

## 10.1.9 Editors - Compositor Editor - Header - Add Menu - Color

### Table of content

Detailed Table of Content.....	1
Add menu - Color.....	5
Alpha Convert.....	5
Alpha Over.....	5
Set Alpha.....	6
Combine Color.....	7
Separate Color.....	8
Depth Combine.....	9
Mix Color.....	10
Blackbody.....	11
ColorRamp.....	11
Convert Colorspace.....	13
Convert to Display.....	14
Invert Color.....	15
RGB to BW.....	15

### Detailed Table of Content

#### Detailed table of content

Detailed Table of Content.....	1
Add menu - Color.....	5
Alpha Convert.....	5
Input.....	5
Image.....	5
Properties.....	5
Mapping Type.....	5
Output.....	5
Image.....	5
Alpha Over.....	5
Inputs.....	5
Background.....	5
Foreground.....	6
Factor.....	6
Straight Alpha.....	6
Properties.....	6
Over.....	6
Disjoint Over.....	6
Conjoint Over.....	6
Outputs.....	6
Image.....	6
Set Alpha.....	6
Inputs.....	6
Image.....	6
Alpha.....	6

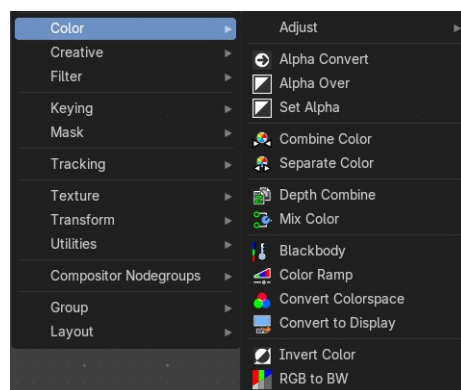
Type.....	7
Apply Mask.....	7
Replace Alpha.....	7
Outputs.....	7
Image.....	7
Combine Color.....	7
Input.....	7
Mode.....	7
Input – RGB mode.....	7
R, G, B and A.....	7
Input – HSV mode.....	7
H, S and V.....	7
Input – HSL mode.....	8
H, S and L.....	8
Input – YCbCRA mode.....	8
Y, Cb, Cr and A.....	8
Properties.....	8
Modes.....	8
Input – YUVA mode.....	8
Y, U, V and A.....	8
Output.....	8
Image.....	8
Separate Color.....	8
Input.....	8
Mode.....	8
Output – RGB mode.....	9
R, G, B and A.....	9
Output – HSV mode.....	9
H, S and V.....	9
Output – HSL mode.....	9
H, S and L.....	9
Output – YCbCRA mode.....	9
Y, Cb, Cr and A.....	9
Properties.....	9
Modes.....	9
Output – YUVA mode.....	9
Y, U, V and A.....	9
Depth Combine.....	9
Inputs.....	9
A.....	9
Depth A.....	10
B.....	10
Depth B.....	10
Use Alpha.....	10
Anti-Alias Z.....	10
Outputs.....	10
Result.....	10
Depth.....	10
Mix Color.....	10
Inputs.....	10
Factor.....	10
Image 1.....	10

<i>Image 2</i> .....	10
Properties.....	11
Mix.....	11
Clamp.....	11
Outputs.....	11
Image.....	11
Blackbody.....	11
Inputs.....	11
Temperature.....	11
Outputs.....	11
Color.....	11
ColorRamp.....	11
Inputs.....	11
Factor.....	11
Properties.....	11
Color Ramp.....	11
Controls.....	12
+.....	12
-.....	12
Tools menu.....	12
Flip Color Ramp.....	12
Distribute Stops from Left.....	12
Distribute Stops Evenly.....	12
Eyedropper (pipette icon) E.....	12
Reset Color Ramp.....	12
Color Mode.....	12
RGB.....	12
HSV/HSL.....	12
Interpolation.....	12
Ease.....	12
Cardinal.....	12
Linear.....	12
B-Spline.....	12
Constant.....	12
Color Ramp.....	13
Active Color Stop elements.....	13
Choose active color stop.....	13
Pos.....	13
Outputs.....	13
Color.....	13
Alpha.....	13
Convert Colorspace.....	13
Input.....	13
Image.....	13
Properties.....	13
From.....	13
To.....	13
Output.....	14
Image.....	14
Convert to Display.....	14
Inputs.....	14
Image.....	14

