

Skip to content

Color Management

Color management is important to create renders and assets that are physically accurate and look great on multiple display devices. It is used both to ensure all parts of the pipeline interpret colors correctly, and to make artistic changes like exposure and color grading.

Blender's color management is based on the [OpenColorIO](#) library. By using the same OpenColorIO configuration in multiple applications, the same color spaces and transforms will be available for consistent results.

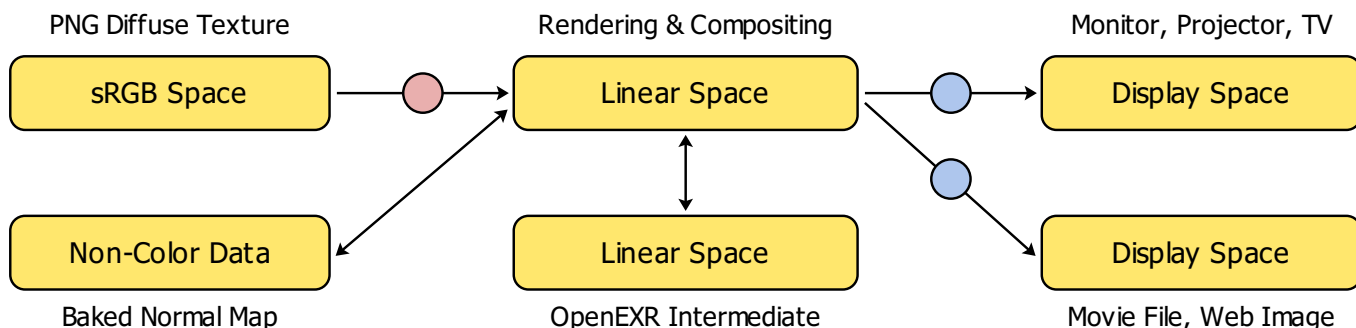
Workflow

Scene Linear Color Space

For correct results, different [Color Spaces](#) are needed for rendering, display and storage of images. Rendering and compositing is best done in *scene linear* color space, which corresponds more closely to nature, and makes computations more physically accurate.



Different views and exposures of the same render.



An example of a linear workflow.

If the colors are linear, it means that if in reality, we double the number of photons, the color values are also doubled. Put another way, if we have two photos/renders each with one of two lights on, and add those images together, the result would be the same as a render/photo with both lights on. It follows that such a radiometrically linear space is best for photorealistic rendering and compositing.

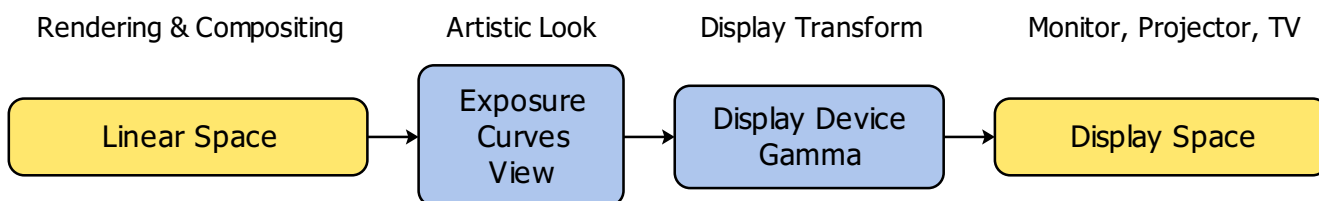
However, these values do not directly correspond to human perception or the way display devices work. and image files are often stored in different color spaces. So we have to take care to do the right conversion into and out of this scene linear color space.

Display Transforms

Transforming scene linear colors to display involves both technical and artistic choices.

Correct display of renders requires a conversion to the display device color space. A computer monitor works differently from a digital cinema projector or HDTV, and so needs a different conversion.

There is also an artistic choice to be made. Partially that is because display devices cannot display the full spectrum of colors and only have limited brightness, so we can squeeze the colors to fit in the gamut of the device. Besides that, it can also be useful to give the renders a particular look, e.g. as if they have been printed on real camera film. The default Filmic transform does this.



Conversion from linear to display device space

Image Color Spaces

When loading and saving media formats it is important to have color management in mind. File formats such as PNG or JPEG will typically store colors in a color space ready for display, not in a linear space. When they are used as textures in renders, they need to be converted to linear first, and when saving renders for display on the web, they also need to be converted to a display space.

