



OctaneRender™ for Maxon® Cinema 4D®

User Manual

Version 2.01

<http://render.otoy.com>

OctaneRender™ for use with
Maxon® Cinema 4D® Software

© OTOY INC. 2014.

All rights reserved.

OctaneRender™ and OTOY® and their logos are trademarks of OTOY Inc.
Maxon® and Cinema 4D are registered trademarks of MAXON Computer GmbH, and/or its subsidiaries
and/or affiliates in the U.S. and/or other countries.

Octane Render for Cinema 4D plugin

Octane for Maxon® Cinema 4D™ is a plugin which allows you to use Otoy OctaneRender™ GPU render engine inside Maxon® Cinema 4D™ Layout.

This manual is currently a work in progress, and is by no means an exhaustive guide to Octane rendering. We hope to post revisions on-line on a regular basis.

Octane for Cinema 4D has been integrated using the Cinema 4D external render class plugin architecture to allow use it with the same workflow that user has working with native Cinema 4D render engine. It also support Octane realtime rendering using a custom Interactive Render Window.



Features of Octane Render for Cinema 4D plugin

This is a list of current Maxon® Cinema 4D and Octane Render features supported by the plugin:

- Loading of the Cinema 4D scenes into the OctaneRender engine and rendering to the Live Viewer, Picture Viewer, Batch rendering and network rendering
- Live update camera, lights, environments, materials and also geometry(experimental)
- Full geometry support, including n-gons, HyperNurbs
- Support for Mograph, particle and hair with geometry option, also some third-party plugins
- Support of full transformations and deformation in polygonal objects, including parenting, bones, etc.
- Support for material selection tags
- Support of dynamics, mograph objects
- Arealights, Infinite Light and Sun Light, Sky supported, **Daylight+Environment texture**
- Support of Octane thinLens, panoramic, orthographic Camera and C4D standart camera
- Render instances
- **Live renderer window** is a interactive render window. It shows rendering in progress with current rendering information. Support for full interactive light/environment/geometry with motion blur, material editing with drag&drop functionality.
- Exporting the scenes and materials to Octane Standalone Software.
- Material converter for changing standart materials and some other types to Octane materials
- Outputs for Alpha, z-depth, material id, wireframe, shading/geometry normals, positions, texture uv channels
- Material LiveDB acces, cached online materials. User created materials for offline usage.



- **Region rendering (2.0 feature)**
- **Camera, Object transform and deformation motion blur (2.0 feature)**
- Subframe motion blur is supported
- Multipasses (Octane Native channels and Cinema 4D standart channels)
- **Native Distributed Rendering (2.0 feature)**
- **Native Hair rendering (2.0 feature)**
- **Displacement (2.0 feature)**
- **Randomcolor texture for instances (2.0 feature)**
- **Region rendering (2.0 feature)**
- **OpenSubDiv surfaces (2.0 feature)**
- **Unlimited texture count (2.01 feature)**
- **Rounded edges (2.0 feature)**
- Cinema 4D Team render(R15) or NET rendering



Limitations:

- Material stacking is not supported
- Spheres in Picture Viewer(you should set RenderPerfect is off)
- Render instances in other render instances is not supported

Octane nodes supported inside Cinema 4D:

- Render target: Camera Imager, Camera postprocessing, Daylight environment, Directlight kernel, Infochannel
- kernel, PMC kernel, Pathtracing kernel, Texture environment, Postprocessing and Thinlens/Panoramic camera.
- Emission nodes: Blackbody emission and Texture emission.
- Material nodes: Diffuse, Glossy, Specular, Lightportal and material mixer nodes.
- Medium nodes: Absorption and Scattering.
- Procedural nodes: Cheks, Marble, Rigidfractal, Turbulence and Wave nodes.
- Texture nodes: Float, GaussianSpectrum, RGBSpectrum and texture image nodes.
- Tools nodes: Clamp, ColorCorrect, CosineMix, Falloff, Invert, Mix texture and

Demo version limitations:

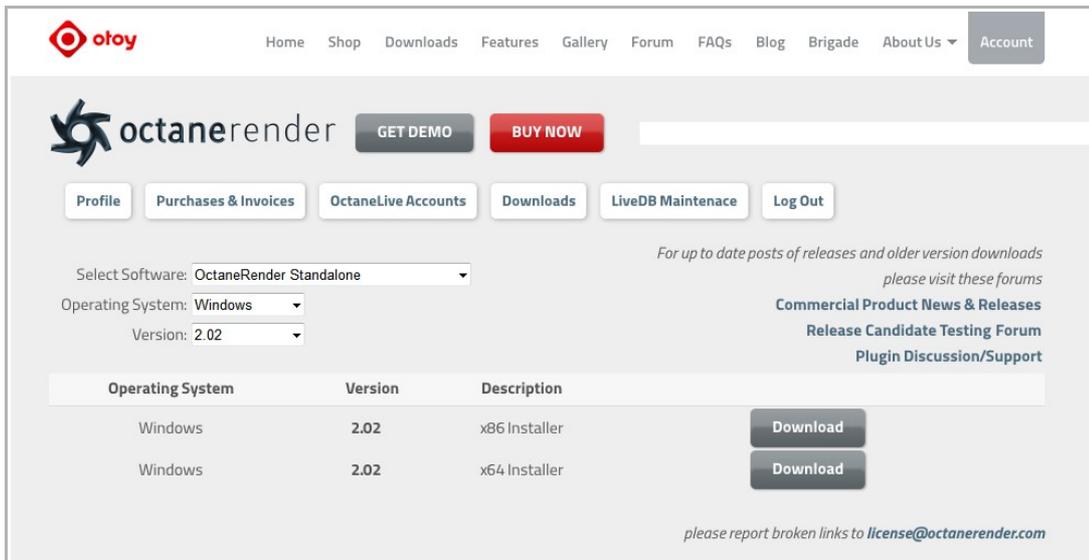
- Watermarked render output with limited resolution. Disabled save option for Liver Viewer.
- Livedb is disabled.
- Network rendering is disabled.

GETTING STARTED

Downloading plugin

The plugin requires OctaneRender™ Standalone Version installed and working properly as a prerequisite. Standalone and plugin can be found from Otoy web site, in Accounts -> Downloads section. Your standalone and plugin userid and passwords are in OctaneLive Accounts section. Please be sure have one of latest Nvidia driver and Standalone can run and render correctly.

To install OctaneRender™ Standalone Version, please refer to the installation chapter of the OctaneRender™ User Manual.



The screenshot shows the OctaneRender website's Downloads section. The navigation bar includes Home, Shop, Downloads, Features, Gallery, Forum, FAQs, Blog, Brigade, About Us, and Account. The main header features the OctaneRender logo, a 'GET DEMO' button, and a 'BUY NOW' button. Below the header, there are buttons for Profile, Purchases & Invoices, OctaneLive Accounts, Downloads, LiveDB Maintenance, and Log Out. The 'Select Software' dropdown is set to 'OctaneRender Standalone'. The 'Operating System' dropdown is set to 'Windows' and the 'Version' dropdown is set to '2.02'. A table lists two download options: 'x86 Installer' and 'x64 Installer', both with 'Download' buttons. To the right, there are links for 'Commercial Product News & Releases', 'Release Candidate Testing Forum', and 'Plugin Discussion/Support'. A footer note asks users to report broken links to license@octanerender.com.

Operating System	Version	Description	Download
Windows	2.02	x86 Installer	Download
Windows	2.02	x64 Installer	Download



The screenshot shows the OctaneRender website's Downloads section for the Cinema 4D plugin. The 'Select Software' dropdown is set to 'OctaneRender for Cinema 4D'. The 'Operating System' dropdown is set to 'Windows' and the 'Version' dropdown is set to '2.01'. A table lists one download option: 'Installer for Cinema4D R12/13/14/15' with a 'Download' button. To the right, there are links for 'Commercial Product News & Releases', 'Release Candidate Testing Forum', and 'Plugin Discussion/Support'. A footer note asks users to report broken links to license@octanerender.com.

Operating System	Version	Description	Download
Windows	2.01	Installer for Cinema4D R12/13/14/15	Download

Hardware and Software requirements

Software Requirements

The plugin requires OctaneRender™ Standalone Version installed and working properly as a prerequisite. To install OctaneRender™ Standalone Version please refer to the installation chapter of the OctaneRender™ User Manual.

IMPORTANT NOTE:

This version requires Nvidia driver 335.xx or newer. Matched Standalone version is required for network rendering. And remove all old plugin files before installing.

Hardware Requirements

OctaneRender™ requires a NVIDIA CUDA-enabled video card. OctaneRender™ runs best on Fermi (e.g. GTX 480, GTX 580, GTX 590, GTX 770/780) and Kepler (e.g. GTX 680, GTX 690) GPUs, but also supports older CUDA enabled GPU models. GeForce cards are fast and cost effective, but have less VRAM than Quadro and Tesla cards. OctaneRender scales perfectly in a multi GPU configuration



and can use different types of Nvidia cards at once e.g. a GeForce GTX 260 combined with a Quadro 6000. The official list of NVIDIA CUDA enabled products is located at <https://developer.nvidia.com/cuda-gpus>.



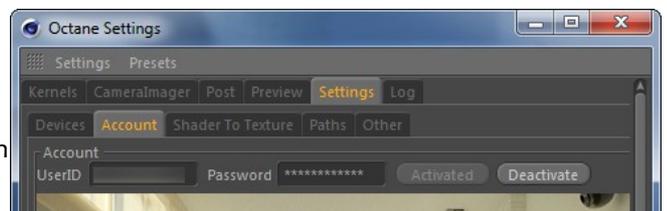
OctaneRender™ is available for the following operating systems :

- Windows XP, Windows Vista, Windows 7, Windows 8 (32 and 64 bit)
- Macintosh OS X (32 and 64 bit)

Internet access

The OctaneRender™ for Maxon® Cinema 4D plugin requires Internet access for the first time launch. Internet access is not required to use both OctaneRender™ and OctaneRender™ for Maxon® Cinema 4D® plugin.

When no internet access is present, the Live Database will not be accessible. Upon launching the plugin for the first time, the user will need to log in to their OctaneLive™ account which associated with their forum log-in information. This information can be found in the customer section of the OctaneRender™ website.



Setting the Plugin into Cinema 4D®

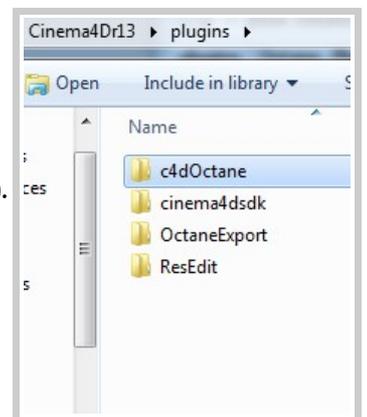
To install OctaneRender™ for Maxon® Cinema 4D Plugin;

Decompress the archive file or use plugin installer which is downloaded from Otoy web site.

Copy the uncompressed files into the Cinema 4d plugin folder (e.g. /Cinema4Dr13/plugins).

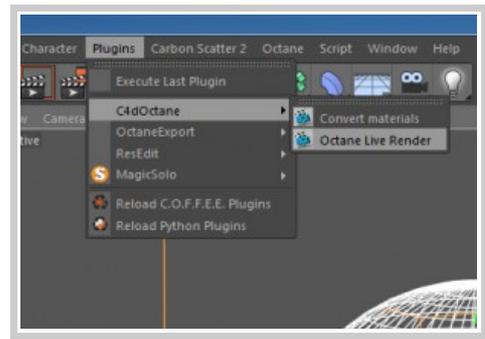
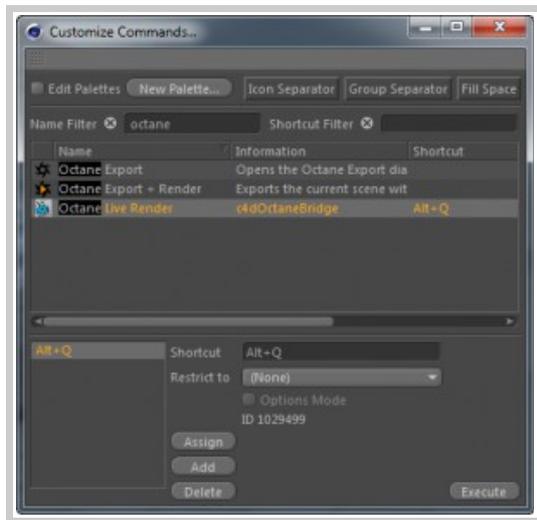
If you're using installer select the version of Cinema 4D and directory's, and wait until install process is completed.

Finally you should have a c4dOctane directory in Cinema4D plugins folder



Starting the Plugin

Start Maxon® Cinema 4D. If everything is correctly setted, you should see an Octane Menu as shown in below image. If the Command is not visible, please check your installation.



CREATE A SHORTCUT FOR THE OCTANE RENDER LIVE VIEWER

Select Window -> Customization -> Customize Commands, or Press ALT+F12, type 'octane' in the Name Filter field and Select Octane > Live Render. Then select and assign a key combination

Plugin Options

Activating the OctaneRender for Cinema4D plugin

OctaneRender for Maxon® Cinema 4D plugin requires Internet access for the first time launch.

When internet access is not available, the Live Database will not be accessible.

When launching the plugin for the first time, the user will need to be logged in to their Octane Live account.

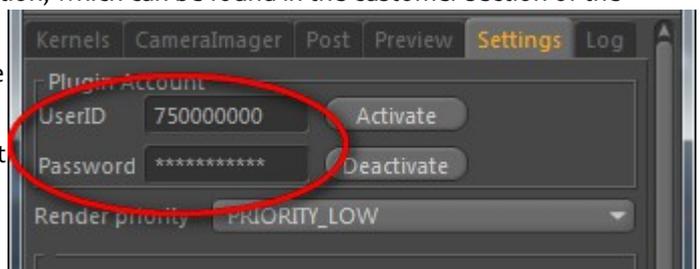
This account is associated with their forum log-in information, which can be found in the customer section of the OctaneRender™ website: <http://render.otoy.com/>

Your plugin account may be disabled using the Deactivate button.

Note: Please be sure you are entering your plugin account details, not your OctaneRender™ standalone account information.

Activating or deactivating the plugin may be done through the Account Page on the OctaneRender™ website.

If you wish to use the plugin on another machine, the plugin must first be deactivated from the current machine, and then activated on the new machine.

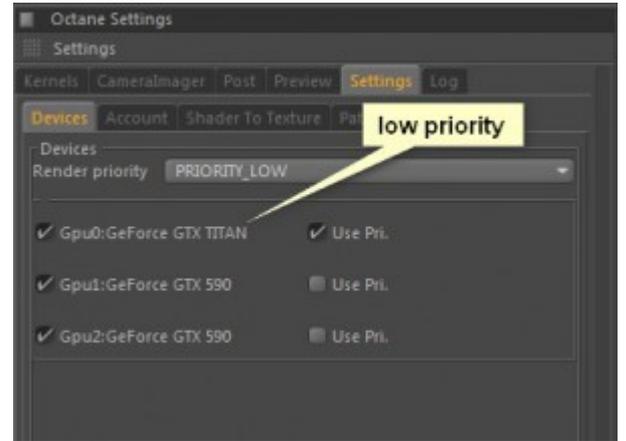


Gpu options

Single or multiple GPUs may be used for rendering. Render devices may be selected or deselected at any time. While rendering you can change active GPU devices. Using an additional card dedicated to the UI and the viewport will ensure faster feedback while working on scenes. The 'Render Priority' option is for adjusting the GPU usage. It is similar to the task manager 'CPU Priority'.

Note about GPU usage:

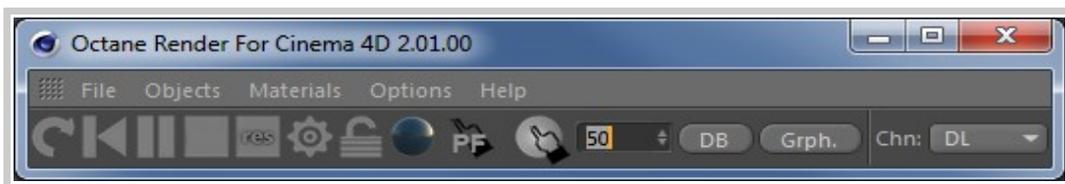
The Plugin requires a NVIDIA CUDA-enabled video card to work. Using an additional card dedicated to the UI and the viewport that will ensure faster feedback while working on scenes. If you don't have an additional card, we recommend using 'low priority' for the video card. If you have additional cards available for rendering, please try with 'low priority' for the viewport card.



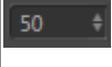
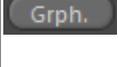
Using The Cinema 4D Plugin

Octane Dialog

Octane Dialog is main control window with several buttons and menus to manage Octane Renderer and creating object, materials, saving the render images and settings for update controls.



Commands

	Send scene and start a render		Activate focus picker
	Restart current render		Activate material picker
	Pause/resume current render		Refresh rate for updates
	Reset all render data		Open Octane Livedb Window
	Open Settings Window		Quick switch for several kernel modes
	Lock Live Viewer resolution to Render Output size		Shows node graph for material or scene structure
	Clay rendering modes		

Objects menu

Octane Camera: Creates a camera with OctaneCameraTag. Which has all octane camera settings

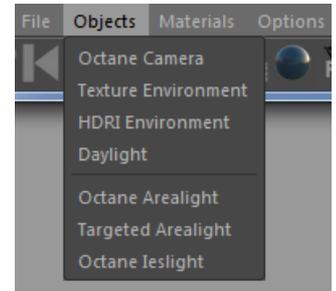
Texture Environment: Creates a Octane Texture Environment. Result object is Sky+OctaneEnvironmentTag. Which has settings of Texture Environment. You can change environment texture/color nodes and other settings. You can generate IBL environments.

Daylight: Creates a Directlight + OctaneDaylightTag + Sun Expression tag. Which has all Daylight settings of Octane Renderer. OctaneDaylightTag has Turbidity, power, north offset and new model parameters. Also you can enable/disable Sun Expression for exact simulation.

Octane Arealight: Creates Arealight + Octane Lighttag. Actually Octane has not any light object. All lighting is done with emission nodes on geometries. Plugin can export Cinema 4D arealights as geometry and attach Texture/Blackbody emission nodes. Lighttag allows to adjust emission parameters.

