

[Skip to content](#)

Kuwahara Node

The Kuwahara node implements the Kuwahara filter as well as its anisotropic variant. The Kuwahara filter is a smoothing filter that tries to preserve the edges in the image. The smoothing effect of the anisotropic variant is similar to brush strokes, so the node can be used to create stylized painting effects.

Inputs

Image

Standard color input.

Size

Controls the size of the smoothing neighborhood. Large values may introduce artifacts for highly detailed areas. For the anisotropic method, the larger the size, the slower the filter.



Properties

Type

Classic:

A simple smoothing method that averages the local square neighborhood of the image while preserving edges. Produces blocky results due to the square neighborhood and provides no tuning parameters, but is faster to compute.

Anisotropic:

A complex smoothing method that averages the local neighborhood of the image in the direction of the flow of the edges, thus preserving the edges in the output. Produces painterly-like results and provides multiple tuning parameters, while being slower to compute.

High Precision

Uses a more precise but slower method. Use if the output contains undesirable noise.

Uniformity

Controls the uniformity of the directions of the edges of the image. Non uniform directions are nearly never desirable, so this should typically be increased until the user notices the result is no longer changing in a significant way. Further increases would produce worst results and increase compute time.

Sharpness

Controls the sharpness of the edges of the image.



Eccentricity

Controls how thin and directional the filter is. Low eccentricity corresponds to circular omnidirectional features while high eccentricity corresponds to thin directional features.



<input type="text"/>	<input type="text"/>
Eccentricity: 1.	Eccentricity: 2.

Outputs

Image

Standard color output.

Notes

Iterations

The filter can be applied multiple times by chaining the node multiple times. This chaining can produce more flat filtering.

<input type="text"/>	<input type="text"/>
Original.	Iterations: 1.
<input type="text"/>	<input type="text"/>
Iterations: 2.	Iterations: 3.

Performance

The filter can be expensive to compute for high size input and high resolution images. To improve performance, consider scaling down the image, applying the filter, then scaling it up again. This can work well because the filter already attenuates low frequency details.