For intermediate files in production, it is recommended to use *OpenEXR* files. These are always stored in scene linear color spaces, without any data loss. That makes them suitable to store renders that can later be composited, color graded and converted to different output formats.

Images can also contain data that is not actually a color. For example normal or displacement maps merely contain vectors and offsets. Such images should be marked as *Non-Color Data* so that no color space conversion happens on them.

Render Settings

Reference
Editor: Properties
Panel: Render Properties ▶ Color Management

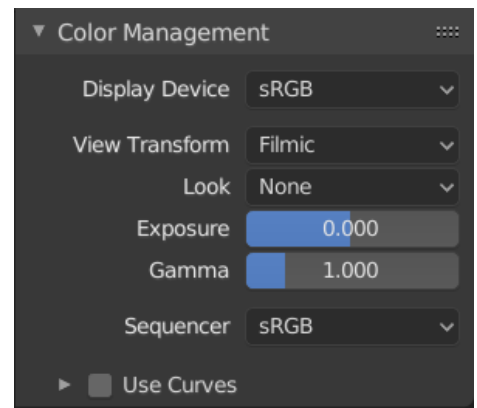
These are color management settings that are used across Blender. These color management settings are Scene specific so settings can be customized per Scene. Color management can also be overridden when saving images; this behavior can be set in the [Output Color Management Properties](#).

Display Device

The color space for the display that Blender is being viewed on.

Most displays are sRGB by default with some newer displays having the option to use Rec. 2020. These displays have a wider color gamut and can display high dynamic range content. If you have an Apple display you probably will want to use Display P3.

It is important to check your OS and display setting to make sure they all match the display in use to view the most accurate image.



Color Management properties.

sRGB:

Used by most displays.

Display P3:

Used by most Apple devices.

Rec. 1886:

Used by many older TVs.

Rec. 2020:

Used for newer wide gamut HDR displays.

View Transform

These are different ways to view the image on the same display device.

Standard:

Does no extra conversion besides the conversion for the display device. Often used for non-photorealistic results or video editing where a specific look is already baked into the input video.

Khronos PBR Neutral:

A tone mapping transform designed specifically for PBR color accuracy, to get sRGB colors in the output render that match as faithfully as possible the input sRGB base color in materials, under gray-scale lighting. This is aimed toward product photography use cases, where the scene is well-exposed and HDR color values are mostly restricted to small specular highlights.

AgX:

A tone mapping transform that improves on *Filmic*, giving more photorealistic results. AgX offers 16.5 stops of dynamic range and desaturates highly exposed colors to mimic film's natural response to light.

Filmic:

A tone mapping transform designed to handle high dynamic range colors. Filmic is deprecated and is superseded by AgX which improves handling of saturated colors.

Filmic Log:

Converts to Filmic log color space. This can be used for export to color grading applications, or to inspect the image by flattening out very dark and light areas.

False Color:

Shows a heat map of image intensities, to visualize the dynamic range, and help properly expose an image.

Below is a table that represents how normalized linear color data is represented with False Color.

Luminance Value	Color
Low Clip	Black
0.0001% to 0.05%	Blue
0.05% to 0.5%	Blue-Cyan
0.5% to 5%	Cyan
5% to 16%	Green-Cyan
16% to 22%	Gray
22% to 35%	Green- Yellow
35% to 55%	Yellow
55% to 80%	Orange
80% to 97%	Red
High Clip	White

Raw:

Intended for inspecting the image but not for final export. Raw gives the image without any color space conversion.

Look

Choose an artistic effect from a set of measured film response data which roughly emulates the look of certain film types. Applied before color space conversion.

Exposure

Used to control the image brightness (in stops) applied before color space conversion. It is calculated as follows: $\text{output_value} = \text{render_value} \times 2^{\text{exposure}}$

Gamma

Extra gamma correction applied after color space conversion. Note that the default display transforms already perform the appropriate conversion so this mainly acts as an additional effect for artistic tweaks.

Sequencer

The color space that the [Sequencer](#) operates in. By default, the Sequencer operates in sRGB space, but it can also be set to work in Linear space like the Compositing nodes, or another color space. Different color spaces will give different results for color correction, crossfades, and other operations.

The list of color spaces depends on the active [OCIO config](#). The default supported color spaces are described in detail here: [Default OpenColorI Configuration](#)

Display

High Dynamic Range

Enable high dynamic range display in rendered viewport, uncapping display brightness. This requires a monitor with HDR support and a view transform designed for HDR (Filmic does not generate HDR colors).