Targetted Arealight: Creates Arealight + Octane Lighttag + Target Expression. Similar above command. Also has a target node. If selection is a object, set the object as a target, otherwise it creates null object for targetting

Octane Ieslight: Creates Arealight + Octane Lighttag with predefined ies light settings. You must set the distribution node with ies file



Updates menu

Check Camera: Updates octane camera

Update Materials: Updates Octane Materials on any material changes

Update Lights: Updates Octane Lights on any light changes

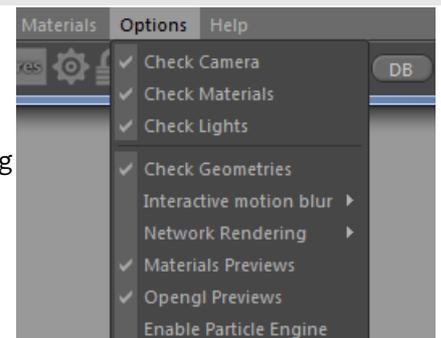
Update Geometries: Reflects any changes on objects which have OctaneObject Tag (MeshNo:1-8)

Material Previews: On/Off for Octane Render material previews

OpenGL Previews: On/Off for OpenGL material previews

Enable Particle Engine: Experimental

Sub Sampling: Set sub-sampling 2x2, 4x4 or off



Materials menu

Octane Diffuse Material: Creates a Octane diffuse material

Octane Glossy Material: Creates a Octane glossy material

Octane Specular Material: Creates a Octane specular material

Octane Mix Material: Creates a Octane mix material

Octane Portal Material: Creates a Octane portal material

Convert Materials To Octane: It converts selected materials to Octane material

Remove Unused materials: Removes unused materials in scene

Files menu

- **Save as png8 image:** Saves current render buffer of Live Viewer as a 8-bit png image
- **Save as png16 image:** Saves current render buffer of Live Viewer as a 16-bit png image
- **Save as exr image:** Saves current render buffer of Live Viewer as a linear exr image
- **Save as tonemapped exr image:** Saves current render buffer of Live Viewer as a tonemapped exr image
- **Save C4D scene with assets:** Saves scene to directory with all used image textures
- **Import Orbx:** Load materials from a file which is in Octane orbx format
- **Save Ocs:** Saves current Octane Render data as OCS file (Octane Standalone scene format)
- **Edit in Standalone:** Saves current Octane Render data as OCS file and open in Octane Standalone

Live Viewer Window

Live Viewer is an interactive render window for Octane Render. It supports several realtime updates

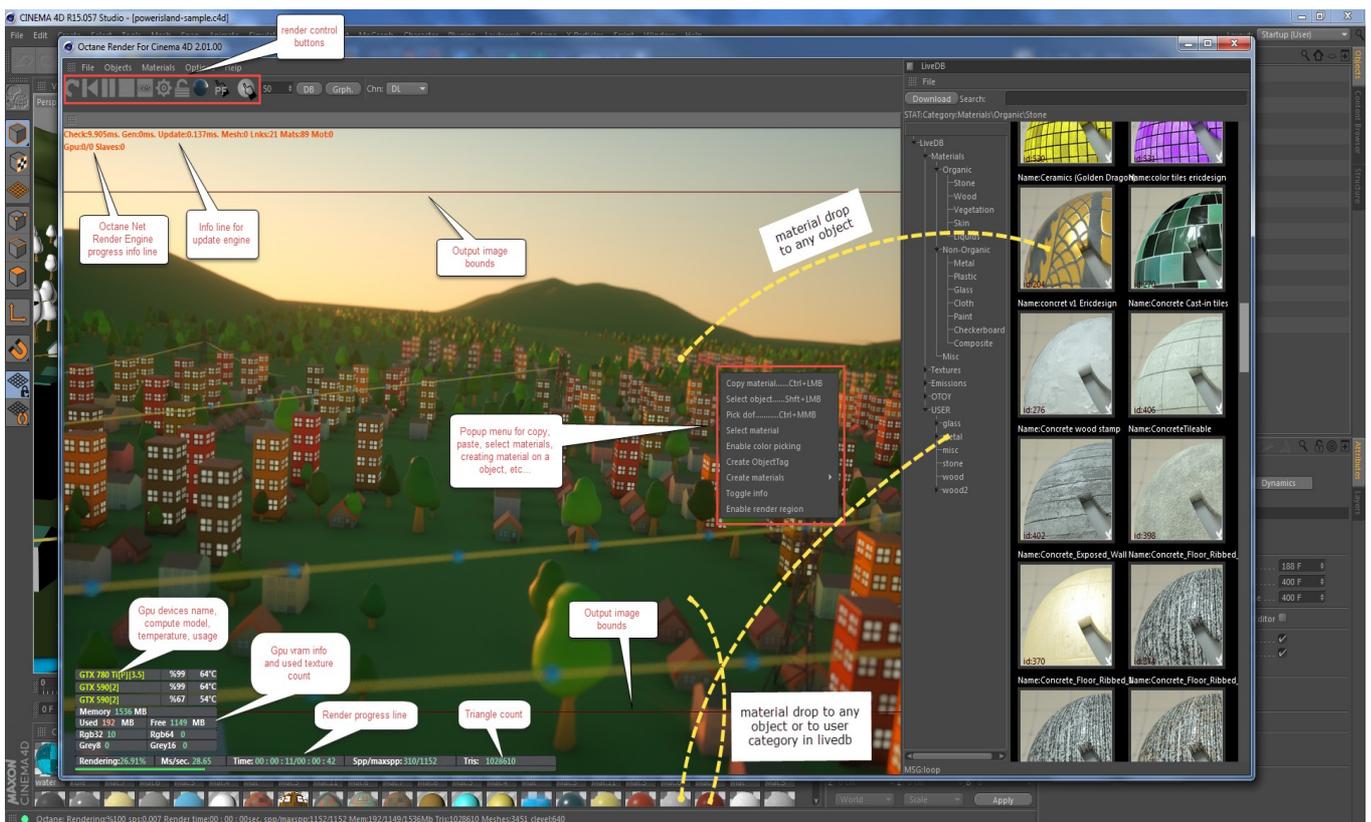
In the viewer window, you can navigate the current camera with ALT+mouse button, similar to controls of the OpenGL Viewport.

Also you can select materials from render view and pick and change camera focal distance.

It has some usefull information about render progress and system.

- Render progress, speed, render and completion times, current/max samples, total triangles count.
- Used/Free Memory, used/maximum texture counts, used GPU's, usage of GPU power and temperature Modes of Information lines can be changed by left mouse click on Live Viewer.
- Support for material picking, drag and drop from livedb and material manager to any objects in render view.
- Support for dof, color picking.
- Material creation on any object in render view.
- Copy/paste/select materials, select objects from render view.
- Enable/disable the render region commanf from popup menu.
- Clay rendering modes: Gray, solid and normal.

OVERALL LOOK TO INTERFACE



Octane Render Settings Window

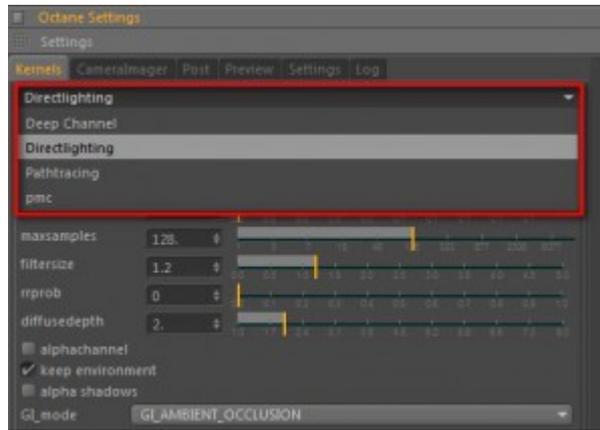


To open Render Settings window press “Open Render Settings” button on Live Viewer

Using the Kernels

There are four major rendering kernels in OctaneRender™:

Direct Lighting, Path Tracing, PMC and Deep Channel



1 - Direct Lighting

Direct Lighting is used for faster preview rendering. Direct Lighting is not unbiased but is useful when creating quick animations or renders.

-**Specular depth** controls the number of times a ray can be refracted before dying. Higher numbers mean higher render times but more color bleeding and more details in transparent materials. Low numbers can introduce artifacts, or turn some refractions into pure black.

<p>Specular Depth 0 All the objects appear black</p>	<p>Specular Depth 2 Rays can enter transparent objects but cannot render the other chess pieces</p>

	
<p>Specular Depth 5 Rays can find the other chess pieces but not ones in their refraction</p>	<p>Specular Depth 8 All chess pieces are properly rendered</p>

-Glossy depth controls the number of times a ray can be reflected before dying. Higher numbers mean higher render time. Low numbers under "4" can introduce artifacts, or turn some reflections into pure black. You should setup this setting based on the complexity of the scene you are working on, and especially based on how many reflective parallel surfaces you have.

-Diffuse Depth: Gives the maximum number of diffuse reflections if GI Mode is set to Diffuse (4)

-AO Distance: The distance of the ambient occlusion in units. Always check if the amount is right related to your model scale. For example you don't need an amount of "3" units if your object is a small toy. But if your model is a house or something large, you can increase the value. The more you increase the value the darker your render will be.

The **ray epsilon** is the distance to offset new rays so they don't intersect with the originating geometry. If the scale of your scene is too large, precision artifacts in the form of concentric circles may appear. In that case, increasing the ray epsilon can make these artifacts disappear.

Maximum Samples: This sets the maximum number of samples per pixel before the rendering process stops. The higher the number of samples per pixel, the cleaner the render. For quick animations and scenes with predominantly direct lighting, a low amount of samples (500-1000) may suffice. In scenes with lots of indirect lighting and mesh lights, a few thousand samples may be required to obtain a clean render.

Filter Size: This sets the pixel size for filtering the render. This can improve aliasing artifacts in the render. Noise can also be reduced this way, but if the filter is set too high, the image can become blurry.

Alpha Channel : This option removes the background and renders it as transparent (zero alpha). This can be useful if the user wants to composite the render over another image and does not want the background to be present.

Keep Environment: This option is used in conjunction with the Alpha Channel setting. It allows the background to be rendered with zero alpha but is still visible in the final render. This allows even further flexibility in compositing images.

Alpha Shadows: This setting allows any object with transparency (specular materials, materials with opacity settings and alpha channels) to cast a proper shadow instead of behaving as a solid object.

Russian Roulette Probability (rrprob): In path tracing, maxdepth is the maximum amount of bounces a ray can make, but, after 3 bounces, there is a 50 % chance that the ray is killed, if you set the rrprob parameter to 0.5. In practice it will rarely exceed more than 16-20 bounces once every million pixel samples (generating a firefly usually). If the rrprob is set to 0, it uses a automatic setting. If it is set to anything else, the user directly controls the probability the ray is ended at random after 3 bounces and so forth.

GI Modes

There are **five** different Global Illumination modes in the Direct Lighting Kernel:

None: Only direct lighting from the sun or area lights is included. Shadowed areas receive no contribution and will be black.

Sample Environment: Use a simple ambient colour from the environment/horizon. Together with ambient and none, these modes are all very fast, as no Monte Carlo sampling is required. These give a very unrealistic, classic z-buffer/whitted raytracing style look, but are very fast, and very handy for interactive tuning of complex scenes or on slow hardware.

Diffuse: Indirect diffuse, with a configuration to set the number of indirect diffuse bounces to include. This gives a GI quality that is in between Ambient Occlusion and pathtracing, without caustics and a decent realistic quality (better than AO), but much faster than pathtracing/PMC. It is very good for quick finals and animations. It is similar in some ways to 'bruteforce' indirect GI in other engines.

Ambient: Use a simple ambient colour from the environment above.

Ambient Occlusion: Standard ambient occlusion. This mode can often provide realistic images but offers no color bleeding.

2 - Path Tracing

Path Tracing is best used for realistic results (together with PMC). The render times are higher than Direct Lighting but the results can be photorealistic. It can have some difficulties with small light sources and proper caustics (for which pmc is better suited).

Maximum Depth: The maximum number of times a ray can bounce/reflect/refract in a surface. Higher amounts mean also higher render time but more realistic results. For outdoor renders a good setting is around 4 maxdepth. For lighting interior with natural light (the sun and the sky) you will need higher settings such as 8 or higher to allow enough light to bounce around in the scene. While high values are possible, in reality rays will not usually go beyond 16 ray depth.

Russian Roulette Probability (rrprob): The RRprob is the probability of terminating the path randomly during raytracing. This value should be left as the default. A value of zero allows OctaneRender™ to determine the value.

Ray Epsilon: The ray epsilon is the distance to offset new rays so they don't intersect with the originating geometry. If the scale of your scene is too large, precision artifacts in the form of concentric circles may appear. In that case, increasing the ray epsilon can make these artifacts disappear.

Maximum Samples: This sets the maximum number of samples per pixel before the rendering process stops. The higher the number of samples per pixel, the cleaner the render. There is no rule as to how many samples per pixel are required for a good render.

Filter Size: This sets the pixel size for filter for the render. This can improve aliasing artifacts in the render. If the filter is set too high, the image can become blurry.

Alpha Channel: This option removes the background and renders it as transparent (zero alpha). This can be useful if the user wants to composite the render over another image and does not want the background to be present.

Keep Environment: This option is used in conjunction with the Alpha Channel setting. It allows the background to be rendered with zero alpha but is still visible in the final render. This allows even further flexibility in compositing images.

Alpha Shadows: If alpha maps are used in the scene, this setting controls whether the shadows will be calculated from the mesh geometry or from the alpha map.

Caustic Blur: This is used to approximate caustics on rough surfaces and increase or decrease the sharpness of caustic noise. A zero value provides the sharpest caustics and increasing this value increases the blurring effect to make caustics appear softer.

3 - PMC

PMC is a custom mutating unbiased kernel written for GPUs. It allows for complex caustics and lighting to be resolved.

Maximum Depth: The maximum number of times a ray can bounce/ reflect/refract in a surface. Higher amounts mean also higher render time but more realistic results. For outdoor renders a good setting is around 4 maxdepth. For lighting interior with natural light (the sun and the sky) you will need higher settings such as 8 or higher. While high values are possible, in reality rays will not usually go beyond 16 ray depth.

Russian Roulette Probability (rrprob): The RRprob is the probability of terminating the path randomly during raytracing. This value should be left as the default. A value of zero allows OctaneRender™ to determine the value.

Ray Epsilon: The ray epsilon is the distance to offset new rays so they don't intersect with the originating geometry. This value should be left as the default.

Exploration Strength: This specifies how long the kernel investigates good paths before it tries to find a new path. Low values can create a noisy image while larger values can create a splotchy image.

Direct Light Importance: The direct light importance makes the kernel focus more on paths with indirect light. For example, imagine sunlight through a window that creates a bright spot on the floor. If the direct light importance is 1, the kernel would sample this area a lot, although it becomes clean very quickly. If the direct light importance is reduced, the kernel reduces its efforts to sample that area and focuses more on more tricky areas that are harder to render.

Max Rejects: This can control the "bias" of the render. By reducing the value, the result will be more biased, but the render time will be shorter.

Maximum Samples: This sets the maximum number of samples per pixel before the rendering process stops. The higher the number of samples per pixel, the cleaner the render.

Filter Size: This sets the pixel size for filter for the render. This can improve aliasing artifacts in the render. If the filter is set too high, the image can become blurry.

Alpha Shadows: If alpha maps are used in the scene, this setting controls whether the shadows will be calculated from the mesh geometry or from the alpha map.

Alpha Channel: This option removes the background and renders it as transparent (zero alpha). This can be useful if the user wants to composite the render over another image and does not want the background to be present.

Keep Environment: This option is used in conjunction with the Alpha Channel setting. It allows the background to be rendered with zero alpha but is still visible in the final render. This allows even further flexibility in compositing images.

Caustic Blur: This is used to approximate caustics on rough surfaces and increase or decrease the sharpness of caustic noise. A zero value provides the sharpest caustics and increasing this value increases the blurring effect to make caustics appear softer.

Parallelism:

This is used to reduce the number of samples that are investigated in parallel to make caustics appear earlier at the expense of some performance.

4-Deep Channel Kernel

The `deep_channel_kernel` creates false-color images of the scene, containing various types of information about the scene. In scenes where the environment is visible you should enable the alpha channel.

The following settings are available:

Geometric normals: the vectors perpendicular to the triangle faces of the mesh.

Shading normals: the interpolated normals used for shading. This does not take into account the bump map of the object. For objects without smoothing this is identical to the geometric normals.

Position: The position of the first intersection point.

Z-depth: The distance between the intersection point and the camera, measured parallel to the view vector.

Material ID: Every material pin is represented as a separate color.

Texture Coordinates:

Tangent Vector: shows the UV coordinates for the surface (this is only useful for export script writers).

Wireframe: Shows the mesh as represented by edges, vertices and surfaces

Interpolated Vertex Normals: Similar to shading normals preset but also strictly calculates the shading based on vertex normals provided by actual data in the mesh. This makes flawed normals resulting from stuffed vertices visible in the shading.

For display these values are scaled to get values approximately between 0 and 1. All tone mapping settings except for min_display_samples and gamma are ignored. Exposure is enabled for Z-depth, and will indicate the value which gets mapped to white. To save these channels you should use untonemapped EXR.

Maximum Samples: This sets the maximum number of samples per pixel before the rendering process stops. The higher the number of samples per pixel, the cleaner the render. There is no rule as to how many samples per pixel are required for a good render.

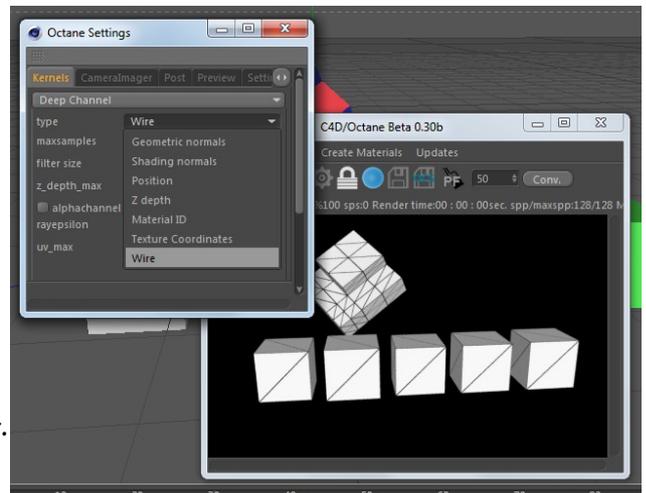
Filter Size: This sets the pixel size for filter for the render. This can improve aliasing artifacts in the render. If the filter is set too high, the image can become blurry.

