanline Zip**, but can be slower to decompress.

PIZ – Uses a new combined wavelet / Huffman compression. This form of compression is quite effective when dealing with grainy images, and often surpasses any of the other options for grainy images.

PXR24 – Converts data to 24 bit, then uses Zip compression. It is lossless for half data and 32bit integer data, and slightly lossy for 32bit float data.

B44 – This form of compression is lossy for half data and stores 32bit data uncompressed.

B44A – An extension to B44 where areas of flat color are further compressed. Regular B44 compresses uniformly, regardless of image content.

DWAA – JPEG-like lossy compression format. Compresses 32 scanlines together. ²

DWAB – Same as DWAA, but compresses blocks of 256 scanlines to achieve better compression ratio.

Bits per Channel – The number of bits for each of the PNG channels (red, green, blue, alpha). Allows you to choose between **16-bit (half float)** and **32-bit (full float)** precision.

Extra Attributes – Allows you to specify additional attributes for the OpenEXR file. These are of the format `attr1=val1;attr2=val2;...` where the values can be either integer or floating-point numbers, or lists of numbers in brackets representing vectors, matrices etc. For example:

```
int_attr=53;float_attr=3.14;vec4_attr=(1, 2, 3, 4)
```

If a value cannot be recognized as a number or a sequence of numbers, it is assumed to be a string value.

Auto Data Window – Determines what data window information is written to OpenEXR files. This option works only when the **Memory frame buffer** option of V-Ray is set to **Full**.

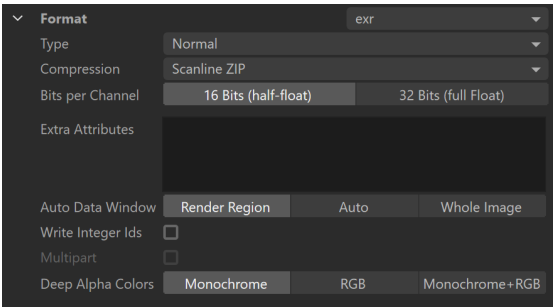
- Render Region** – Sets the data window to the render region.
- Auto** – The data window is determined from the pixels that have non-zero alpha values.
- Whole Image** – Sets the data window to the image resolution.

Write Integer Ids – When enabled, allows you to write ID information in integer form rather than RGB colors.

Multipart – When enabled, multichannel EXRs are written to a single multi-part file. This feature requires OpenEXR 2 support.

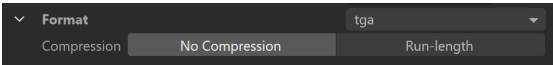
Deep Alpha Colors – Specifies how the alpha is written in the deep .exr file:

- Monochrome** – One alpha exr channel is generated, called "A";
- RGB** – Three color alpha channels are generated: "AR", "AG" and "AB" (red, green, blue alpha);
- Monochrome+RGB** – Generates both the monochrome alpha channel ("A") and the three color alpha channels ("AR", "AG" and "AB").



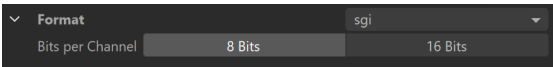
TGA

Compression – Specifies compression options for the resulting image.



SGI

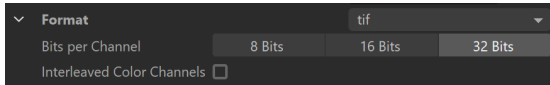
Bits per Channel – The number of bits for each of the PNG channels (red, green, blue, alpha). You can choose between 8 or 16 bits per channel in the resulting image.



TIF

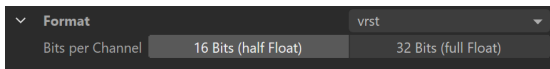
Bits per Channel – The number of bits for each of the PNG channels (red, green, blue, alpha). You can choose between 8, 16 or 32 bits per channel in the resulting image.