This feature is currently only supported on macOS.

Use Curves

Adjust RGB Curves to control the image colors before the color space conversion. Read more about using the [Curve Widget](#).

White Balance

Adjusts colors so that a given white point (expressed in color temperature and tint) ends up as white on the display.

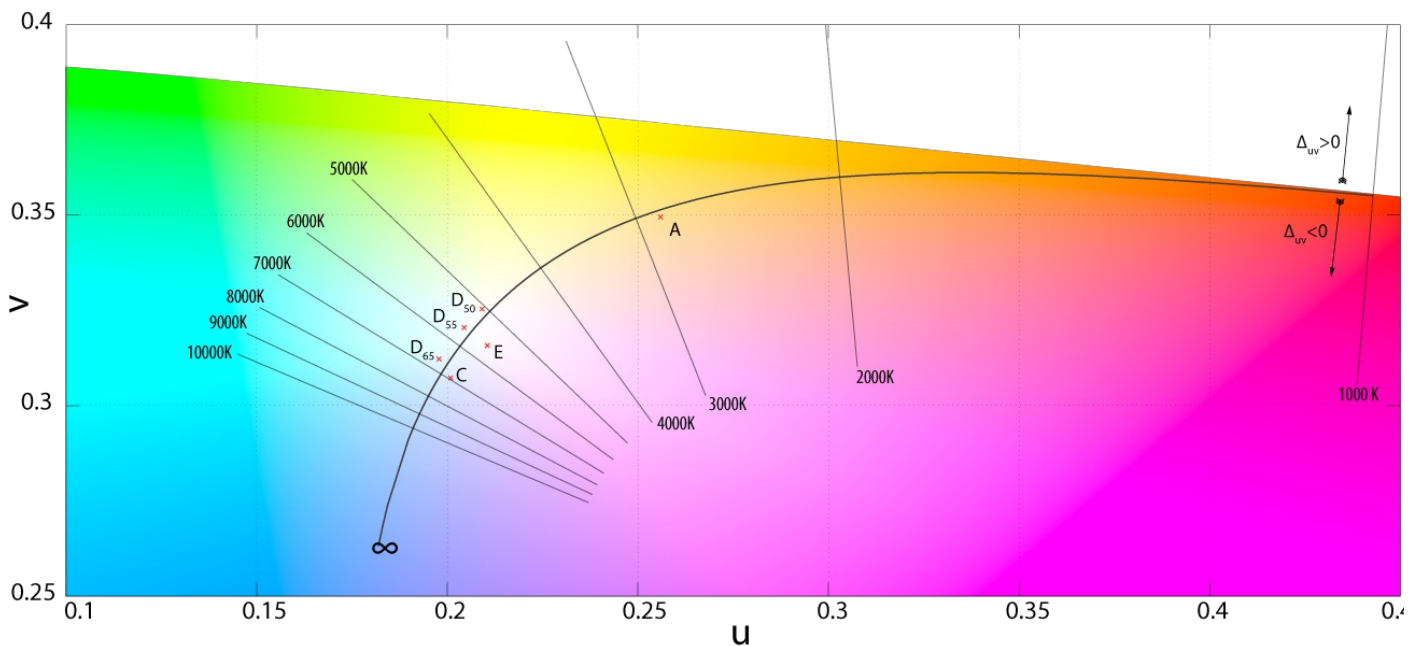
As an alternative to manually specifying the values, there's also a color picker. When a color is selected, temperature and tint are set such that this color ends up being balanced to white. This only works if the color is close enough to a blackbody emitter.

Temperature

The blackbody temperature of the primary illuminant. By default a D65 white point is used.

Tint

The amount of green/magenta shift of the blackbody curve.



Blackbody temperature curve.

Tip

White balancing can also be accomplished as part of the compositing pipeline by using the [Color Balance Node](#)

Image Files

When working with image files, the default color space is usually the right one. If this is not the case, the color space of the image file can be configured in the image settings. A common situation where manual changes are needed is when working with or baking normal maps or displacement maps, for example. Such maps do not actually store colors, just data encoded as colors. Those images should be marked as *Non-Color Data*.

Image data-blocks will always store float buffers in memory in the scene linear color space, while a byte buffer in memory and files in a drive are stored in the specified [color space](#) setting.