Maximum Z-Depth: Gives the maximum z-depth that can be shown.

Alpha Channel: This option removes the background and renders it as transparent (zero alpha). This can be useful if the user wants to composite the render over another image and does not want the background to be present.

Ray Epsilon: The ray epsilon is the distance to offset new rays so they don't intersect with the originating geometry. This value should be left as the default.

Maximum UV value: Gives the maximum value that can be shown for the texture coordinates.



Camera Imager Settings

Exposure: Controls the exposure of the scene. Smaller values will create a dark scene while higher values will brighten the scene.

f-Stop: This controls the f-Stop just as in a real camera. Lower values allow more light into the camera and therefore brighter scenes. Higher values allow less light in, so the scene will be darker.

ISO: This controls the ISO setting as in real photography. The higher the ISO setting, the noisier the final picture.

Gamma: This adjusts the gamma of the render and controls the overall brightness of an image. Images which are not properly corrected can look either bleached out, or too dark. Varying the amount of gamma correction changes not only the brightness, but also the ratios of red to green to blue.

Response: The use of measured camera response curves can be selected. Please see Appendix 2 for examples of all the settings.

Vignetting: Adjusting this parameter increases the amount of darkening in the corners of the render. Used sparingly, it can greatly increase the realism of the render.

Saturation: Adjusts the amount of color saturation of the render

Hot Pixel Removal: The Hot Pixel Removal slider is used to remove the bright pixels (fireflies) during the rendering process. While many of the pixels can disappear if the render is allowed to progress, the Hot Pixel Removal feature allows the bright pixels to be removed at a much lower Sample per Pixel.

Pre-multiplied Alpha: Checking the Pre-multiplied Alpha button multiplies any transparency value of the output pixel by the pixels color.

Min Display Samples: This is minimum amount of samples that is calculated before the image is displayed. This feature can significantly reduce the noise when navigating and is useful for real-time walkthroughs. When using multiple GPUs, it's recommended to set this value as a multiple of the number of available GPUs for rendering, e.g. if you're rendering with 4 GPUs, set this value at 4 or 8.

Dithering: Adds random noise which removes banding in very clean images.

Saturate To White: When the sun is too bright, it can create multicolored reflections. Increasing this value will change the colours to white. This is also applicable to all sources of light. Fully saturated parts of the render can be pushed towards pure white with this option. This helps avoid large patches of fully saturated colours caused by over-bright light sources such as very bright coloured emitters or reflected sunlight off coloured surfaces.

White Balance: White Balance is used to adjust the temperature in the image resulting from the tints produced by different light sources. This allows you to achieve the most accurate colors possible. White Balance is on by default.



Image rendered without White Balance



Image rendered with White Balance

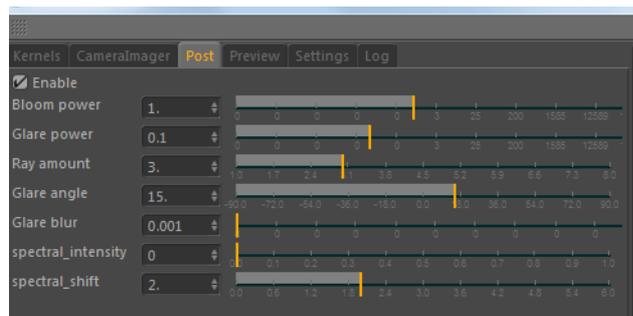
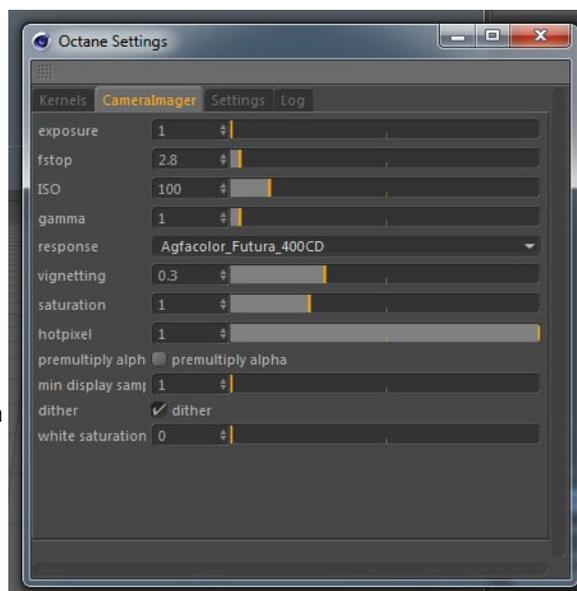
Postprocessing

On/Off: This is a boolean value used to enable or disable post-processing effects on the resulting render. Post-processing is disabled by default.

Bloom Power: Controls the size of the glow originating from an emitter, the size of the halo of light originating from the sun and/or concentrated light on reflective glossy materials to add a bloom effect to the rendered image.

Glare Power: Controls the size of the visible rays originating from an emitter, the size of the glare originating from reflective glossy materials at a point where concentration of light is at the highest to add a glare effect to the rendered image.

Glare Ray Amount: Controls the number of visible rays radiated or reflected.



Glare Angle: This is used to adjust the direction of the glare relative to the object. A glare angle of -90 and 90 results to one main horizontal glare, and a glare angle of 0 results to one main vertical glare.

Glare Blur: Controls the sharpness of the glare. Smaller values will result to a crisp linear glare and this is softened as the value is adjusted and set higher.

Settings

This Section has some options are related with plugin.

Plugin Account: Plugin owner id/password and activation/deactivation buttons

Render devices: To activate GPU devices and set render priority

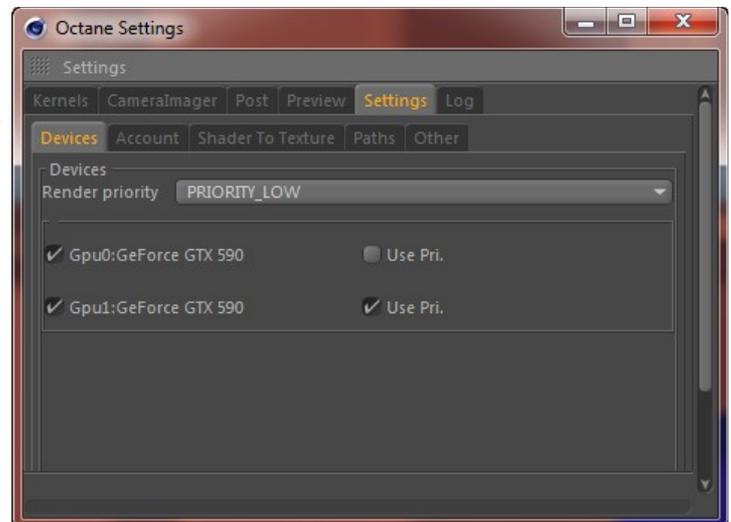
Shader to texture: A experimental feature which convert C4D native shaders to textures. If it's activated, Octane Render plugin use unsupported shaders as rendered textures.

Only uv mapped shaders works. Plugin tries to convert Cinema 4D textures to images and put as imagetextures. Some shaders cannot supported in this way.

Texture size option is to change the baked texture resolution. Bigger values takes longer times while material exporting.

Paths: Texture search paths to replace Cinema 4d paths. The Plugin search unfound textures in this directories.

Others: Some other setting for plugin



Settings/Devices

OctaneRender™ is a GPU based render engine and it is important to manage the GPU(s) in the system that are used for rendering. This is done through the Devices tab. Under this tab, the checkboxes for GPU's that are not supported are not shown and users can only enable GPU's with a supported compute model.

Checkboxes which is on left names of GPU, allows selecting GPUs to use for rendering if more than one GPU is installed.

The Use priority option can throttle down rendering on one or more GPU's to improve system responsiveness especially when rendering on a GPU used for the display.

GPU memory/texture usage, can be checked from on-screen info lines of Live Viewer.

Settings/Account

You need a plugin and standalone Account separately. This section is for plugin Account data. First time you can enter your plugin USER/PASSWORD to activate plugin. After pressing the Activate button, plugin will register your Account and also check for Standalone Account.

If your Standalone is not activated, should be seen a second window which is for entering Octane Standalone Account data. For some reasons like usage for other machine or os, you can use Deactivate button. Also you can check Account section from OTOY website. Sometimes, you may need to restart Cinema 4D after activating.

Settings/Shader to texture

A experimental feature which convert C4D native shaders to textures. If it's activated, Octane Render plugin use unsupported shaders as rendered textures. Only uv mapped shaders works. Plugin tries to convert Cinema 4D textures to images and put as imagetextures. Some shaders cannot supported in this way.

Texture size option is to change the baked texture resolution. Bigger values takes longer times while material exporting.

Settings/Texture Paths

OctaneRender™ for Maxon® Cinema 4D plugin uses this Texture paths for looking texture files. The plugin searches for texture files and animation files in the following locations:

- In the same folder as the scene.
- In the Tex folder within the scene's folder.
- If a given bitmap was saved to an absolute path, CINEMA 4D will, in this case, only search within this path, and will ignore all other defined search paths.
- If textures from CINEMA 4D libraries are used, plugin copies to tex folder of project and uses it from there.
- Also it looks to search paths and subfolders of each paths.

Settings/Others

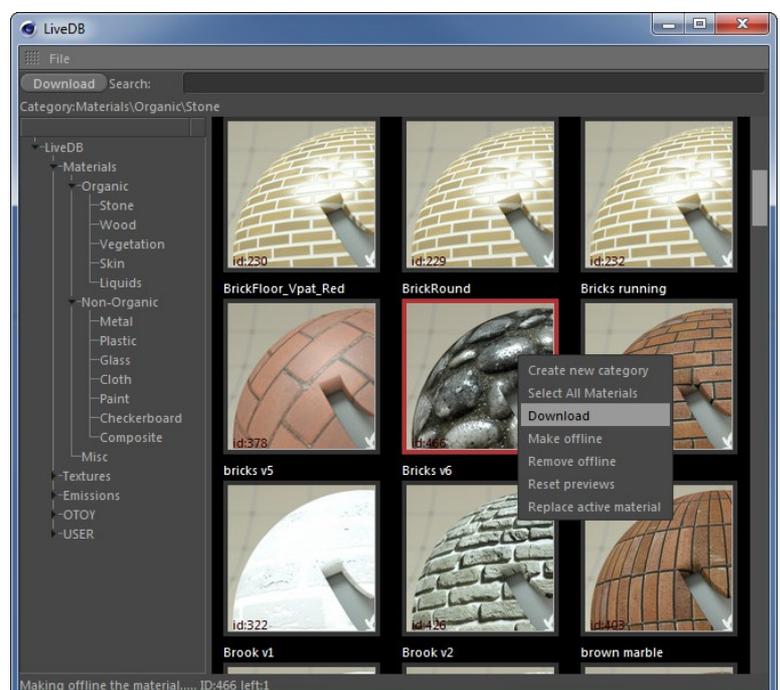
- **Preview samples:** Octane render maxsamples for material previews. Set to lower values for faster material previews.
- **Preview size:** Render size of procedural textures for material previews. This affects appearance of material previews if it has some procedural textures.
- **Opengl samples:** Octane render maxsamples for opengl material previews
- **Texture power:** This is default xbitmap(c4d standart image) output power factor. 1 is solid(white), 0.9 is transparent for opacity images
- **Log output:** This gives some outputs for about progress of plugin. It's usefull for catching some troubles.
- **Xbitmap gamma:** Gamma value of Standart texture maps.
- **Auto backup:** This enables Live Viewer autosave feature.
- **Backup time:** Value of backup timer for autobackup.
- **Use octane for opengl:** use octane render preview function for generating opengl textures, otherwise will use software functions
- **Render stats:** Render info line
- **Gpu stats:** Gpu names, compute model, temperature and usage info line.
- **Texture Stats:** This gives used and max texture amounts of gpus. Types are rgb32, rgb64, grey8 and grey16.

LiveDB Support

The The Cinema 4D® plugin has access to the OctaneRender™ online LiveDB:
Most materials contains textures with images. Textures of materials will be saved to tex directory of current project.

You can select different categories from tree control. Several materials can be selected at one time.

Selected materials can be inserted with download command. Also you can drag materials and drop to movable objects in Live viewer .

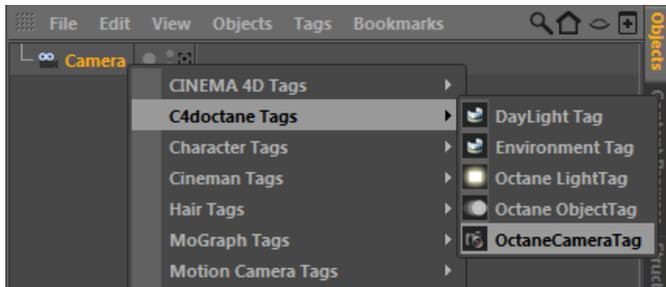


Octane Camera

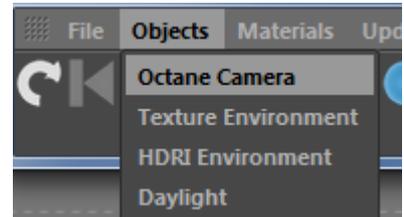
Octane plugin can work with standard Cinema 4D cameras. OctaneCamera Tag should be added to a camera as in picture to use the Octane Camera properties,



CREATION:



From object popup menu, c4doctane Tags -> OctaneCameraTag



LiveViewer -> Objects -> Octane Camera

Camera focus can be managed in several ways:

1. Auto Focusing (Auto Focus is checked in Thinlens section in OctaneCameraTag), this parameter has priority over 2., 3. and 4. clauses
2. Focal Depth parameter from Thinlens section in OctaneCameraTag, this parameter has priority over 3. and 4. clauses
3. Focus Object from camera properties
4. Focus Distance from camera properties

Activating **Depth of field** effect, it's done with Aperture value.



Camera Types

Thinlens Camera

Aperture (aperture)

The aperture of the camera in the scene. Choosing a low value will have a wide depth of field where everything is in focus. Choosing a high value will create a shallow depth of field (DOF) where objects in the foreground and background will be out of focus.

Stereo (stereo)

This enables whether anaglyphic stereo rendering is active. When active, the render will be able to be viewed with Red / Blue 3D glasses or side by sides mode.

LeftFilter (leftFilter)/ RightFilter (rightFilter)

The left and right filter colors are used to adjust the colors used to create the anaglyphic stereo affect in the render.

Perspective correction

Lens Shift (lensShift)

This is useful for architectural rendering, when you want to render images of tall buildings/structures from a similar height as the human eye, but keeping the vertical lines parallel

Near Clip Depth (nearClipDepth)

The main purpose was for interior scenes where you want to get a good shot of the whole room but you cannot do so without a very large FOV, because you need to keep the camera inside the room. With camera clipping (near plane), you can position the camera outside the room - lower the FOV and increase the clipping plane distance in front of you until the closest walls are clipped out. The geometry is not altered, only the clipping of the camera, which means that shadows, reflections and refractions are still affected by the clipped geometry.

Orthographic (orthographic)

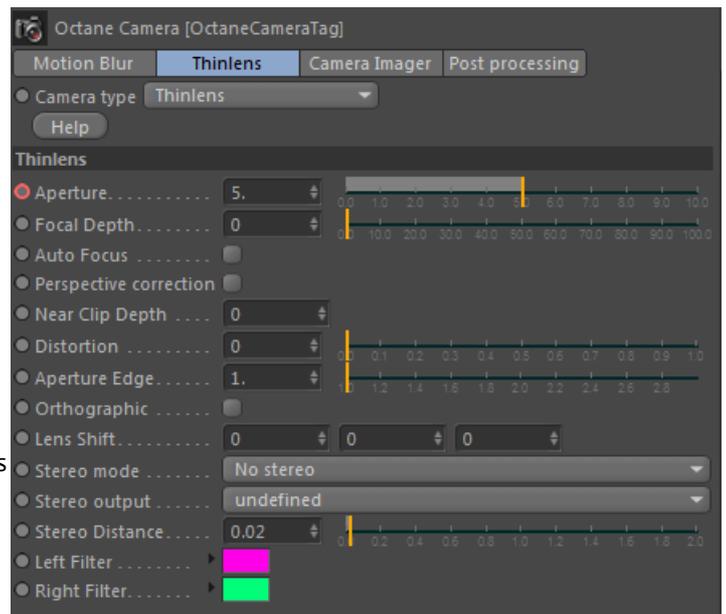
This enables or disables Orthographic Camera mode. When active, all same-length parallel edges display as the same length regardless of the perspective that should result from the object's orientation.

Aperture Edge (aperture_edge)

This controls aperture edge detection at all points within the aperture. The lower values will give more pronounced edges to out of focus objects affected by the a shallow depth of field (DOF) such as objects in the foreground and background.

Distortion (distortion)

This adjusts the spherical and cylindrical distortion. The rendered image displays the entire sphere and uses equidistant cylindrical projection also known as lat-lon projection.



Panoramic Camera

Camera Mode (cameramode)

Provide the option to use either a Spherical Camera Lens or a Cylindrical Camera Lens

Field of View (fovX)

This sets the x coordinate for horizontal field of view of the camera in the scene.

Field of View (fovy)

This sets the y coordinate for the vertical field of view of the camera in the scene.

Adjusting Lighting

OctaneRender™ currently supports Environment Maps / HDRI files, Sun /Sky environments and mesh emitters as lighting options.

Sky Object, Physical Sky, Infinite/Sun/Area lights are supported by Octane Render for Cinema 4D plugin. Octane Render need a light source to rendering.

The Plugin uses a default texture environment, if there is no any Sky object .

If you put one or more Sky Objects, The Plugin uses first one.

If you put a directlight/sunlight then Sky objects will be ignored. Daylight Environment will be activated.

Adding a arealight will not disable the Sky object effect. You must put a Environment Tag with texture or color nodes or a standart material.