Interleaved Color Channels – When enabled, the color information for each pixel is saved using the Interleaved order which means that the full color for each pixel is saved before moving to the next one. Some image processing applications require that the image is saved in this mode.



VRST

Bits per Channel – The number of bits for each of the PNG channels (red, green, blue, alpha). You can choose between 16 or 32 bits per channel in the resulting image.



Denoiser

The V-Ray Denoiser takes an existing render and applies a denoising operation to it after the image is rendered normally. The denoising operation detects areas where noise is present and smooths them out.

Enable – When enabled, starts denoising the rendered image in the V-Ray Virtual Frame Buffer.

Denoising Engines

V-Ray Denoiser offers a choice between the **Default V-Ray denoiser**, the **NVIDIA AI denoiser** and the **Intel Open Image denoiser**. Each offers a different denoising algorithm that comes with different benefits.

Default V-Ray Denoiser - V-Ray's denoising algorithm. It can utilize the CPU or the GPU (AMD or NVIDIA GPUs) to perform the denoising. It is consistent when denoising render elements, as it applies the same denoising operator to all render channels. This means that it is recommended for denoising the render elements to be used for compositing the back to beauty image. In addition, it comes with a [Denoiser Tool](#), which is recommended for denoising animation by using frame blending.

NVIDIA AI Denoiser - V-Ray's integration of NVIDIA's AI-based denoising algorithm. The NVIDIA AI denoiser requires an **NVIDIA GPU** to work. It does not matter whether the actual rendering was performed on the CPU or GPU. This means that rendering on the CPU still requires an NVIDIA GPU for denoising with the NVIDIA AI denoiser and has some advantages and drawbacks compared to the Default V-Ray Denoiser. For example, the NVIDIA AI denoiser performs the denoising faster, but is not consistent when denoising render elements. This means that there will be differences between the original RGB image and the one reconstructed from render elements that are denoised with the NVIDIA AI denoiser. It also doesn't support cross-frame denoising and will likely produce flickering when used in animation.

The NVidia AI denoiser only works on Nvidia Maxwell and newer GPU architectures. If no proper devices are found, V-Ray silently falls back to the Default V-Ray denoiser.

Intel Open Image Denoiser – V-Ray's integration of [Intel Open Image Denoise](#). The Intel Open Image denoiser works with your CPU device and does not use hardware acceleration. This denoiser has no controllable options.

Engine – Allows choosing between the **Default V-Ray denoiser**, the **NVIDIA AI denoiser** and the **Intel Open Image Denoise**. Note that the **NVIDIA AI denoiser** requires an NVIDIA GPU.

Preset – Offers presets to automatically set **Strength** and **Radius** values.

Default – Applies mid-level denoising.

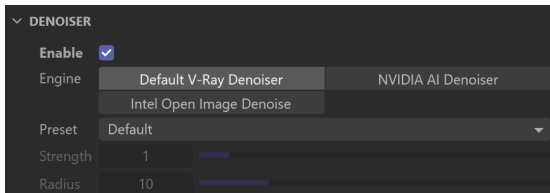
Mild – Applies a more subtle level of denoising than the Default preset.

Strong – Applies a stronger level of denoising than the Default preset.

Custom – Allows the **Strength** and **Radius** parameters to be set to custom values.

Strength – Determines the strength of the denoising operation. This value is automatically set by preset selection, but can be changed when **Preset** is set to **Custom**.

Radius – Specifies the area around each pixel to be denoised. A smaller radius affects a smaller range of pixels, while a larger radius affects a larger range, which increases the noise removal. This value is automatically set by preset selection, but can be changed when **Preset** is set to **Custom**.



Advanced Denoiser

Use GPU Acceleration – Uses the GPU device(s) to accelerate the denoising calculations. In case there is no compatible GPU device, denoising automatically falls back to use the CPU, even if the option is enabled. When the NVIDIA AI denoiser is used, this option is not available. Note that the NVIDIA AI denoiser requires an NVIDIA GPU.