Invert.....	14
Properties.....	14
Display.....	14
View.....	14
Look.....	15
Output.....	15
Image.....	15
Invert Color.....	15
Inputs.....	15
Color.....	15
Factor.....	15
Invert Color.....	15
Invert Alpha.....	15
Outputs.....	15
Color.....	15
RGB to BW.....	15
Inputs.....	16
Image.....	16
Outputs.....	16
Val.....	16

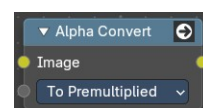
## Add menu - Color

Here you find color related nodes.



### Alpha Convert

This node converts the alpha channel interpretation of an image from pre-multiplied to straight or vice versa. With a straight alpha channel you might run into artifacts at the borders when the pixels in the semi transparent areas.



#### Input

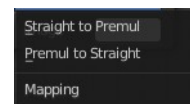
##### Image

The input image.

#### Properties

##### Mapping Type

Convert straight to premultiplied. Or convert premultiplied to straight.



#### Output

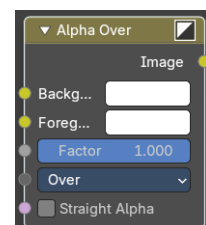
##### Image

The image output.

### Alpha Over

The Alpha Over node is used to layer images with an alpha channel on top of one another.

Where the foreground image pixels have an alpha greater than 0, the background image will be overlaid.



#### Inputs

##### Background

Input for the background image.

## Foreground

Input for the foreground image.

## Factor

Controls the amount of foreground image. A factor less than 1 will make the foreground more transparent.

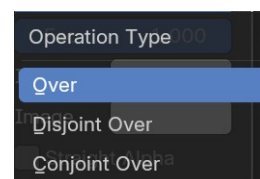
## Straight Alpha

Defines whether the foreground is in straight alpha form, which is necessary to know for proper alpha compositing. Images in the compositor are in premultiplied alpha form by default. So it should be false in most cases. But if, and only if, the foreground was converted to straight alpha form for some reason, this should be set to true.

## Properties

### Over

The foreground goes over the background according to the alpha of the foreground.



### Disjoint Over

The foreground goes over the background according to the alpha of the foreground while assuming the background is being held out by the foreground.

### Conjoint Over

The foreground goes over the background according to the alpha of the foreground. But the foreground completely covers the background if it is more opaque but not necessarily completely opaque.

## Outputs

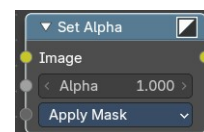
### Image

Image output.

---

## Set Alpha

The Set Alpha Node adds an alpha channel to an image.



## Inputs

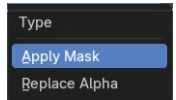
### Image

Standard image input.

### Alpha

The amount of Alpha can be set for the whole image by using the input field or per pixel by connecting to the socket.

## Type



### Apply Mask

Multiply the alpha channel of the input image by the alpha input value.

### Replace Alpha

Replace the alpha channel of the input image by the alpha input value.

## Outputs

### Image

The image output.

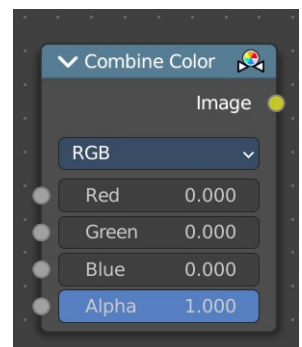
---

## Combine Color

Combine single RGBA floating point channels into a single image.

## Input

### Mode



- **RGB** colour processing
- **HSV** colour processing
- **HSL** colour processing
- **YcbCr** converts an YCbCrA image to RGBA color space and unions the channels.
  - Y: Luminance, 0=black, 1=white
  - Cb: Chrominance Blue, 0=Blue, 1=Yellow
  - Cr: Chrominance Red, 0=Red, 1=Yellow
- **YUV** converts an YUVA image to RGBA color space. Note that U and V values range from -0.5 to +0.5.