Environment Maps / HDRI Environments

Cinema 4D usage

Sky Object + Octane Environment Tag

Preset Buttons

Gray: Create a gray setup on Octane Environment Tag

Hdri: Create a HDRI setup on Octane Environment Tag

PARAMETERS

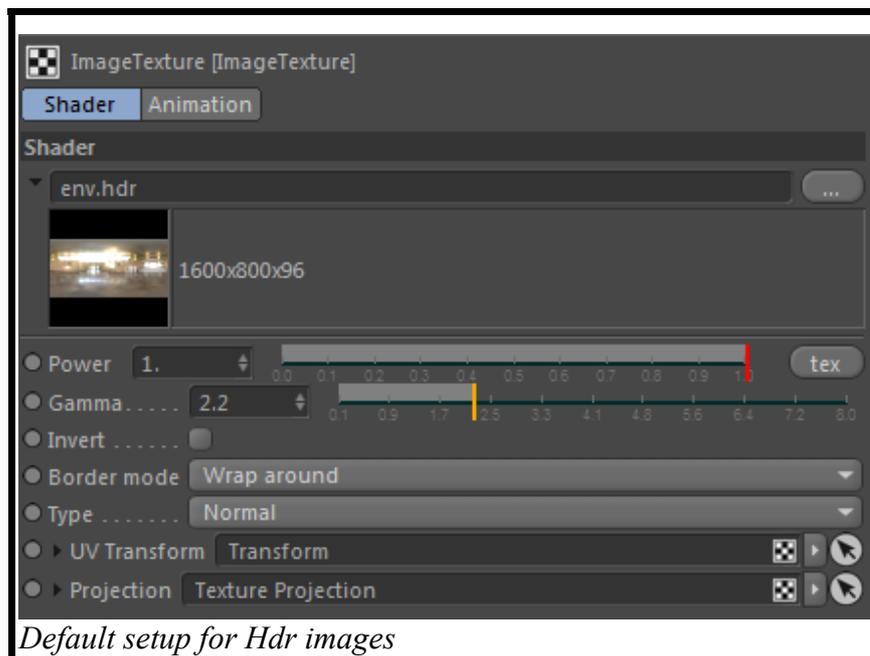
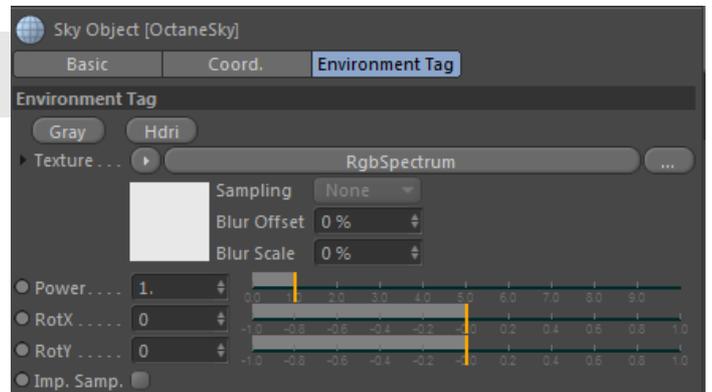
Texture: The Texture affects the color of the environment.

User can link a texture node to set this color from any map, including an HDRI map.

Power: The power of the entire environment can be adjusted using the Power parameter Rotation: If the environment is not in the correct rotation, it can be adjusted in real-time by using the Rotation parameter.

Importance Sampling: Enables quicker convergence (noise reduction) for HRDI images by applying importance to certain areas of the HDRI so as to sample rays that resolve to the important areas more often than unimportant areas. This means the ray tracing is more efficient as it is not wasting time on rays that do not contribute much to the overall image quality. Due to the added complexity this will reduce the Ms/sec of rendering, but each Ms will be more effective, therefore resulting in a higher image quality in less time. Importance sampling is enabled by default.

RotX and RotY: Offset values for Environment texture. This parameter affects the Transform node of Texture. So texture mush have a transform node. To use a ImageTexture Shader that gives more control over Environment Image. If you have washed image try to change gamma value. Gamma 1.0 value is better for most hdr textures mostly



Octane Daylight

Octane has a powerful DayLight Environment node to simulate Sky and Sun lights.

User can select how to set the Sun direction, using the geographic data or using the Sun angles.



Standard Directlight, are supported by Octane Render for Cinema 4D plugin.

Cinema 4D usage:

Infinite/Sun Light + Octane Daylight Tag

Create

Live Viewer->Create->DayLight

Longitude / Latitude

These parameters can be set to get realistic sun settings for the specified geographic location.

Month / Day / GMT Offset / Hour

These parameters can be used to accurately place the sun in the sky according to the date / time for the sun at the current longitude / latitude.

But this parameter replaced with a Sun Expression tag. Default is disabled on creation of DayLight

Turbidity

The Turbidity can be used to adjust the sharpness of the sun lights shadows. A low value creates sharp shadows (like on a sunny day) and a higher value diffuses the shadows similarly to a cloudy day.

Power

The Power slider can be used to adjust the strength of the light. This can affect overall contrast and exposure level of the image.

North Offset

The North offset slider can be used to adjust the actual north direction of the scene. This is useful for Architecture Visualization to ensure the direction of the sun is accurate to the scene.

Model (old/new)

This specifies which daylight model to be used as the current environment. The old daylight model lights a scene with basic spectral radiance assuming that the sun is moving over the horizon at a constant distance from the object. The new daylight model is used to simulate full spectrum daylight providing more sky color variation as the sun moves along and then bears shorter rays as the sun moves closer to the normal plane.

Sky Color/Sunset Color

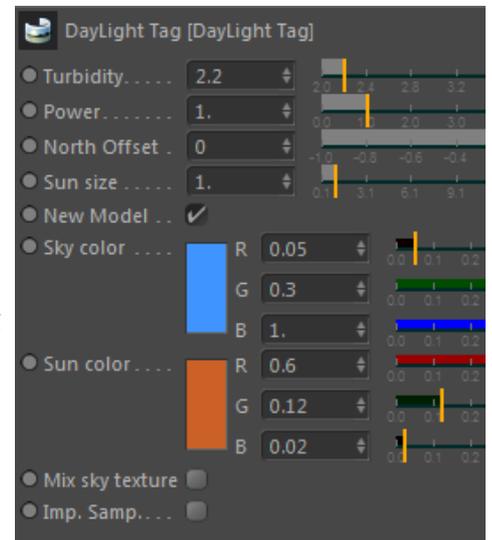
These settings are used by the new daylight model to customize the spectral shade of light. This can affect overall mood expressed by the image.

Sun Size

This allows users to control the sun radius in the daylight environment.

Mix sky texture

Allows to use texture environment and daylight together



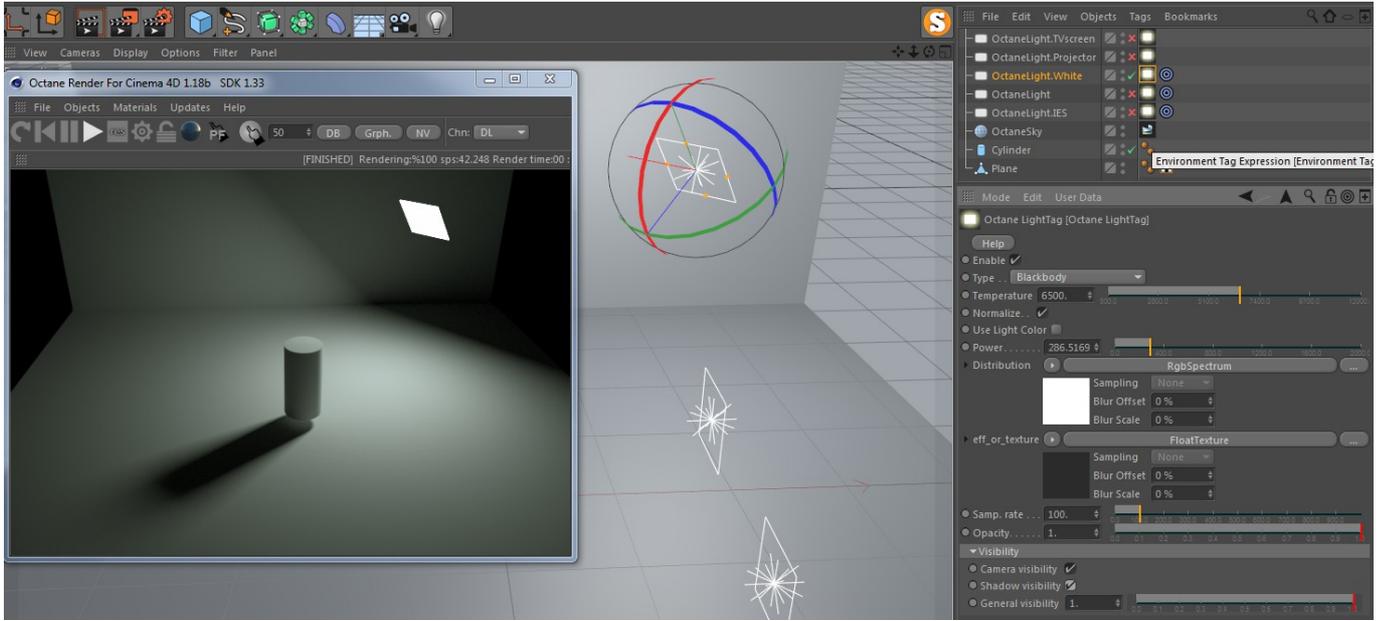
Octane Light

Standart **Arealights** are supported by Octane Render for Cinema 4D plugin.

OctaneLightTag is designed for giving Octane emission texture features to Arealights. Each tag generates a special material with emission and assing to light geometry.

It allows realtime light transforms, geometry and properties editing. Result of updates immediatly affects active render data.

There is two emission types as Blackbody and Texture emission. Also IES lighting effect can be obtain with assignment of IES data to Distribution slot.



A sample for Arealight and OctaneLightTag

OctaneLightTag

Parameters

Enable - Activates tag

Type - Switch for blackbody or texture emissions

Temperature - The temperature (in K) of the light emission.

Normalize - (for blackbody emitters) is used to keep the luminance of the emitted light from a blackbody constant if the temperature varies, instead of making it brighter if the temperature increases.

Use Light Color - This enables output color sync to light color. It's usefull to produce animated colors with mograph effectors

Power - This is the wattage of the light source. Each light in the scene should be set to its real world wattage. For example, a desk lamp could be set to 25 watts, a ceiling lamp to 100 watts, and an LED light to 0.25 watts. This setting should not be used to balance the lighting power of the scene.

Distribution - The distribution controls the pattern of the light. This can be set to a floattexture and an image or IES file can be loaded.

Efficiency or Texture - This setting is used to set the efficiency of the light source. No light is 100% efficient at delivering the power at the specified wattage (a 100 watt light bulb does not actually deliver 100 watts of light.) The efficiency setting can be used to enter the real world values.

These values can be used to create very realistic light settings. For example, a standard 100 watt incandescent bulb would only be approximately 2.0% efficient where as a 25 watt compact fluorescent light will be 10% efficient. These will both produce around the same quantity of light in real life.

Light source sampling rates - in earlier versions of OctaneRender™ light sources were sampled according to area. In

some scenes this will lead to a poor balance between light sources. In a room with a big TV (which emits some light) and a couple of small bright spotlights, the TV would have been sampled most of the time, leading to a lot of noise in the light from the spotlights.

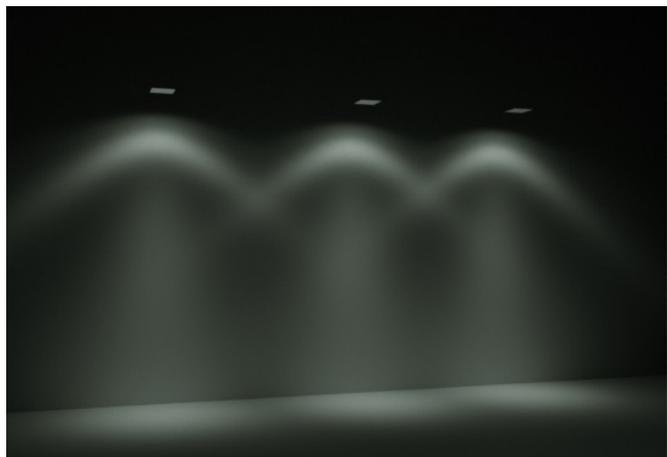
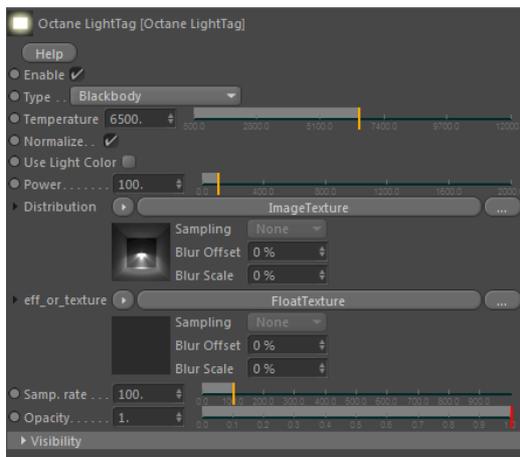
Emission nodes have a parameter called `sampling_rate`, to control how much weight is given to the emitter when picking an emitter to sample. This allows you to choose which light sources will receive more samples. In the above example you can increase the rate for the spotlights, so they get sampled adequately.

Producing IES Lighting effect

Shortcut:

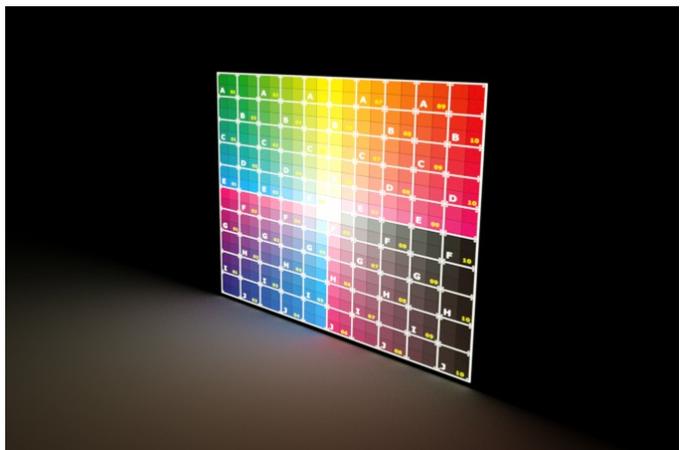
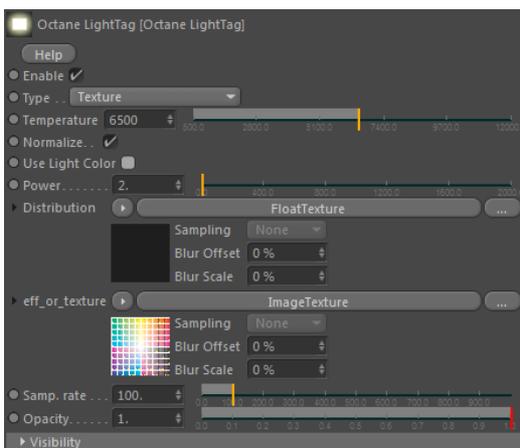
Octane Dialog->Objects->Octane Ieslight

It creates premade setup with ies file in distribution, texture projection is setted to spherical with rotX=90 degree. Transform node in ies image allows to scale ies effect. Make sure projection is setted Object Coordinates



Projection Lighting effect

Efficiency or Texture – this parameter can produce the effect with a imagetexture



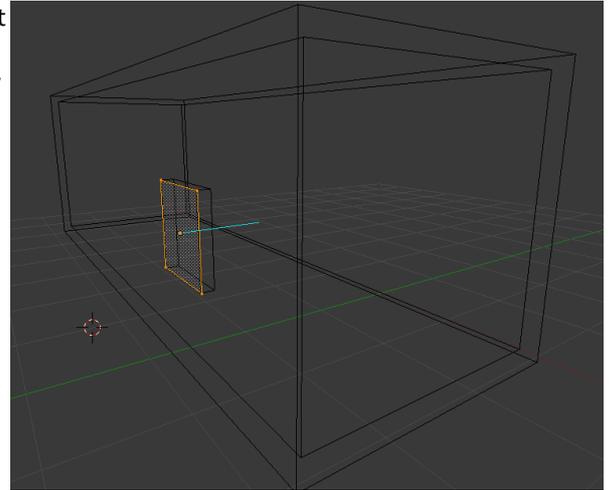
Mesh Emitters

Mesh Emitters allow an object to be used as a light source. This can be used with both Texture and Sun / Sky lighting systems. In order to use a mesh as a light source, it needs to be changed to a Diffuse material type. The Emission parameter has two different options: Blackbody and texture emissions

Using Portals

Portals are a technique to help the render kernel find important light sources. In interior renderings with windows, it is difficult for the path tracer to find light from the outside environment and optimally render the scene. Portals are planes that are added to the scene in the host modeling program that are used by OctaneRender™ to more efficiently render the scene.

In the image to the right, a room is being prepared with a small, single window. This would be a difficult scene to light with a sun/sky or HDRI file with no lighting on the interior of the room. A single plane was placed over the window (orange) with the normal for the plane facing into the room (blue line).



Portal Usage Notes:

When using Portals, all opening must be covered with a portal. It will not work if only one window has a portal over it when all other windows do not have a portal over them. The normals on the Portal object **MUST** be pointed into the scene or the render kernel will not use it properly. Currently, portals cannot be placed in openings which are not open, eg a window with a portal cannot contain glass at this time. In some complex scenes and situations, portals might slow down the render, so a bit of experimentation with/without should be done. Portals only apply to pathtracing type kernels, eg pathtracing and PMC. (not directlighting/ambient occlusion) It is best to try to use the least amount of geometry for portals, eg only a few simple rectangular planes are best, the more geometry your portals contain, the slower the engine might become.

Sometimes it is better to place one large portal over many small windows due to the above. It's ok to make a portal larger than the opening, just make sure it closes/covers all opening(s). A portal which is unnecessarily large will end up slowing down the efficiency, as some of the rays through the covered parts of the portal will not go outside the space. Portals, when defined with the portal material, will not show up in your render, eg this will be invisible geometry.

Adjusting Materials

There are five types of materials in OctaneRender™ currently.

- **Diffuse** - Used for dull, non-reflecting materials or mesh emitters
- **Glossy** - Used for shiny materials such as plastics or metals
- **Specular** - Used for transparent materials such as glass and water
- **Mix** - Used to mix any two material types
- **Portal** - Used to designate openings in scenes to allow the render kernel to better sample light from those areas

But Cinema 4D plugin has three types of Octane materials: Octane Material, Octane Mix Material and Octane Portal Material.