Use NVIDIA Temporal Mode – This option is only available when the Engine is set to **NVIDIA AI Denoiser**. When enabled, the Denoiser uses information from previous frames to create a smoother transition. Useful for rendering animation. Not compatible with the **NVIDIA AI Upscale** option.

NVIDIA AI Upscale – This option is only available when the Engine is set to **NVIDIA AI Denoiser**. When enabled, the image is internally downsampled to half of the resolution specified in the Render Settings, and then upsampled to the full resolution using NVidia technologies. This allows for up to 4 times faster renders, however, it can cause some loss of detail. We recommend using this option only for preview renders. [Render elements](#) that don't have the **Denoise** option enabled are upsampled using simple interpolations. This option is not compatible with the **Use NVIDIA Temporal Mode** option. This option is not supported on macOS.

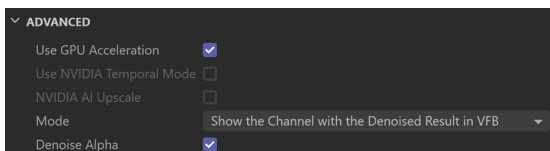
Mode – Specifies how the results of the Denoiser is saved.

Only generate render elements – All render elements required for denoising are generated so that denoising can be done with the [Standalone Denoiser Tool](#) or the V-Ray Denoiser plugin for Nuke. The information calculated within them is not applied to other render elements, and no VRayDenoiser Render Element is generated.

Hide the channel with the Denoised Result in the VFB – The **Denoiser** channel is not present separately in the VFB. The **effectsResult** channel is generated with the denoised image.

Show the channel with the Denoised Result in the VFB – The channel is generated to contain a denoised version of the RGB Color Render Element using the specified settings. The original render elements, including the RGB Color Render Element, are not changed.

Denoise Alpha – Controls whether the alpha is denoised or not.



Distributed Rendering

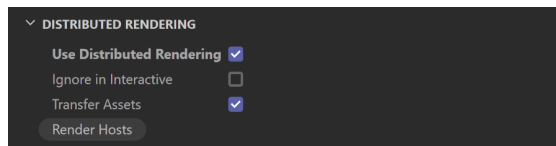
For more information on distributed rendering with V-Ray, see the [Distributed Rendering](#) page.

Use Distributed Rendering – Enables additional parameters and settings needed to distribute the rendering across additional computers. Note: The **distributed_render_hosts.json** file stores the settings information for your DR setup. By default, it is located in `C:\Users\Username\AppData\Roaming\Maxon\Maxon Cinema 4D RXX_XXXXXXX\plugins\vray\`.

Ignore in Interactive – When enabled, **Interactive only** runs on the local machine rather than the whole distributed rendering network.

Transfer Assets – Allows render servers to automatically download missing render assets from the client machine, including textures, V-Ray proxy files, IES profiles, GI solution caches (irradiance maps, light caches), etc.

Render Hosts – Opens the V-Ray DR settings window.



Render Hosts

These options set up the machines that V-Ray uses for Distributed Rendering.

Render on Local Machine – The client/local machine functions as a render server along with the other computers used to complete the rendering task. Disabling it means that all rendering is done on other machines apart of the DR set up.

Host IP or Name – A text field to enter the name or IP address of the machine to use as a DR server.

Port – Specifies the port number where the DR server operates.

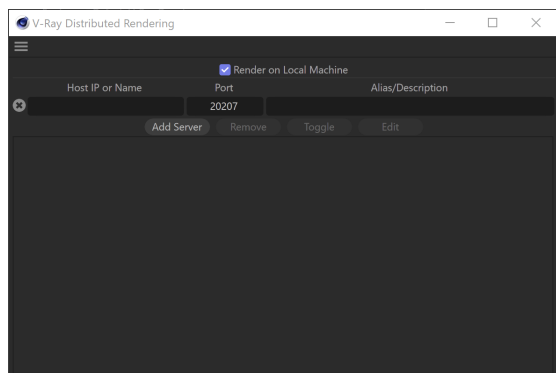
Alias/description – A text field to place notes about the particular DR server.