## Input – RGB mode

### R, G, B and A

The red, green, blue and alpha channels of an image.

## Input – HSV mode

### H, S and V

The Hue, Saturation and Value channels of an image.

## Input – HSL mode

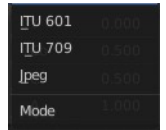
### *H, S and L*

The Hue, Saturation and Luminescence channels of an image.

## Input – YCbCrA mode

### *Y, Cb, Cr and A*

Luminance, Chrominance Blue, Chrominance Red and Alpha input.



## Properties

### *Modes*

ITU 601, ITU 709, Jpeg. These are encoding standards for the YCbCrA color space.

## Input – YUVA mode

### *Y, U, V and A*

Luminance, Chrominance U, Chrominance V and Alpha channel.

## Output

### *Image*

The output image.

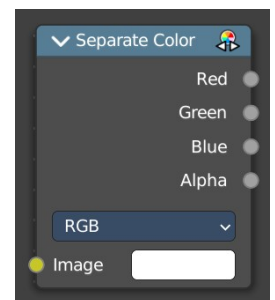
---

## Separate Color

Separates the single RGBA channels from a single image.

## Input

### *Mode*



- **RGB** colour processing
- **HSV** colour processing
- **HSL** colour processing
- **YcbCr** converts an YCbCrA image to RGBA color space and unions the channels.
  - Y: Luminance, 0=black, 1=white
  - Cb: Chrominance Blue, 0=Blue, 1=Yellow
  - Cr: Chrominance Red, 0=Red, 1=Yellow
- **YUV** converts an YUVA image to RGBA color space. Note that U and V values range

from -0.5 to +0.5.

## Output – RGB mode

### *R, G, B and A*

The red, green, blue and alpha channels of an image.

## Output – HSV mode

### *H, S and V*

The Hue, Saturation and Value channels of an image.

## Output – HSL mode

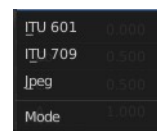
### *H, S and L*

The Hue, Saturation and Luminescence channels of an image.

## Output – YCbCrA mode

### *Y, Cb, Cr and A*

Luminance, Chrominance Blue, Chrominance Red and Alpha input.



## Properties

### *Modes*

ITU 601, ITU 709, Jpeg. These are encoding standards for the YCbCrA color space.

## Output – YUVA mode

### *Y, U, V and A*

Luminance, Chrominance U, Chrominance V and Alpha channel.

---

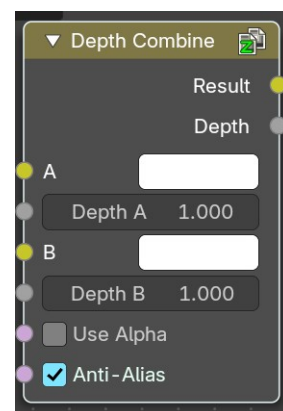
## Depth Combine

The Depth Combine node combines two images based on their Z-depth maps. It overlays the images using the provided Z values to detect which parts of one image are in front of the other.

## Inputs

### *A*

The background image.



### ***Depth A***

Z depth of the background image.

### ***B***

The foreground image.

### ***Depth B***

Z depth of the foreground image.

### ***Use Alpha***

The chosen Image pixel alpha channel is also carried over. If a pixel is partially or totally transparent, the result of the Z Combine will also be partially transparent; in which case the background image will show through the foreground (chosen) pixel.

### ***Anti-Alias Z***

Applies Anti-Aliasing to avoid artifacts at sharp edges or areas with a high contrast.

## **Outputs**

### ***Result***

If both Z values are equal, it will use the foreground image. Whichever Z value is less decides which image pixel is used. See Z-buffer.

### ***Depth***

The combined Z depth, which allows to thread multiple Z-combines together.

## **Mix Color**

The Mix Node mixes images by working on the individual and corresponding pixels of the two input images. Called “MixRGB” in the shader and texture context.