Octane Material owns Diffuse, Glossy and Specular types can be switched by Material Type control.

So material type easily can be changed without losing the parameters.



Diffuse Material

The Diffuse value gives the material its color.

Bump / Normal

Both the Bump and Normal channels can load images to control the amount of bump mapping and normal mapping. The Bump channel should be set to floatimage to load a bump map. The Normal channel should be set to the image data type to load a full color normal map.

Opacity

Opacity sets the transparency of the material. Set the data type to alphaimage (if the image has an alpha channel) or floatimage (for black/white images) to load an image to set the transparency (use the Invert checkbox if necessary to adjust whether black or white regions are considered transparent).

Normal Smoothing

Normal Smoothing is a Boolean value that sets whether to smooth the normals of all meshes sharing that material. When off, the materials can be faceted and polygonal.

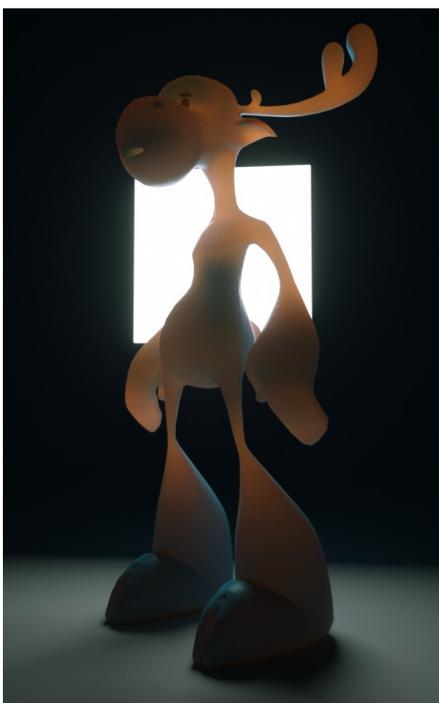
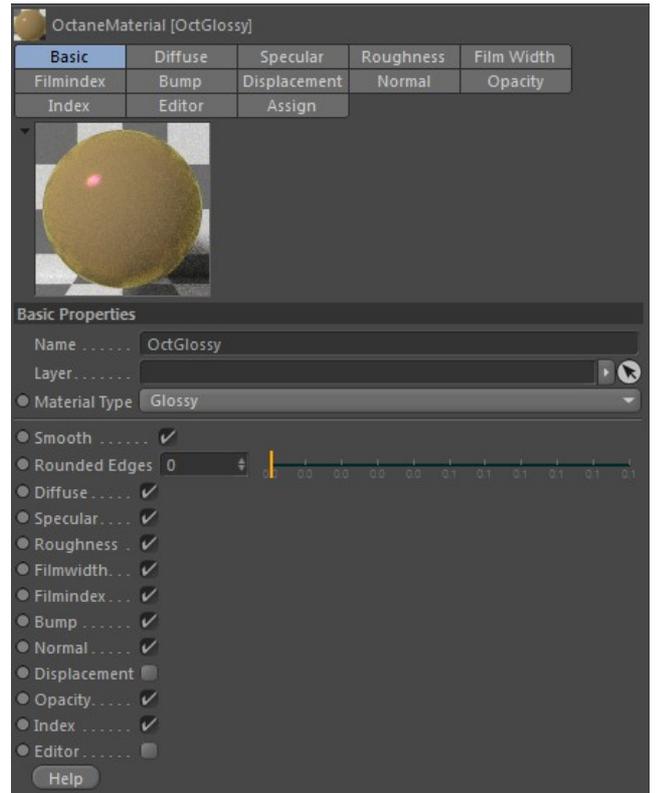
Emission

The Emission setting controls whether the material acts as a light source.

Matte

This is a Boolean value to enable or disable a matte mask of a diffuse material.

For applying absorption, SubSurfaceScattering and emission on Diffuse Material, refer to Chapter 5 Section 5.9.



SSS sample with diffuse material which has transmission value and absorption medium in medium channel



Glossy Material

Glossy materials have these parameters to adjust:

Diffuse (diffuse)

The value gives the material its color.

Specularity (specular)

The value determines the amount of specularity on the mesh.

Roughness (roughness)

The roughness determines the amount of reflection that will be present. A low roughness value will create blurry reflections and a high value will produce a mirror like reflection.

Bump / Normal

Both the Bump and Normal channels can load images to control the amount of bump mapping and normal mapping (respectively.) The Bump channel should be set to floatimage to load a bump map. The Normal channel should be set to the image data type to load a full color normal map.



Film Width (filmwidth)

This controls the thickness of a optical, thin film on the material. This is useful in creating rainbow or oil slick effects.

Film Index (filmindex)

This controls the Index of Refraction of the thin film.

Opacity (opacity)

Opacity sets the transparency of the material. Set the data type to alphaimage(if the image has an alpha channel) or floatimage(for black/white images) to load an image to set the transparency (use the Invert checkbox if necessary to adjust whether black or white regions are considered transparent).

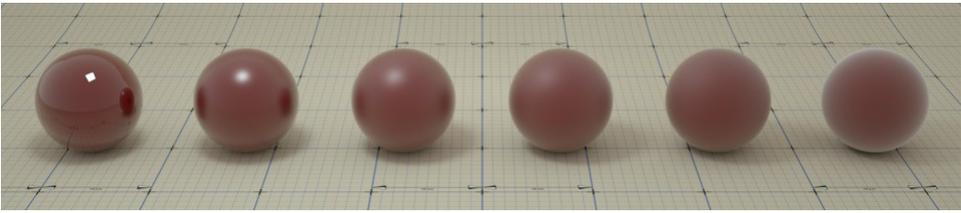
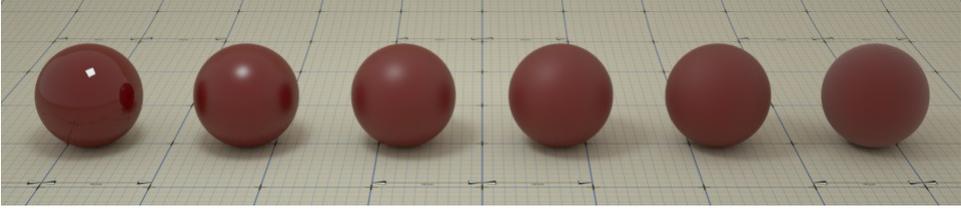
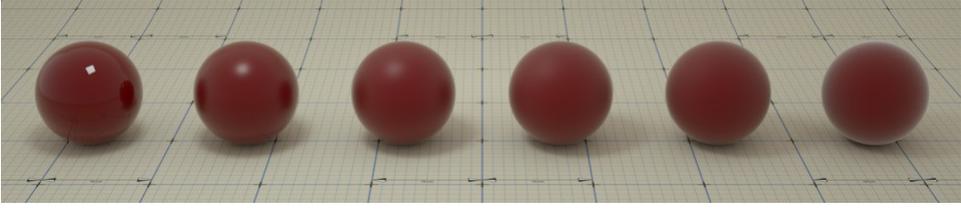
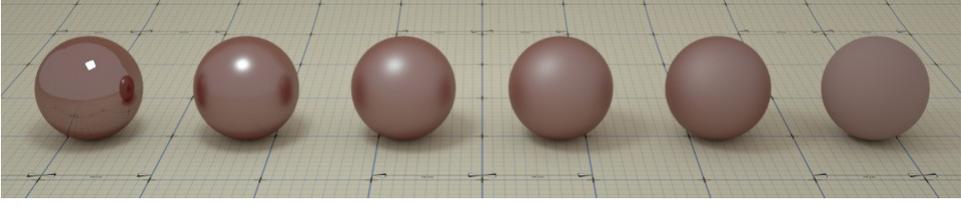
Normal Smoothing (smooth)

Normal Smoothing is a Boolean value that sets whether to smooth the normals of all meshes sharing that material. When off, the materials can be faceted and polygonal.

Index of Refraction (index)

Index of refraction sets the fresnel effect applied on the glossy material. Setting a value smaller than 1.0 will disable the fresnel effect, so the glossy color will be the color in the glossy input pin, regardless of viewing angle. When selecting a value of 1.0 or bigger, the glossy reflection color will be modulated according to the fresnel law: at grazing angles the color will be the color set in the glossy input pin, at perpendicular angles it will be darker. Fresnel reflection produced becomes stronger as the index of refraction is set higher. If you have a measured index of refraction, set the glossy color to 1.0.

In the following examples, the six balls have roughness 0, 0.2, 0.4, 0.6, 0.8, 1.0 (left to right) and only the specular value and index of refraction have been modified for each rendered image:

	specular = 1.0 index = 2.0 (strong Fresnel reflection)
	specular = 0.3 index = 3.0 (very strong Fresnel reflection)
	specular = 1.0 index = 1.5 (normal Fresnel reflection)
	specular = 0.3 index = 0.0 (no Fresnel reflection)
	specular = 1.0 index = 0.0 (no Fresnel reflection)

Specular Material

Specular(Glass) materials have these parameters to adjust:

Reflection (reflection)

The Reflection value determines the glossiness of the mesh.

Transmission (transmission)

The Transmission value gives the base color to the mesh.

Index of Refraction (index)

The Index represents the Index of Refraction of the material. Standard values of Index of Refraction (IOR) can be readily found via searching the internet. Glass typically has a value of 1.53 and water 1.33.

Film Width (filmwidth)

This controls the thickness of a optical, thin film on the material. This is useful in creating rainbow or oil slick effects.

Film Index (filmindex)

This controls the Index of Refraction of the thin film.

Bump / Normal

Both the Bump and Normal channels can load images to control the amount of bump mapping and normal mapping (respectively.) The Bump channel should be set to floatimage to load a bump map. The Normal channel should be set to the image data type to load a full color normal map.

Opacity (opacity)

Opacity sets the transparency of the material. Set the data type to `alphaimage`(if the image has an alpha channel) or `floatimage`(for black/white images) to load an image to set the transparency (use the Invert checkbox if necessary to adjust whether black or white regions are considered transparent).

Normal Smoothing (smooth)

Normal Smoothing is a Boolean value that sets whether to smooth the normals of all meshes sharing that material. When off, the materials can be faceted and polygonal.

Dispersion Coefficient (dispersion_coefficient_B)

The dispersion in OctaneRender™ is based on Cauchy's equation which has two terms: A which is the index of refraction and B which is the dispersion coefficient. Increasing the value increases the amount of coloration and dispersion in the object and in caustics.

Absorption or Scattering Medium (medium)

Absorption Medium

Absorption means that the material slightly absorbs light while passing through. The color resulting from this absorption is dependent on the distance light travels through the material. With increased distance it will get darker, and if the absorption is colored it will get more saturated.

Scattering Medium

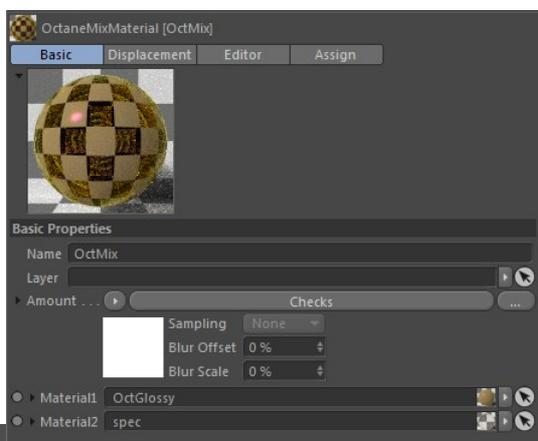
Similar to the absorption medium but with the option to simulate subsubsurface scattering. The scale parameter multiplies the absorption texture, allowing a wide range of values to be set more easily. For applying absorption, SubSurface Scattering and emission on Specular and Glossy Materials, refer to Chapter 5 Section 5.9.

Fake Shadows

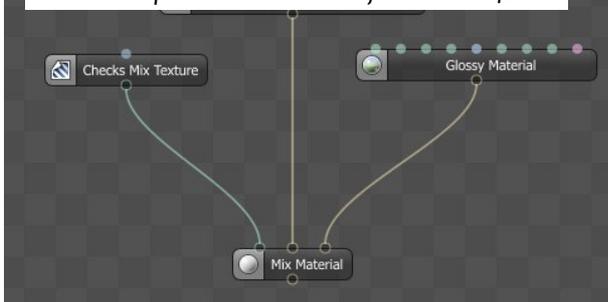
Fake Shadows is a Boolean value that sets the architectural glass option for all meshes sharing that material. This setting is off by default. When enabled, the specular material exhibits the characteristics of Architectural glass with its transparent feature allowing light to illuminate enclosed spaces or frame an exterior view.

Mix Material

The Material Mix option is used to combine two different materials. It accepts any two material nodes and the mix is controlled by a texture node. In the example below, a white glossy material is mixed with a red specular material. A checks texture node is used to control the mixing of the two materials



Similar setup in Octane Render for Cinema 4D



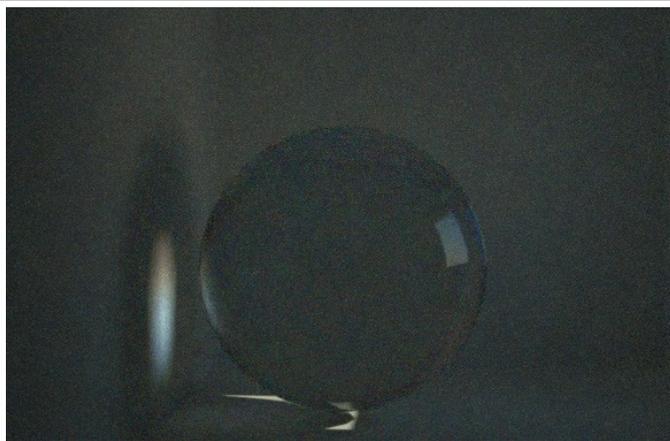
A simple Material Mix Set up from Octane Standalone



A mixmaterial from LiveDB, Nickel Aged_Brass

Portal Material

A portal is a special material applied to a plane to allow the render kernel to efficiently find openings / windows in interior renders. The use of portals will be covered in the next section covering lighting options.



A scene with no portal over a small window lighting a room with a glass sphere. The scene was rendered for 500 samples and is overall noisy.



The same scene with a plane over the window with a Portal material applied. Rendered at 500 samples, the image is cleaner.

Emissions

Blackbody Emission

The Blackbody Emission type uses Colour Temperature (in Kelvin) and Power to control the colour and intensity of the light.

Light source sampling rates – in earlier versions of OctaneRender™ light sources were sampled according to area. In some scenes this will lead to a poor balance between light sources. In a room with a big TV (which emits some light) and a couple of small bright spotlights, the TV would have been sampled most of the time, leading to a lot of noise in the light from the spotlights. Emission nodes have a parameter called `sampling_rate`, to control how much weight is given to the emitter when picking an emitter to sample. This allows you to choose which light sources will receive more samples.

Temperature

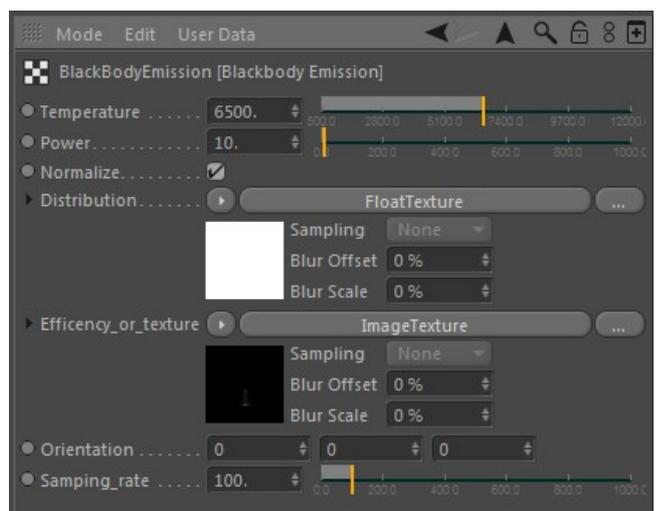
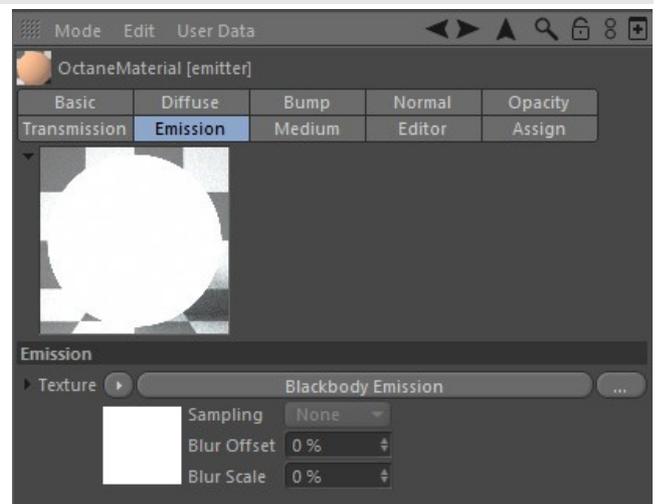
The temperature (in K) of the light emission.

Power

This is the wattage of the light source. Each light in the scene should be set to its real world wattage. For example, a desk lamp could be set to 25 watts, a ceiling lamp to 100 watts, and an LED light to 0.25 watts. This setting should not be used to balance the lighting power of the scene.

Efficiency or Texture

This setting is used to set the efficiency of the light source. No light is 100% efficient at delivering the power at the specified wattage (a 100 watt light bulb does not actually deliver 100 watts of light.) The efficiency setting can be used to enter the



real world values. These values can be used to create very realistic light settings. For example, a standard 100 watt incandescent bulb would only be approximately 2.0% efficient where as a 25 watt compact fluorescent light will be 10% efficient. These will both produce around the same quantity of light in real life.

Distribution

The distribution controls the pattern of the light. This can be set to a floattexture and an image or IES file can be loaded.



Blackbody emission sample

Texture Emission

This allows any valid texture type to set the light intensity. This can be used to create neat effects such as TV screens by using an Image Texture as the source.