By default, only renders are displayed and saved with the render *View Transformation* applied. These images are the “Render Result” and “Viewer” image data-blocks, and the files saved directly to a drive with the Render Animation operator. However, when loading a render saved to an intermediate OpenEXR file, Blender cannot detect automatically that this is a render (it could be e.g. an image texture or displacement map). We need to specify that this is a render and that we want the transformations applied, with these two settings:

View as Render

Display the image data-block (not only renders) with view transform, exposure, gamma, RGB curves applied. Useful for viewing rendered frames linear OpenEXR files the same as when rendering them directly.

Save as Render

Option in the image save operator to apply the view transform, exposure, gamma, RGB curves. This is useful for saving linear OpenEXR to e.g. PNG or JPEG files in display space.

OpenColorIO Configuration

Blender comes with a standard OpenColorIO configuration that contains a number of useful display devices and view transforms. The reference linear [Color Space](#) used is the linear color space with Rec. 709 chromaticities and D65 white point.

However, OpenColorIO is also designed to give a consistent user experience across [multiple applications](#), and for this, a single shared configuration file can be used. Blender will use the standard OCIO environment variable to read an OpenColorIO configuration other than the default Blender one. More information about how to set up such a workflow can be found on the [OpenColorIO website](#).

Blender currently use the following color space rules:

scene_linear

Color space used for rendering, compositing, and storing all float precision images in memory.

data

Color space for non-color data.

aces_interchange

ACES2065-1 color space. Used to derive chromaticities of the *scene_linear* color space, for effects such as blackbody emission.

color_picking

Defines the distribution of colors in color pickers. It is expected to be approximately perceptually linear, have the same gamut as the *scene_linear* color space, map 0..1 values to 0..1 values in the scene linear color space for predictable editing of materials' albedo.

default_sequencer

Default color space for the Sequencer, *scene_linear* if not specified.

default_byte

Default color space for byte precision images and files, *texture_paint* if not specified.

default_float

Default color space for float precision images and files, *scene_linear* if not specified.

The standard Blender configuration includes support for saving and loading images in [ACES \(code and documentation\)](#) color spaces. However, the ACI gamut is larger than the Rec. 709 gamut, so for best results, an ACES specific configuration file should be used. OpenColorIO provides an [ACES configuration](#) file, though it may need a few more tweaks to be usable in production.

Default OpenColorIO Configuration

Color Spaces

Blender's OCIO configuration file is equipped by default to read/write files in these color spaces:

sRGB:

Standard RGB display space using Rec. 709 chromaticities and a D65 white point.

Rec.2020:

BT.2020 2.4 Exponent EOTF Display.

Rec.1886:

BT.1886 2.4 Exponent EOTF Display, commonly used for TVs.

Non-Color:

Generic data that is not color, will not apply any color transform (e.g. normal maps).

Linear Rec.709:

Linear BT.709 chromaticities with illuminant D65 white point.

Linear Rec.2020:

Linear BT.2020 with illuminant D65 white point.

Linear FilmLight E-Gamut:

Linear E-Gamut with illuminant D65 white point.

Linear DCI-P3 D65:

Linear DCI-P3 with illuminant D65 white point.

Linear CIE-XYZ E:

1931 CIE XYZ standard with assumed illuminant E white point.

Linear CIE-XYZ D65:

1931 CIE XYZ with adapted illuminant D65 white point.

Filmic sRGB:

Similar to *sRGB* but uses the Filmic view transform.

Filmic Log:

Intermediate log color space of Filmic view transform.

Display P3:

Apple's Display P3 with sRGB compound (piece-wise) encoding transfer function, common on Mac devices.

ACEScg:

An ACES color space that is designed to be used for rendering and compositing. It uses the AP1 color primaries, a D60 white point, and a linear transfer function. While similar to ACES2065-1, this color space has a smaller color gamut. The smaller gamut allow it to better represent the colors that fit inside the CIE 1931 chromaticities diagram. Colors that lie outside the CIE 1931 chromaticities are generally not important to rendering and compositing because the human stimulus cannot represent these colors.

ACES2065-1:

An ACES color space using the AP0 color primaries, a D60 white point and a linear transfer function. This color space is meant to store and transfer data with the most amount of possible color information.

[Previous](#)
[Open Shading Language](#)

Copyright © : This page is licensed under a CC-BY-SA 4.0 Int. License
Made with [Furo](#)
Last updated on 2025-05-10

[No](#)
[Freestyl](#)

[View Source](#)
[View Translation](#)
[Report issue on this page](#)