## **Inputs**

### ***Factor***

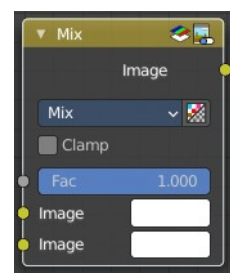
Controls the amount of influence the node exerts on the output image.

### ***Image 1***

Usually the background image. The image size and resolution sets the dimensions of the output image.

### ***Image 2***

Usually the foreground image.



## Properties

### **Mix**

Choose the different blending modes.

Add, Subtract, Multiply, Screen, Divide, Difference, Darken, Lighten, Overlay, Color Dodge, Color Burn, Hue, Saturation, Value, Color, Soft Light, Linear Light.

### **Clamp**

Limit the highest color value to not exceed 1.

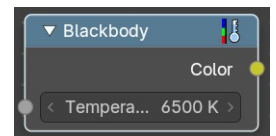
## Outputs

### **Image**

Image output.

## Blackbody

Convert a blackbody temperature to a rgb value.



## Inputs

### **Temperature**

The blackbody temperature.

## Outputs

### **Color**

The output RGB color.

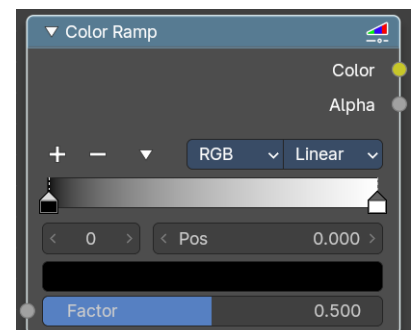
## ColorRamp

The Color Ramp Node is used for mapping values to colors with the use of a gradient.

## Inputs

### **Factor**

The Factor input is used as an index for the color ramp.



## Properties

### Color Ramp

Color Ramps enables the user to specify a range of colors based on color stops. The color between the color stops gets interpolated.

### Controls

+

Add a stop to your color ramp. The stop will be added after the selected one, in the middle to the next one.

-

Deletes the selected color stop from the list.

### Tools menu

#### Flip Color Ramp

Flips the gradient, inverting the values of the color ramp.

#### Distribute Stops from Left

Rearrange the stops so that every step has the same space to the right.

#### Distribute Stops Evenly

Space between all neighboring stops becomes equal.

#### Eyedropper (pipette icon) E

An Eyedropper to sample a color or gradient from the interface to be used in the color ramp.

#### Reset Color Ramp

Resets the color ramp to its default state.

### Color Mode

#### RGB

Blends color by mixing each color channel and combining.

#### HSV/HSL

Blends colors by first converting to HSV or HSL, mixing, then combining again. This has the advantage of maintaining saturation between different hues, where RGB would de-saturate, this allows for a richer gradient.

### Interpolation

#### Ease

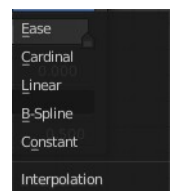
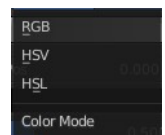
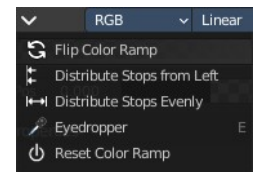
Uses an Ease Interpolation for the color stops.

#### Cardinal

Uses a Cardinal Interpolation for the color stops.

#### Linear

Uses a Linear Interpolation for the color stops.



## B-Spline

Uses a B-Spline Interpolation for the color stops.

## Constant

Uses a Constant Interpolation for the color stops.

## Color Ramp

The color band. A click at one of the color stops makes it the active one. You can move the color stops by clicking at them and dragging them around.



## Active Color Stop elements

Adjust the active color stop.



## Choose active color stop

Choose the color stop by index.

## Pos

The position of the active color stop. The range goes from 0.000 to 1.000

## Outputs

### Color

The color output.

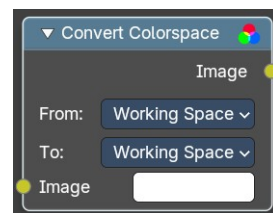
### Alpha

Alpha channel output.

## Convert Colorspace

Convert between color spaces.

Note that the conversion is skipped when converting between the same color spaces or to or from data spaces.