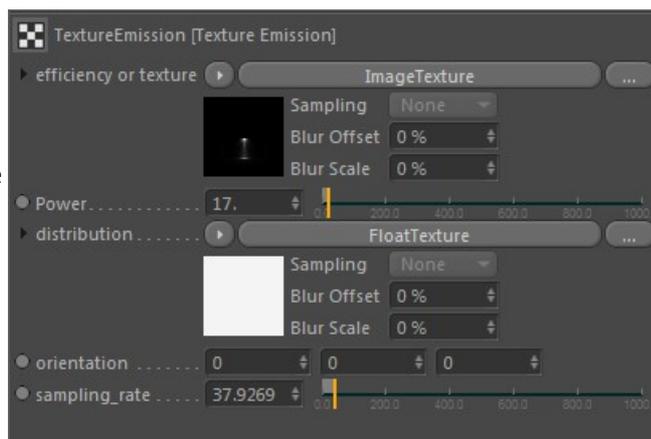
Light source sampling rates – in earlier versions of OctaneRender™ light sources were sampled according to area. In some scenes this will lead to a poor balance between light sources. In a room with a big TV (which emits some light) and a couple of small bright spotlights, the TV would have been sampled most of the time, leading to a lot of noise in the light from the spotlights. Emission nodes have a parameter called `sampling_rate`, to control how much weight is given to the emitter when picking an emitter to sample. This allows you to choose which light sources will receive more samples. In the above example you can increase the rate for the spotlights, so they get sampled adequately.

The following settings are listed for the Blackbody emission settings, but many of the settings are shared with the Texture emission settings. The main difference is whether the colour comes from the Blackbody temperature or from the texture settings of the diffuse material.

Temperature – The temperature (in K) of the light emission.
Power – This is the wattage of the light source. Each light in the scene should be set to its real world wattage. For example, a desk lamp could be set to 25 watts, a ceiling lamp to 100 watts, and an LED light to 0.25 watts. This setting should not be used to balance the lighting power of the scene.

Efficiency or Texture – This setting is used to set the efficiency of the light source. No light is 100% efficient at delivering the power at the specified wattage (a 100 watt light bulb does not actually deliver 100 watts of light.) The efficiency setting can be used to enter the real world values. These values can be used to create very realistic light settings. For example, a standard 100 watt incandescent bulb would only be approximately 2.0% efficient where as a 25 watt compact fluorescent light will be 10% efficient. These will both produce around the same quantity of light in real life.

Distribution - The distribution controls the pattern of the light. This can be set to a floattexture and an image or IES file can be loaded.



Using Medium Nodes

OctaneRender™ supports participating media inside objects (absorption, SubSurface Scattering and emission). These settings are stored in medium nodes, which are attached to the corresponding input pin of diffuse or specular material nodes.

There are two types of medium nodes, **absorption** and **scattering**. Scattering has parameters for absorption and scattering of light passing through the medium, and emission inside the medium. Absorption is a simple version with only absorption.

Rendering a medium requires the path tracing or PMC kernel, with a sufficiently large maxdepth setting. For media inside diffuse transmitting materials the direct light kernel with Diffuse (4) GI mode can be used as well.

Absorption Medium

Absorption means that the material slightly absorbs light while passing through. The color resulting from this absorption is dependent on the distance light travels through the material. With increased distance it will get lighter, and if the absorption is colored it will get more desaturated. It works in a subtractive way, so you need to configure the inverted colour instead to get an absorption colour desired.

Scattering Medium

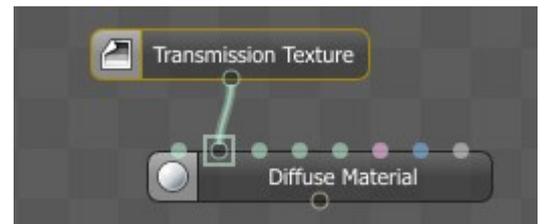
This is a medium with single-scattering SSS, and also absorption. To use this medium you need to create a volume, it will not work correctly on simple surfaces. You can use it to create true unbiased SSS (although not multiple scattering yet), using various parameters, including scattering texture, emission texture and various other parameters. Note that this is single-scattering SSS, not multiple-scattering. It's much faster than the latter, and much more practical, although it does not allow a few things such as volumetric caustics.

Meshes

Participating media should only be added to meshes that define a closed volume, rather than planes. Using planes to model leaves of a plant with SSS will for instance not work. Using a single plane as ground plane should be OK (it will be treated as an infinitely deep material). The mesh can have opaque objects nested inside, but nested participating media are not supported.

How to make a medium node work:

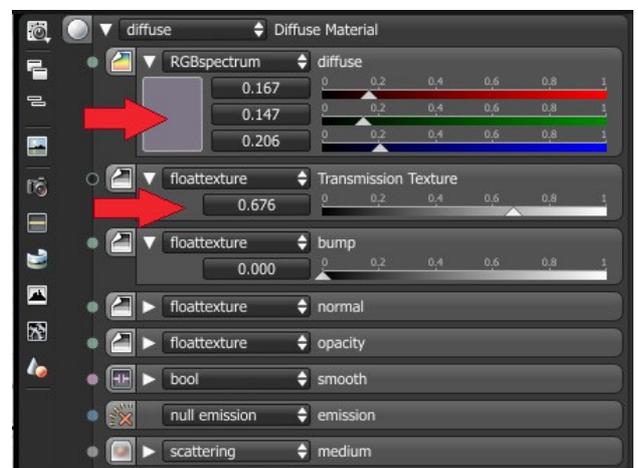
A specular material is the easiest: by default it is set up properly. Transmission must be nonzero; for the best effect reflection and transmission should be set to 1.0. On diffuse materials subsurface scattering only works after they are set up with diffuse transmission. To do so, connect a node to the transmission input pin.



The reflection value needs to be set to a sufficiently low value, as only the part of the spectrum that is not reflected can enter the inside of the object. If the reflection is set to 1.0, all light gets reflected regardless of the transmission value. If set to 0.0, all light gets transmitted, but this gives an unnatural appearance. Values of 0.1–0.2 are a good starting point.



Also, if the reflection is coloured, the transmitted light will have the complementary color (e.g. if the reflection is set to yellow, the transmitted light is blueish).

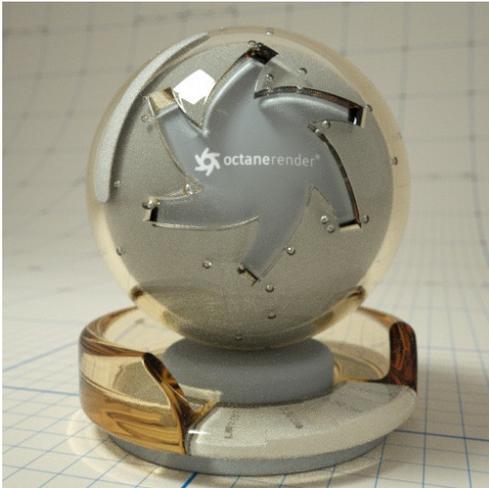


The transmission value works the same as in the previous releases, and is multiplied with the complement of the reflection spectrum. This should be set to a high value. If possible, use a diffuse transmitting material

Medium node parameters

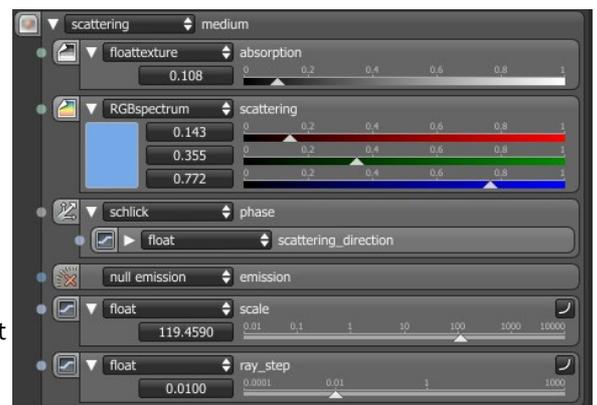
Absorption is controlled with the absorption texture, which defines how fast light is absorbed while passing through a medium.

A setting of 0.0 means no absorption. The higher the value the faster light is absorbed by the medium. This setting is wavelength-dependent, in the following setup for example it is set to absorb blue light faster than other colors, giving the object a yellow appearance. The absorption texture is multiplied with the scale parameter. This allows setting a wide range of values more easily.



Subsurface scattering

is controlled by the scatter texture (multiplied by the scale value), and the phase function. The scattering texture defines how fast light gets scattered when traveling through the medium, in a similar way to how absorption is defined. A very high value means light gets scattered very fast, a value of 0.0 means no scattering. The phase function controls in what direction light gets scattered. There is currently one kind of phase function defined, with one parameter, scattering_direction. If set to 0.0, it means light gets scattered the same amount in any direction. A setting larger than 0.0 means forward scattering, the larger the value, the more light is scattered in a similar direction as it was traveling. 1.0 means it doesn't change direction. A negative value means backscattering, so more light is scattered back to where it came from.



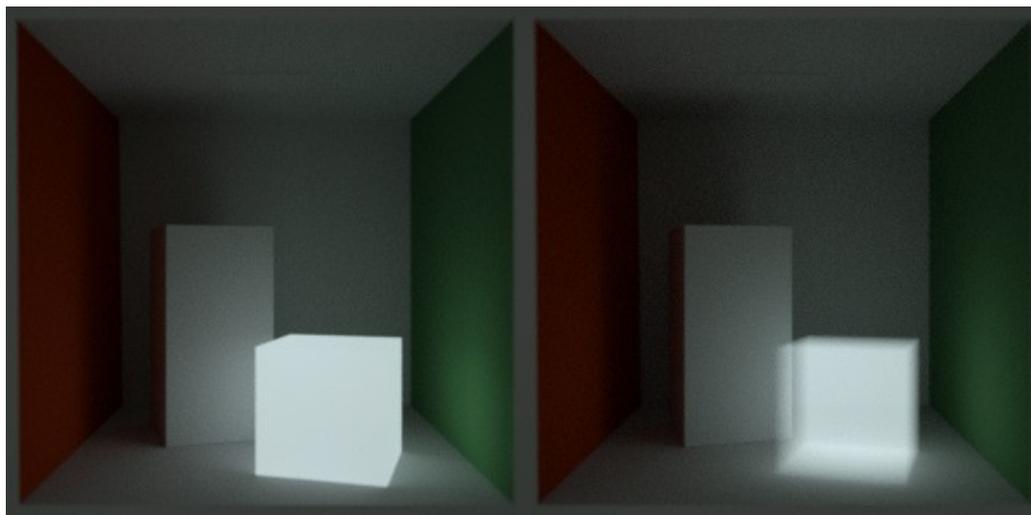
Setup which's in picture uses a wavelength dependent scattering. Blue light is scattered more, so it has more tendency to scatter back out of the material before it gets absorbed. Yellow-red light is scattered less, so more of it goes straight through the material, giving shadowed areas the yellow color.



Scattering inside a specular material will introduce a lot of noise. It will work best with smooth illumination. Finally emission is defined by attaching an emission node to the emission input pin. When connecting an emission node to a medium node, it defines emission inside the volume instead of at the surface of the object. The power parameter gets a different meaning; it controls how fast the radiance along a ray increases while traveling through the volume, and not the total power. It is not multiplied with the scale parameter. This effect works best with large and not too bright objects, very small bright objects will create a lot of noise.

The following image shows a normal mesh emitter (left) and a volume emission (right). The emitter in the right side has specular material with an index of refraction of 1.0 (making it invisible without the emission).

The stepsize parameter allows lowering the average distance between points where Octane samples the absorption, scattering and emission textures. If these textures are not constant, lowering this parameter may improve the appearance of the material, but if set too low the ray tracing may terminate before the ray is traced back out of the material. The priority input is currently not used.



Textures

Textures are used to create and manipulate the colour and appearance of the actual material.

Texture Types

The Texture types allow for creating very flexible materials. There are four main types of Texture Node Types

Colours: Colour textures provide methods of applying a colour to a material.

Generators: Generator textures are used to actually create or modulate other textures. They are grey scale.

Images: Image textures use image files to create the texture.

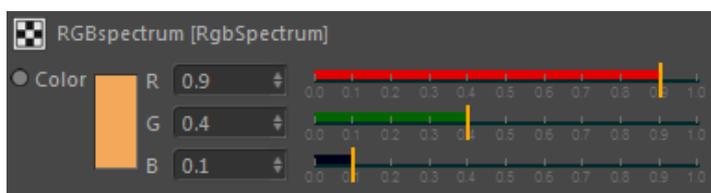
Mappings: Mappings are Nodes that allow mixing and manipulation of other texture nodes.

Colour Textures

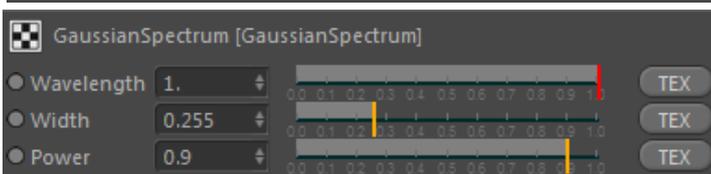
Float: The texture is based on float value.



RGBspectrum: The texture is based on the RGB colour that is selected.



GaussianSpectrum: The texture is based on a Gaussian distribution spectrum. The wavelength is used to set the center of the spectrum and the width is used to set how



wide the curve is. The narrower the width, the more pure and saturated the colour.

Image Textures

ImageTexture

In order to properly utilize Image Textures, the mesh must be UV mapped prior to export from the Modelling software.

Power: A kind of output multiplier for texture output.

Any texture can be assigned to this parameter

Gamma: Controls the Gamma of the image file

Invert: Inverts color output values of image

Border mode:

Type parameter has Normal(colored), Float and Alpha options. It defines image type.

FloatImage: The image is interpreted as gray scale even if it is a full colour image, thus saving GPU ram. The Invert checkbox can be used to invert the image (useful for bump and opacity maps.)

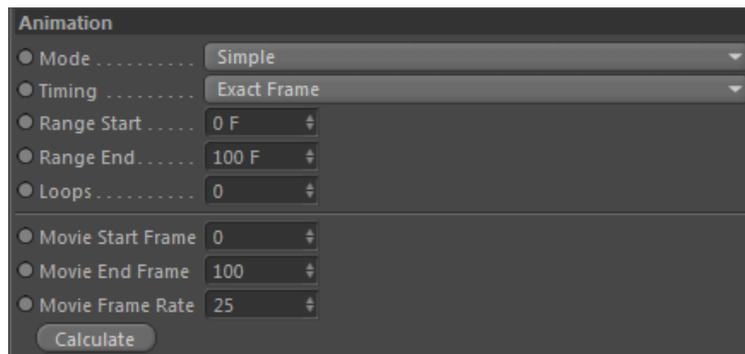
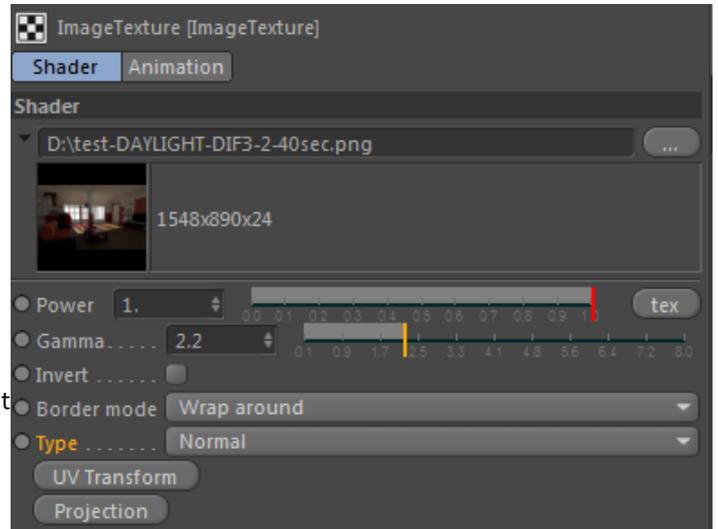
Normal: An image texture is used for the parameter (mesh must be UV mapped prior to export to properly use the Image type). The image is interpreted as being full colour even if it is a grey scale image, and therefore taking up more GPU memory.

AlphaImage: An alpha image utilizes the images native Alpha Channel to provide transparency. This type will only accept PNG and TIF image types.

For additional info please refer to Transform and Projection sections

Image sequence for ImageTexture

Now imagetexture supports image sequences. Beginning and end ranges must be setted properly in **Animation** tab of imagetexture. Also standart bitmap type is working as imagetexture for image sequences.



When to use Floatimage versus Image Data Type?

There are some parameters where full colour data is not used (or useful). If a full colour image is loaded, it can take much more memory in the GPU than a grey scale image, even though the user only really wants the grey scale data.

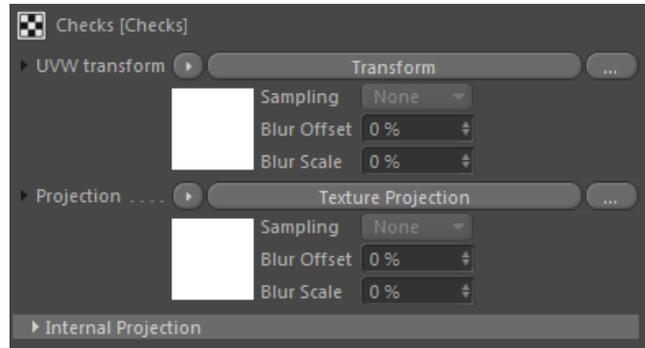
Since memory management is very critical for GPU rendering, the Floatimage type allows a user to load a full colour texture but it will be interpreted as a grey scale image and therefore use less video ram. If the full colour data is needed then use the Image data type (normal maps, diffuse maps)

Texture Generators

Texture Generators are used to create patterns that can be used alone or in combination with the Mapping and Colour textures to create memory-efficient, procedural textures. Procedural Textures can be used to create textures, bump maps and other advanced materials with minimal impact to GPU memory. It is therefore advantageous to explore creating materials using these textures before resorting to image based textures.

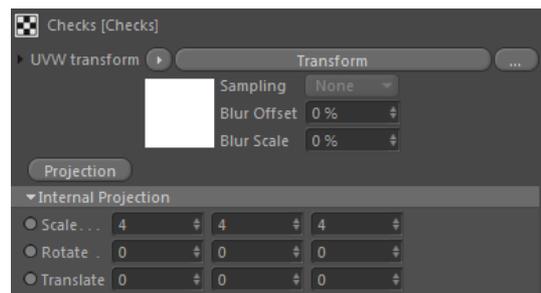
Check

Check procedural texture is useful for making stripes, checkerboard and grid patterns. It is controlled by a float3 value. It is most useful when mixed with other textures.