Add server – Allows a manual way to add a server by entering its IP address or network name.

Remove – Deletes the currently selected server(s) from the list.

Toggle – Toggles the state of the selected server(s) between enabled and disabled.

Edit – Modifies the information for the selected entry in the list.



Translator

Render – Enables V-Ray to render the image in the Frame Buffer and then save an image file. This option can be used at the same time as **Export to a .vrscene file**.

Export to a .vrscene file – Enables V-Ray to export the scene to a `.vrscene` file and expands the options in the rollout. For more information on the `.vrscene` export, refer to the [Exporting V-Ray Scene Files](#) page.

Separate files – When enabled, V-Ray exports separate files for different components of the scene. This also allows you to disable export for certain types of objects.

Each frame in separate file – When enabled, V-Ray makes a separate file for each rendered frame.

Strip paths – When enabled, the paths for assets (bitmap textures, proxies, GI caches loaded from file, etc.) are removed from the exported file and only the file names of the assets remain.

Write mesh data in hex format – When enabled (the default), the mesh data is written in hexadecimal format in the *.vrscene* file. This makes the files smaller and faster for processing, but less readable to a user. When disabled, the mesh data is written in normal decimal format.

Write transform data in hex format – When enabled (the default), transform data is written in hexadecimal format in the *.vrscene* file. This makes the files smaller and faster for processing, but less readable to a user. When disabled, the data is written in normal decimal format.

Compressed – When enabled, basic compression is applied to the *.vrscene* file. This option can only be used together with **Write mesh data in hex format**.

Export Lights – Enable/disable the export of lights. Available when the **Separate files** option is enabled.

Export Nodes – Enable/disable the export of nodes. Available when the **Separate files** option is enabled.

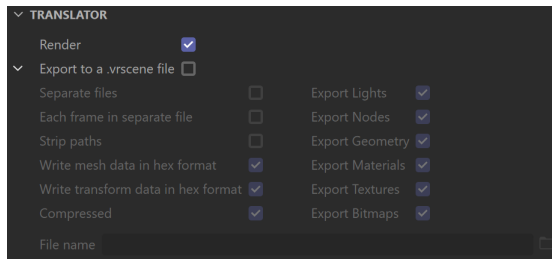
Export Geometry – Enable/disable the export of geometry. Available when the **Separate files** option is enabled.

Export Materials – Enable/disable the export of materials. Available when the **Separate files** option is enabled.

Export Textures – Enable/disable the export of textures. Available when the **Separate files** option is enabled.

Export Bitmaps – Enable/disable the export of bitmaps. Available when the **Separate files** option is enabled.

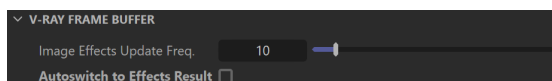
File name – Specifies the location and name for the *.vrscene* file.



V-Ray Frame Buffer

Image Effects Update Freq. – Controls the regularity of updates during progressive rendering. Specifies roughly the percentage of Time Effects, like denoising and/or lens effects, that are allowed to be updated compared to the frame render time. A value of 0 disables updates during the progressive rendering, whereas a value of 100 causes updates as often as possible. Values ranging from 5 to 10 are usually sufficient. The resulting effects are saved as the **EffectsResult** render pass, which appears when the Denoiser or Lens Effects layers are enabled in the VFB.

Autoswitch to Effects Result – With this option enabled, V-Ray Frame Buffer automatically displays the **EffectsResult** channel after calculation of the post effects is done. When disabled, the RGB color channel is displayed instead.



Notes

1. When included in a multichannel EXR, Cryptomatte render elements are always saved with lossless compression. ExtraTex has its own parameter for disabling lossy compression like DWAA/DWAB.