## Input

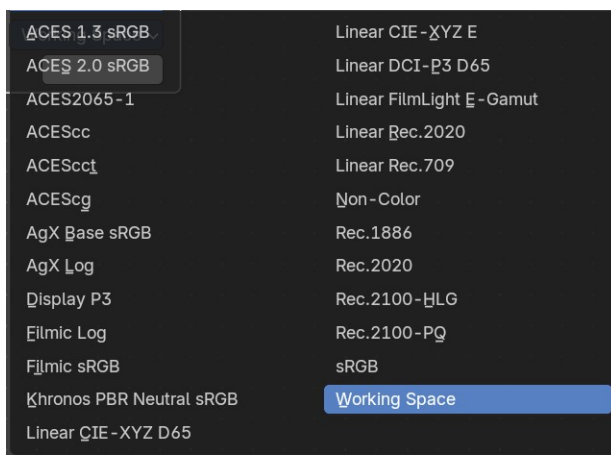
### Image

The input image.

## Properties

### From

The current color space.



## To

The destination color space.

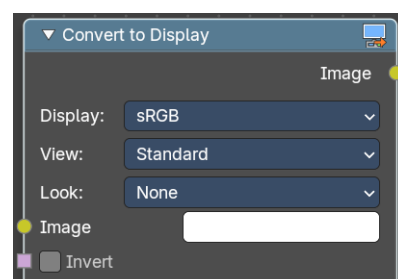
## Output

### Image

The image output.

## Convert to Display

Transforms an image from **scene-linear color space** into a **display-referred format**, using Blender's color management pipeline. This is essential when working with raw data, custom view transforms, or prepping visuals for final output.



## Inputs

### Image

The input image in scene-linear space—typically the result of render layers, raw file input, or procedural compositing.

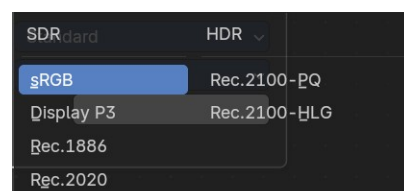
### Invert

Reverses the display transform—useful for decoding display-referred images back to linear.

## Properties

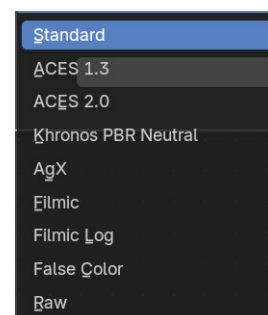
### Display

Selects the target display device (e.g. sRGB, Rec.709, Display P3).  
Determines how the image is mapped for viewing on screens.



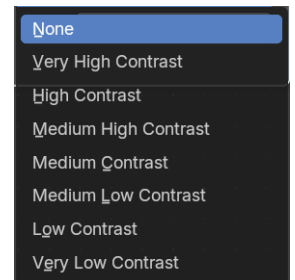
### View

Applies a tone mapping curve (e.g. Standard, Filmic, Raw).  
Controls contrast, highlight rolloff, and perceptual brightness.



## Look

Optional artistic adjustment layer (e.g. Medium Contrast, High Contrast, None). Adds stylized tweaks on top of the view transform.



## Output

### Image

The display-referred result, ready for preview, export, or further compositing.

---

## Invert Color

The Invert Node inverts the colors in the input image, producing a negative.

### Inputs

#### Color

Standard input.

#### Factor

Controls the amount of influence the node exerts on the output image.

#### Invert Color

Invert the RGB values.

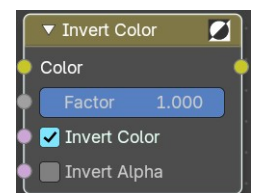
#### Invert Alpha

Invert the Alpha values.

### Outputs

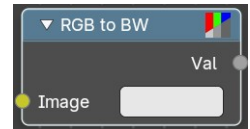
#### Color

Standard image output.



## RGB to BW

The RGB to BW Node converts an RGB color image to a gray-scale image based at its luminance.



### Inputs

#### *Image*

Color image input.

### Outputs

#### *Val*

Gray-scale value output.

---