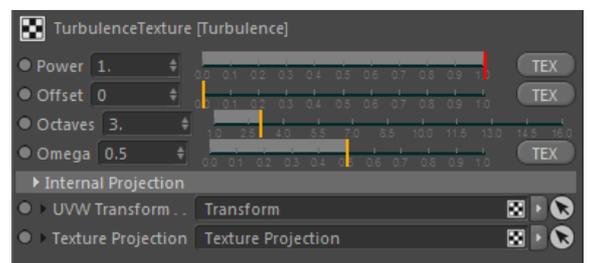
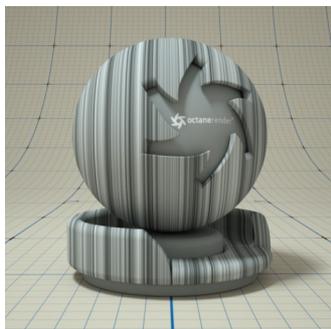
Sine Wave, Saw Wave and Triangle Wave

These textures can be used to create various banding effects.



Turbulence

The Turbulence texture can be used to create many different effects based on banded noise. This flexible texture can be used to create wood, marble, flesh, and many other useful textures.



Marble

The Marble texture is a procedural texture that can be used to create marble-like noise. It is similar to a turbulence texture but more fine-tuned to create marble-like patterns.

Ridged Fractal

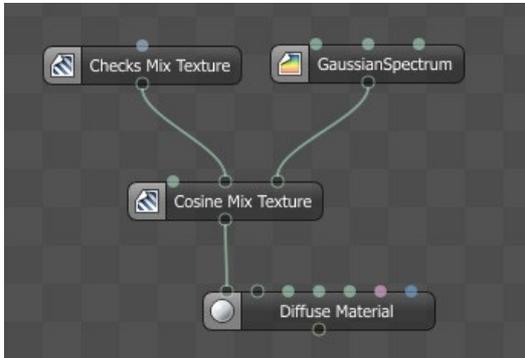
Mapping Texture Types

Clamp

The Clamp Texture requires a texture input and then allows for the texture to be “clamped” with the minimum and maximum slider.

Mix / Cosine Mix

The Mix textures are used for mixing two textures together (either linearly or according to a cosine wave). In the example below, a Checks Mix was combined with a Gaussian Spectrum using a Cosine Mix Texture and connected to the Diffuse channel of a Diffuse material.



Combining a Checks Texture with a Gaussian Spectrum by using a Cosine Mix Texture which is set to a Diffuse Channel of a Diffuse Material



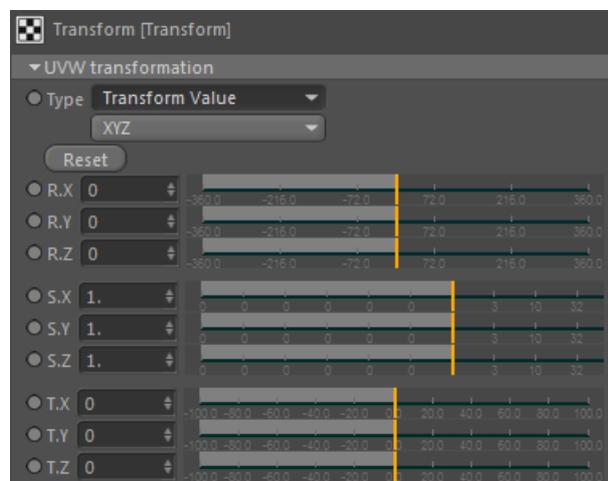
The resulting material

Fall Falloff Texture Map

The fall off map is a texture node typically used to control the blend of two materials depending on the viewing angle of the materials geometry.



Texture Transform



Texture Projection

Projection Types

Mesh UV

Use the mesh UV coordinates. Nothing changed here.

For environment textures and IES Light distributions this will apply a spherical mapping, but for more control over the projection (mainly rotation) you may switch to a spherical projection node in these cases.

XYZ to UVW

also known as: Planar, Flat mapping

Just take the coordinates in world or object space and use them as UVW coordinates. For images only the X and Y coordinates are relevant, which are mapped to U and V. In other words the images use flat mapping projected along the Z axis.

In 1.30 the projection will map image textures to the $(-1, -1)$ – $(1, 1)$ range. So rotating the mapping around the Z axis will rotate the image around the center, as the UVW rotation would do.

Box

also known as: Cube mapping (although some applications may have a different mapping called Cube mapping)

This is an extension of XYZ to UVW mapping, it will pick a different projection axis depending on the normal. This gives a quick way to map a texture on any object without too much distortion, but possibly with a lot of seams. The same change in mapping applies here: image textures are mapped to the $(-1, -1)$ – $(1, 1)$ range.

Spherical

Perform latitude-longitude mapping for the U and V coordinates, and for procedural textures the W coordinate is the distance from the origin. This is mostly used for environment textures and IES Light distributions. Nothing changed here.

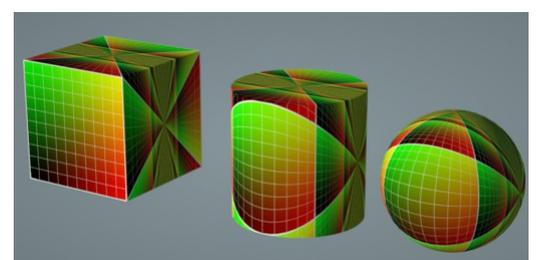
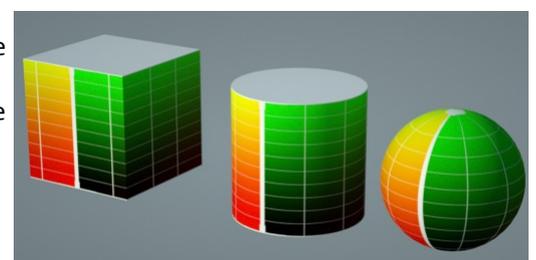
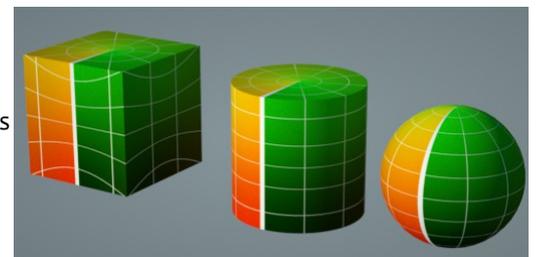
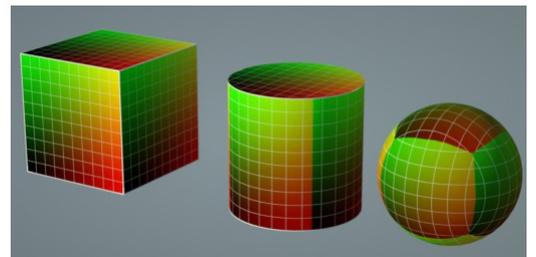
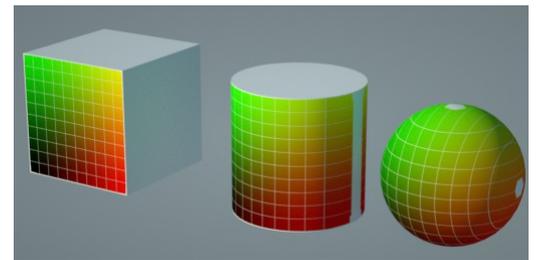
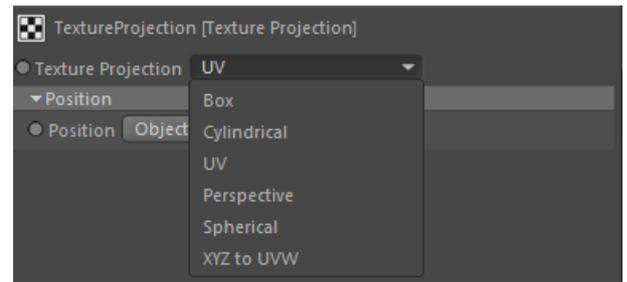
Cylindrical

This performs cylindrical mapping: the U coordinate is the longitude, the Y coordinate is the world space Y coordinate. For images the mapping on the Y axis has changed to map the image to the $[-1, 1]$ interval. For procedural textures the W coordinate is the distance from the Y axis. For points on the ground plane ($Y = 0$) cylindrical and spherical mapping now map to the same points on the images (what would be the equator on spherical mapping).

Perspective

This mapping takes the world space coordinates, and divides the X and Y coordinates by the Z coordinate. This can be useful if you want to model a projector (you use a texture with this projection as the distribution, with black border mode). It can also be used for camera mapping.

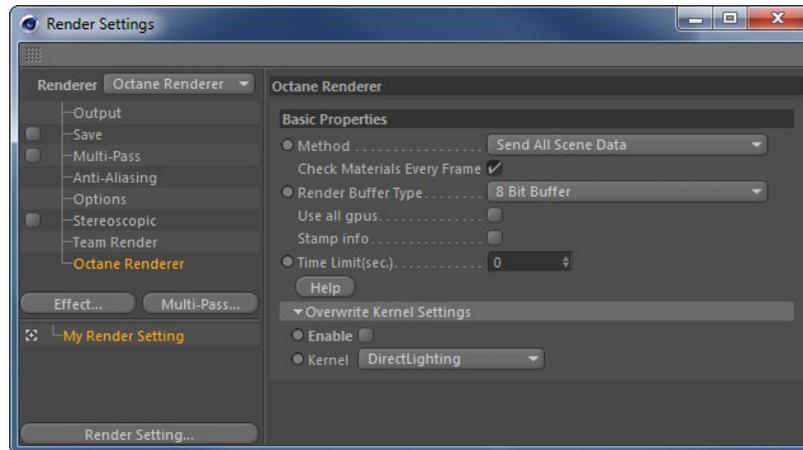
The same change as the other projections applies here: the image is mapped to $(-1, -1)$ – $(1, 1)$, so by default you don't need any offset to use this mapping for projectors or camera mapping.



Animation rendering

Setting up the Octane Render as Current Renderer for Animation Output

Select Octane Renderer from Renderer Control as shown in below picture.



Parameters:

Methods:

1. **Send All Scene Data:** This most reliable and slowest render sending method. Reset all data and completely send scene for each frame. Preparation times of frames should be identical on each frame.
2. **Resend Only Updated Objects:** This faster method for animation rendering. First frame, all scene data will be sent to Octane Render. Next frames, only updated parts of scene will be refreshed. For this method, user have to set which part of objects is movable. OctaneObjectTag is designed for this and more. Please look to usage of the tag.

Check Materials Every Frame

Plugin checks or not the materials and refresh for each frame. Option is effective for method2. Please try with scenes which has not animated materials.

Check Meshes Every Frame

Option is effective for method2 and on object groups which have OctaneObject

Update Method

1. **Faster rendering:** Option for faster rendering on complex scenes, but time of frame preparation can be longer. This is recommended method to ensure updates will be detected correctly. Works with method2.
2. **Faster updates:** Option for faster preparation times for each frames. Works with method2. This algorithm is same which is used for Live Viewer on realtime update detection. Sets the update engine to minimal data feeding state. This method creates separated meshes for objects which is under a octaneobjecttag(Mesh Nr:1-7). Use previous method if you are not sure about that all geometry will be correctly updated. *Note:Please try on scenes which has lot of objects with transform animations.*

Render Buffer

- **8 bit:** Colors are converted 8 bit RGBA channels
- **Float tonemapped:** Colors are direct output of Octane Render and applied tonemapping. This works well with 16/32 bit images(tiff,exr,etc..)
- **Float linear:** Colors are direct raw output of Octane Render without tonemapping. This works well with 16/32 bit images(tiff,exr,etc..)

Use all gpus: Activates all GPU's for rendering

Time Limit: It's maximum time(second) for rendering for each frames

Overwrite Kernel Settings

This section has all kernel settings to use and change during render output. Parameters has animation support. To change the any parameter for different frames is possible. This allow to change set maxsamples or kernel type or any parameter for different frames.

Multipass rendering

Plugin supports multipass rendering. To use multipass output, you must activate multipass option and select needed channels. Works as usual in standart rendering. Octane Render has naturally a few multipass channels(zbuffer/normals/material color,alpha,...) at this moment. Other channels(motion, objectid, shadow, reflection,etc..) will be generated by Standart Renderer of Cinema 4D. You should be aware of some possible inconsistency. Ex. DOF or some transparency can causes this.

Octane Render 2.0 will have native render passes with a update.

Motion Blur

Octane Render 2.0 has motion blur effect for camera, object transform and object deformations.

Main control is on Octane camera tag. Motion blur section has **shutter time**, **time shift** and **alignment** parameters. You can enable/disable the motion blur effect from **enable** control.

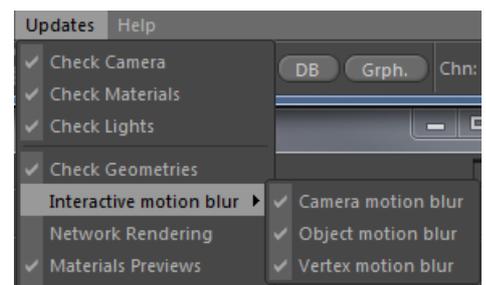
Motion blur for objects can be activated by Octane object tag. As you know this tag allow to define movable objects. It defines each object will be exported individual meshes and will have a transform and visibility node. Now, we obtain the motion blur of objects by this tag. **Vertex motion** blur is a switch on tag. This activates deformation caches.



Live Viewer and Picture Viewer support motion blur.
Below image shows the required settings for object motion blur.

Live Viewer also has support for realtime motion blur. It's little tricky and will be improved.
It needs some caching of matrices and vertices to can produce the effect. So you should play or scrub the timeline toward to next frames.

Interactive motion blur menu has some switches. These are only for Live Viewer and not affect the Picture Viewer renderings.



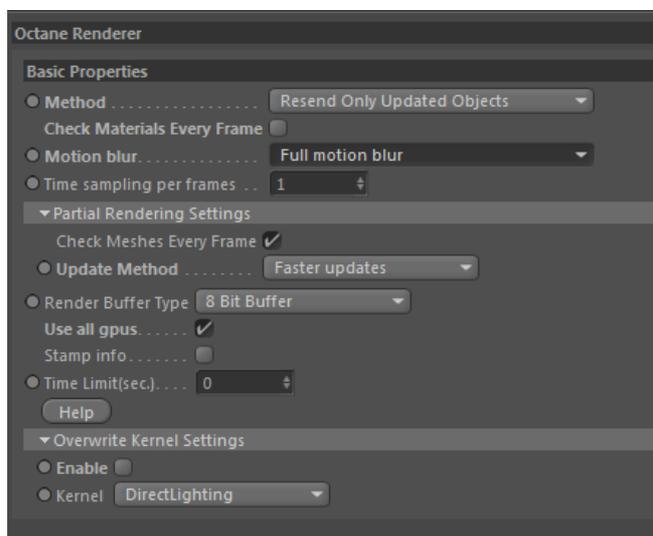
Required render settings for producing motion blur

Motion blur:

This control changes the required settings to default for producing the motion blur effect.

Time sampling per frames

Default value is 1. This control a parameter. It produces sub keyframes for objects. Value 4 means that will export 4 keyframes between two frames. It can be required for some objects which are rotation very fast.



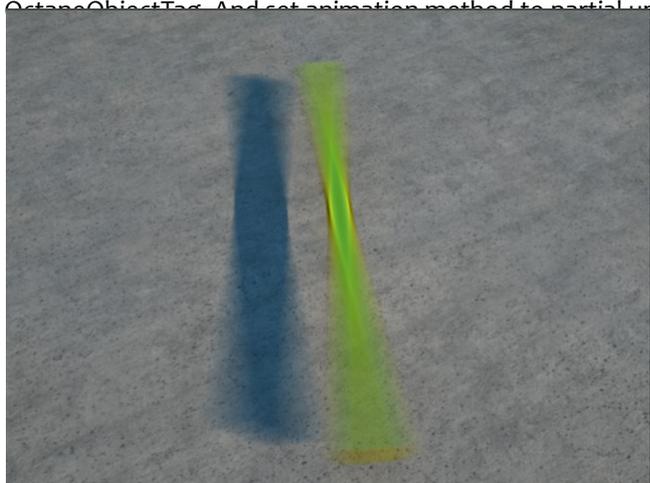
Sub-Frame Motion Blur

We have other option to generate some motion blurred images.

It can produce these images with Sub-Frame Motion Blur effect. It can be found from effects of Picture Viewer.

Camera shutter speed affects the duration of frames. Count of passes affects the output quality.

For lower render times, there is a nice trick. That is to make animated objects making movable geometry by `OctaneObjectTag`. And set animation method to partial updates from Octane Renderer settings.



shutter speed=1/125s
passes 36



shutter speed=1/30s
passes 36

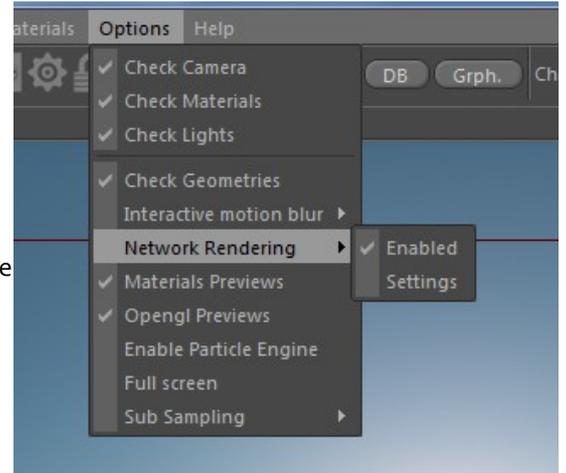
Network Rendering

Octane Native Distributed Rendering

It can be activated with **Network Rendering->Enabled** from Options menu in Live Viewer.

