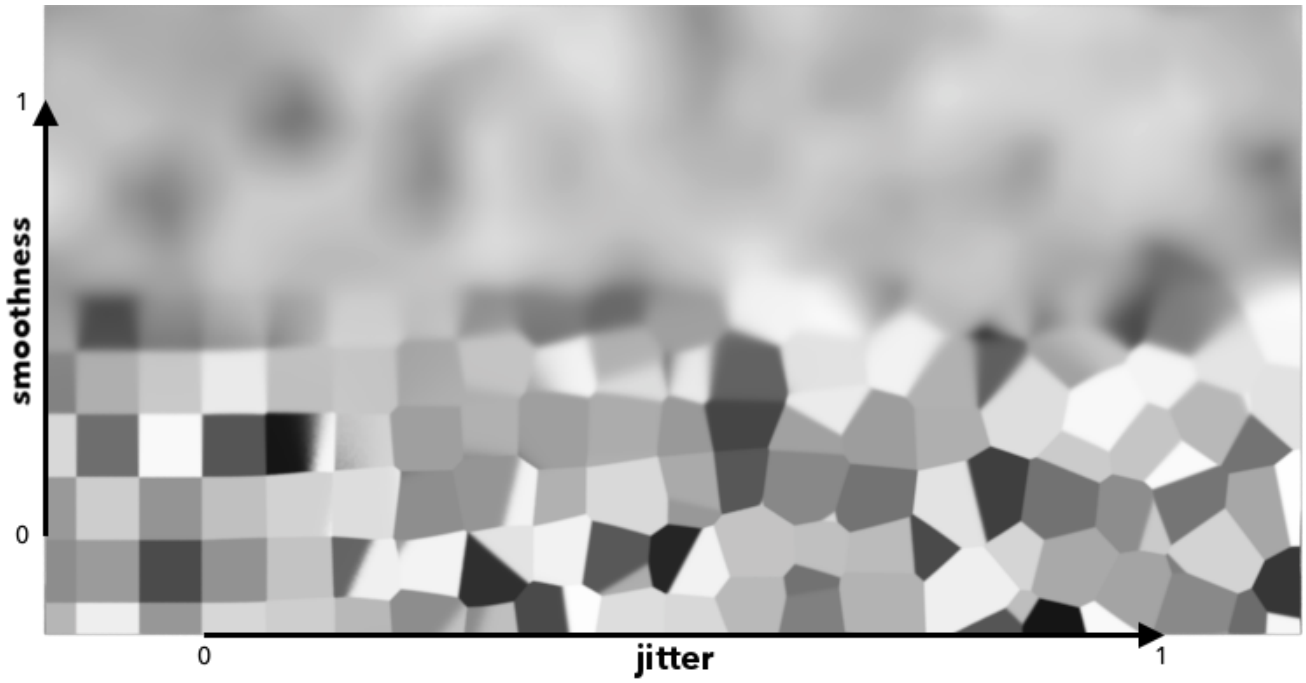
PxrVoronoise



Like all texture style nodes, this node takes a manifold that describes either a 2D or 3D domain to apply a voronoise texture to. The default behavior if no manifold is attached is to apply over P in 3D. This node computes the voronoise function, which is a blend of noise and Voronoi as described by Inigo Quilez.

The jitter and smoothness parameters allow you to create different types of patterns.

- `jitter=0,smoothness=0` produces a minimum distance non-jittered grid of values, like RSL's cellnoise.
- `jitter=0,smoothness=1` gives a noise function, similar to perlin's.
- `jitter=1,smoothness=0` produces a jittered, minimum distance Voronoi pattern.
- `jitter=1,smoothness=1` produces a combination of jittered Voronoi and noise : "voronoise" !

> Compared to Inigo's code, this version also adds fractal octaves and turbulence

Input Parameters

Frequency
Sets the lowest (starting) frequency of the noise layers.

Octaves
The number of noise octaves used. More octaves add successively more noise.

Gain
Also known as persistence. Used to shrink or expand the amplitude of each successive octave of noise. 1/f noise is achieved when gain = 1/lacunarity

Lacunarity
The scaler used to determine the frequency of each sucessive octave of noise. Smaller numbers will cause the layers to be more closely spaced in frequency. Larger values will space them further apart.

Jitter
Controls the amount of jitter applied. 0 gives regular grid, 1 gives a Voronoi like grid

Smoothness
Using 0 produces a voronoi/cellnoise pattern, 1 results in smooth noise.

Turbulent
Should the noise be turbulent

Manifold
The manifold over which to apply the noise. (The default is P).

Output Parameters

resultF

The result of voronoise texture.

resultRGB

The voronoise texture as a monochromatic color.