Edge

When turned off, the edge detection is disabled (enabled by default).

⚠️ To view the `toon_edge`, you **must** change the `filter_type` (sampling settings) to `contour`.

- Increasing the `contour_filter_width` (sampling settings) value will increase render times.
The **Contour Filter** can be found in the **Filter** settings of the **Render Settings** window.

**Edge Color**

The color of the toon edge. The line style can be controlled with a texture here.

![Red](image1.png) ![Stylized edge with ramp](image2.png)

**Edge Tonemap**

Connect a ramp node here to change the **Edge Color** based on the shading result of the **Base**.

![Default (nothing connected)](image3.png) ![Gradient ramp](image4.png)

The direct diffuse shading contribution is mapped into the UVs. So in the example below, the back of the sphere (which has the darkest direct diffuse) is **blue** and the part of the sphere hit directly by the directional light is **green**. This then becomes yellow as the exposure of the lighting is increased.
In the example below, the *edge_color* (yellow) is multiplied by the blue ramp color (connected to *edge_tonemap*) to give green.

**Edge Opacity**

Controls the transparency of the edge.
Width Scaling

The maximum width of the \textit{contour} lines is determined by the \textit{width} parameter of the \textit{contour} filter. The actual width is the multiplication of it and this parameter. The line style can be controlled by combining this with a texture.

\begin{itemize}
\item To increase the \textit{edge\_width\_scale} above 1, you will need to increase the \textit{contour filter width}. However, this will increase render times.
\end{itemize}
You can avoid edge ‘cluttering’ in the center by connecting a facing_ratio shader to width_scale.

The width_scale can be used to control the edge depending on the distance from the camera.

You may notice flickering in animations using width_scale values below 1. Enabling lock_sampling_pattern (sampling render settings) should fix this issue.

Edge Detection

ID Difference

If enabled, edge detection uses the difference of IDs from neighboring pixels.
Enabled (default)  Disabled

Enabled (default). Rollover image for disabled.
**Shader Difference**

Detects the difference of shaders of neighboring samples. This is useful when multiple shaders are assigned to a single polymesh, for example.

**Curve, Point, and Shape IDs**

*ID Difference* draws edges on curves, points, and shapes. Points also have a different ID for each point. A sphere is covered in particles (as spheres) that use the same toon shader. `Emission:color = Utility shader with Color = Uniform ID` (the ID used by the Toon shader). The particles have edges because each sphere is a point with a different ID. The large sphere is a shape, so the Toon shader uses the name as the ID (otherwise every primitive e.g. polygon would have an edge). For curves, every curve has a different ID, so *ID Difference* enabled vs disabled would look like the images below.
Mask Color

The edge is detected when the mask color of neighboring pixels is different. The \textit{mask\_color} is assumed to have a texture connected to draw an arbitrary shape by detecting color differences. With this parameter, lines can be drawn anywhere (the filter draws lines where the color of the \textit{mask} changes).

Stylistic effects achieved by connecting different shaders to \textit{mask\_color}

The \textit{Shade Modes} of the Utility shader can be used to add more detail to the \textit{edge} when connected to \textit{mask\_color}.

\textbf{UV Threshold}

\textit{A tutorial that covers how to use the Edge Mask Color can be found here.}
If enabled, edge detection uses the difference of UVs from neighboring pixels.

Angle Threshold

When less than 180, the edge detection uses the difference of the angle between neighboring pixels.
Edge Detection

The normal used in the edge detection. Choose from: Shading Normal, Smoothed Normal, and Geometric Normal.

Shading Normal (default) with bump.  
Smoothed Normal
Advanced Edge Control

Priority

Changes the sorting priority of the edge.

Red sphere 1.

0 (all objects).

Blue plane 1 (both spheres 0).

Rollover image for red sphere 0, green sphere 1.
Ignore Throughput

By default, the contour color is affected by ray throughput. If a specific color is required for a reflected/refracted object, enable this and use a `ray_switch` shader.

Use Toon ID

Edge detection can be controlled using a `STRING` type user data called `Toon ID`. Otherwise, the detected edges will be driven by the object's own name as a toon-specific ID. This allows for grouping of objects, where `Toon ID` specifies which objects belong together regarding edge detection.
'Use Toon ID' disabled (default). 'Shader Difference' disabled. 'Use Toon ID' enabled. Toon ID '1' added to left sphere shape.

Cross Hatching

Displacing the surface with a texture can help to create a crosshatch effect with Toon Edge shading.

Teapot displaced with noise shader

Pepe model by Daniel M. Lara (Pepeland).