Overview

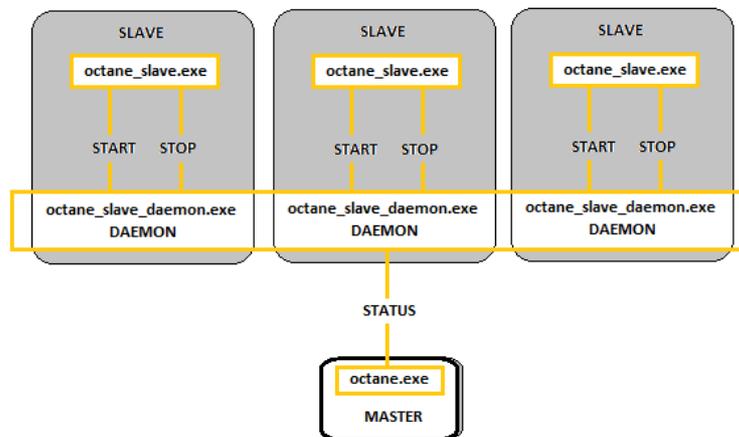
Network rendering allows additional computers to be utilized in rendering images. OctaneRender distributes compiled render data and not scene data, so no file management is required by the user. Conceptually it is similar to working with additional GPUs by allowing the distributed rendering of single images over multiple computers connected through a fast local area network. Network rendering requires a master and one or more slaves on different computers. The OctaneRender instance that drives the rendering is referred as the “master” and the OctaneRender instances that are helping are referred as the rendering “slaves”.



Since an OctaneRender slave currently requires an activated Standalone license, it is advisable to run the Standalone first to activate a Standalone license on that computer, if necessary. It is best to copy the whole folder of the released archive onto the slave computer. Also ensure that the master and the slave are not blocked by the Operating System firewall or any firewalls in the network. This can be done, for example, by turning off the firewall for home/work networks on the master. If that does not help, also try switching off the firewall on the slave computer for home/work networks.

Master, Slaves and Daemons

The Standalone version or the octane.exe act as master and a special console version of OctaneRender, octane_slave.exe, can run on other computers as slaves. Of course, they should be all on different computers or they would have to share the same GPUs. Running the slave on the same computer as the master is pointless.



The OctaneRender network render slave is fairly dumb and all the render data processing is actually done on the master side. The slave does not need to have a powerful CPU at all, but the slave is of course required to have enough memory (RAM) to store the render data plus some render results. The operating systems of the slaves can also be different since the communication between the machines is cross platform. No data is stored on the slaves' discs, it all happens 100% in memory.

Each time network rendering is required, the slave process has to be launched on the slave machines. The slave daemon makes the control of the slaves more practical, as slave daemon can be set up to be launched at every start up of the operating system of each machine in the network. The daemon is the little program that starts a slave process on the machine (on request by a master), monitors it and stops it (on request by a master). Monitoring means making sure that a running slave sends a regular “heartbeat” to the daemon and if that doesn't happen it will try to stop the slave gracefully and if that does not work, it kills the process. The daemon runs all the time and starts/stops a slave process if a master requests it. The daemon also listens for the “heartbeat” of the slave to check if the slave process is still running at all. This slave daemon eliminates the need to launch the slave process manually on each computer each time rendering is required on the slave.

Setting Up The Slave Daemon

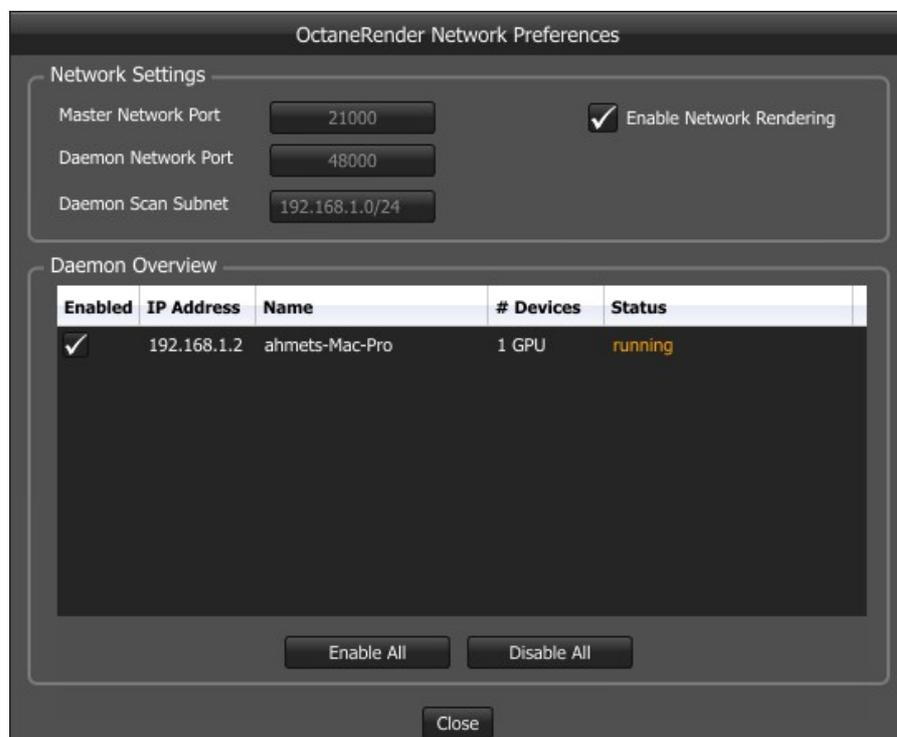
To set up the daemon, simply run the batch script `_install_daemon.bat` on the slave computer. During the setup, it will ask if the user wants to use all GPUs or only a sub-set and on which port the daemon should listen to for master requests. After that, the daemon will be resident on that machine and it will be active at all times.

When a daemon is invoked by a master, the slave is quickly launched to get some information about the number of GPUs, version, bitness, etc. and then closed again. After that there is no slave process running. So the daemon just sits there and waits for masters (there could be multiple masters in the local network) to detect it, by scanning the complete local network in regular intervals. The daemon should appear in the daemon list of the network preferences of the masters. If it does not, it can have the following reasons:

- The network rendering in the master is not enabled.
- The daemon is listening on a different port than the master is scanning. In your case both are 48000, so that's fine.
- The daemon is in a different sub-net than the master is scanning. If you have only one Ethernet adapter on the daemon and master PCs, you can safely ignore this case.
- The Windows firewall keeps the master from connecting to the daemon or the daemon from responding to the master. That's the most likely reason. To verify it, disable the firewall for private or home/work networks on both PCs. You have to close the Windows dialog containing the firewall options. Only then the change is actually applied. If the daemon is now detected (should take only up to 10-20 seconds), you can try enabling one firewall after the other to see which one is causing trouble. If you want to have the firewall running, you may have to poke a hole into it, to allow the communication between daemon and master.

Only when you enable a daemon in the master settings, the slave gets actually launched and will eventually appear in the status bar of the master. One daemon can be activated only by one master at a time. If daemon is currently "occupied" by another master the user will see the daemon state change accordingly.

You can open Network Preferences from, Live Viewer->Options->Network Rendering->Settings



Team render

This is easier to setup. Only need a client or other copy of Cinema 4D which is working on Client Machine. It should work smoothly if can work with Standart Renderer

SETUP FOR CINEMA 4D NATIVE NET RENDERING

NET client wants to read preferences file of Octane Render

You should copy oct.dat file from c:\Users\USERNAME\AppData\Roaming\MAXON\Cinema4Dr15_ABCDEFGH directory of Cinema 4D

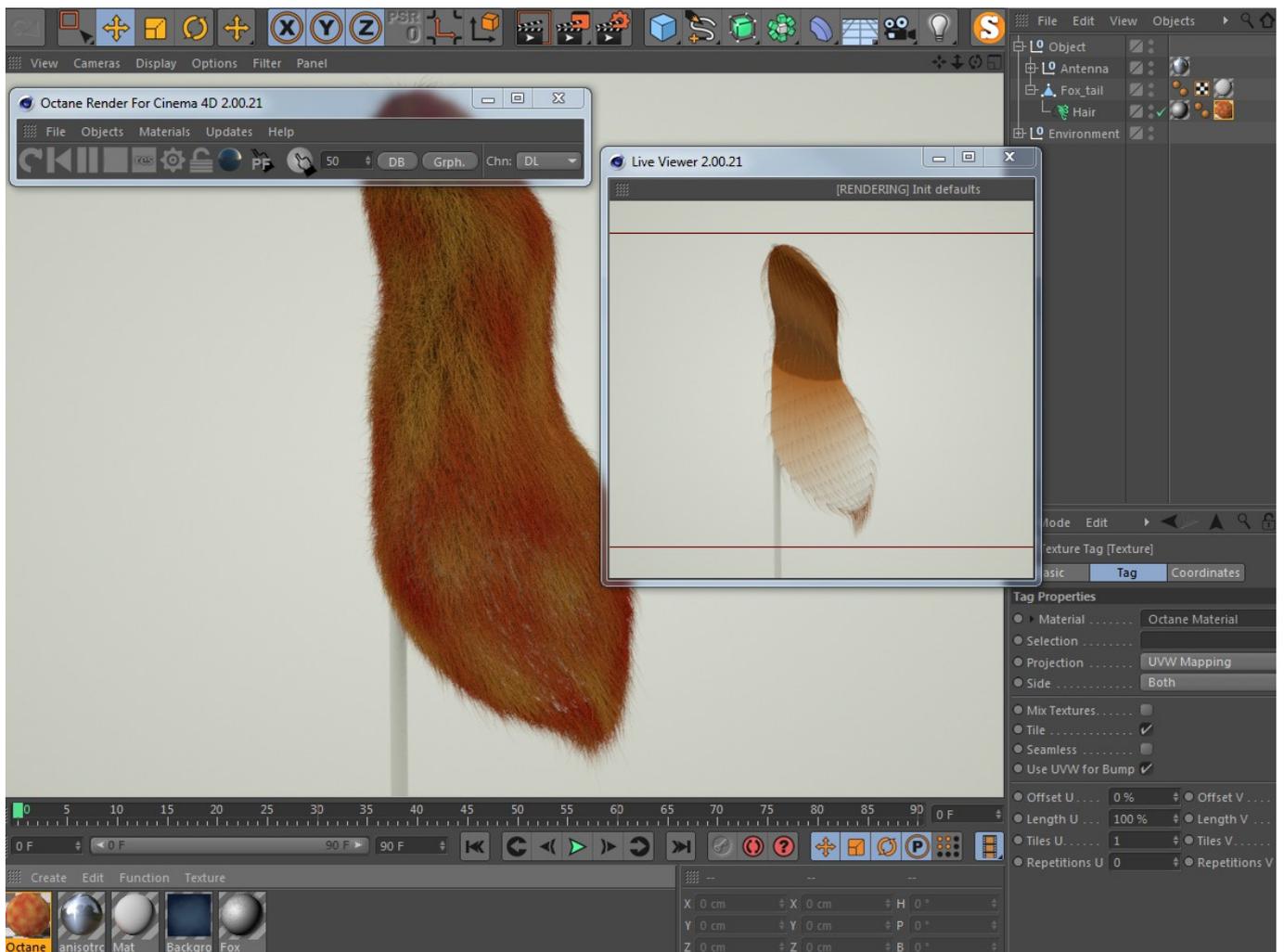
to c:\Users\USERNAME\AppData\Roaming\MAXON\Cinema4Dr15_ABCDEFGH\
Also relative texture maps shall be in a texture path of Octane Settings

Hair Rendering

Octane Render 2.0 has support for direct rendering of hair geometries without converting to polygons. Octane Render has a new hair engine which is optimised for hair rendering.

It exports only vertices of per hair strands and diameter of each vertex. Renderer generates faces on runtime in render sessions. This requires less memory and export times for preparation and rendering.

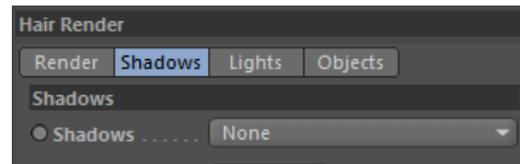
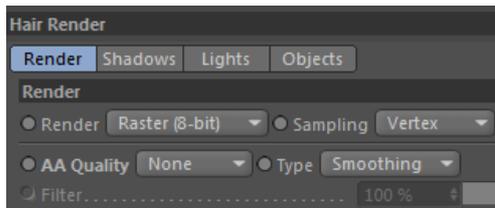
Implementation of hairs in plugin is still in beta period. There is some needs to improve it. UV mapping and motion blur is still in implementation list.



Some words for usage:

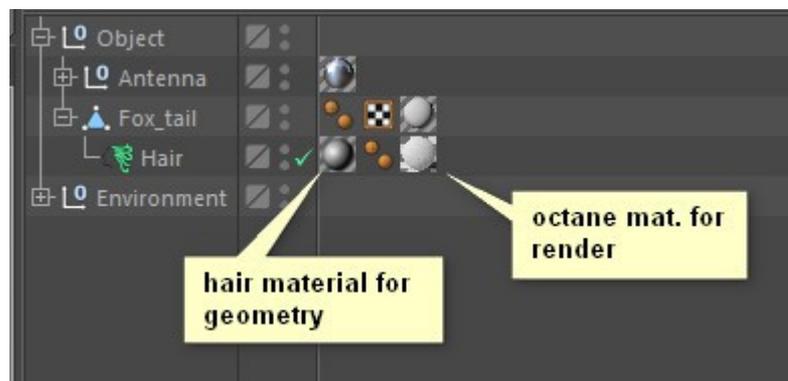
It's little tricky. It uses hair guides for Live Viewer and final output is available in Picture Rendering. You can think that LV behaves as a viewport for hairs. Another limitation is that Octane Renderer should be located after Hair Render videopost effect. Otherwise, you couldn't get geometry of generated hairs.

Also set the shadow of Hair Render to None. And AA quality to None. This allows to access hair data more quickly. These options will help to lower preparation times.



Materials:

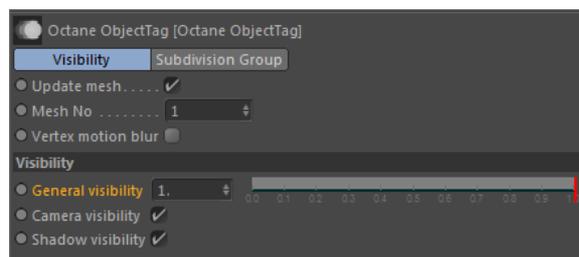
Plugin converts hair materials to Octane Glossy material. This conversion is on a very basic stage. It is required for the usage of hair deformation effects of Cinema 4D. Also, usage of another Octane Material on a Hair object is possible. Please refer to the sample hair scene for Octane Render for Cinema 4D.



Usage of Octane material on Hair Objects

OctaneObject Tag

This tag is designed for real-time geometry updates while moving or editing any objects. A second usage is that making faster renderings on scenes which have a lot of static geometry and some moving objects. Objects which are under this tag will be exported as individual meshes with a transform node. This is required for object transform motion blur and transform-based texture projections. Object motion blur is a feature of Octane Render 2.0.



Mesh No: Object group number for exporting to Octane Render.

Vertex motion blur: Option for activating vertex motion blur effect on attached objects.

Visibility parameters

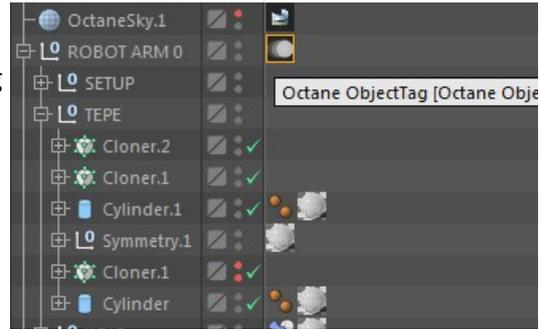
1. **General visibility:** Option for making the attached objects visible/unvisible to the renderer. It works as opacity for each object.
2. **Camera visibility:** Option for making the attached objects visible/unvisible to the camera. It does not affect the reflection/refraction/shadow visibility.
3. **Shadow visibility:** Option for making the attached objects visible/unvisible to shadow rays. This option works only with alpha shadows in the current kernel.

You should add this tag to any object or group to see updates in Live Viewer. But updates for geometry in Live Viewer should be selected.

Putting a tag to the top of any group is enough for assigning to all child objects. Also, this tag is useful for incremental

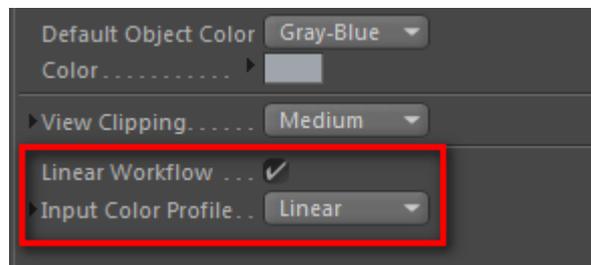
rendering. If you apply a tag any animated group with numbered 1-8, and leave all other static geometry as well as, then you can do animation rendering with very low export times. Please check the render settings section for usage in animation rendering.

Proper setup for a object group for realtime updates for Live Viewer and incremental updates for Animation Rendering



Working with color settings (Colorspace)

Input color profile may be Linear or disabled is best option.



Please check this link:

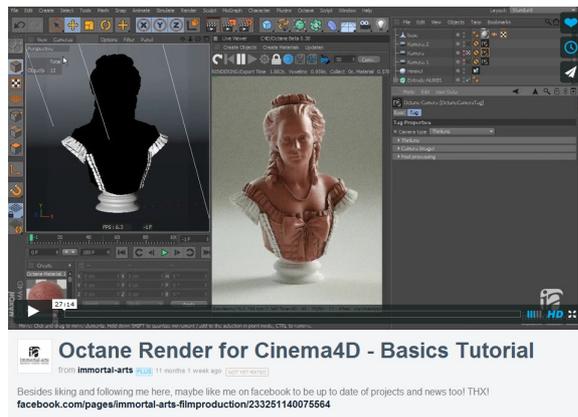
Linear workflow in Octane: <http://render.otoy.com/forum/viewtopic.php?f=21&t=33214>

Tutorials





Cinema 4D / Octane Renderer Basics Tutorial Part I



Octane Render for Cinema4D - Basics Tutorial

[Octane Render For Cinema 4D Tutorials On Vimeo](#)

[Octane Render For Cinema 4D Videos Group on Vimeo](#)

<https://vimeo.com/groups/189033>

<https://vimeo.com/84737320>

<https://vimeo.com/groups/189033/videos/62241847>

<https://vimeo.com/groups/189033/videos/65973357>

<https://vimeo.com/groups/189033/videos/65973358>

<https://vimeo.com/76825482>

<https://vimeo.com/groups/209276/videos/75539091>

<https://vimeo.com/groups/209276/videos/75318077>

Updates and Samples

Please check this forum for updates and sample scenes

<http://render.otoy.com/forum/viewforum.php?f=30>