

All options for the utility take two forms, the **long form** and the **alias form**. Both variants, along with the functionality of the parameter are detailed below. If a parameter takes an argument, it is noted in the tables as well.

Required Options

Parameter	Shortform Alias	Takes argument	Description
<code>--input-file</code>	<code>-I</code>	✓	The input file. Must be a PNG image.
<code>--output-file</code>	<code>-O</code>	✓	The name of the file to output.

NOTE: These two parameters are *positional*, meaning that as long as `--input-file` comes before `--output-file` on the command line, the argument names can be dropped.

Example

`voronoi input.png output.svg`: Using the PNG image `input.png` create an SVG image `output.svg` with the Stippled render.

Basic Options

Parameter	Shortform Alias	Takes argument	Description
<code>--stipples</code>	<code>-s</code>	✓	The number of stipple points to render. Generally speaking, the larger the number of stipples, the more detailed the resulting output image, at the expense of longer render times. Defaults to 4000 .
<code>--colour-output</code>	<code>-c</code>	✗	If this parameter is present, the output render will be in colour. Defaults to OFF (That is, rendered output will be in black and white).

Example

`voronoi -s=18000 -c input.png output.svg`: Using the PNG image `input.png` create a colour SVG image `output.svg` with 18000 stipple points.

Advanced Options

Parameter	Shortform Alias	Takes argument	Description
<code>--threshold</code>	<code>-t</code>	✓	The cut-off point for the convergence of the stippled output. The unit of this value is the averaged displacement of all stipple points after an iteration of Lloyd's method. That is, if the averaged displacement is lower than this value, then the rendering process has finished. After each iteration of the rendering process, the expectation is that the stipple points have moved closer to their ideal spot. Generally speaking, the lower this value, the less noisy the rendering will be (a desirable quality) at the expense of longer render times. As a rule of thumb, the more stipple points there are, the lower this value should be, but lots of fiddling is necessary to get an ideal outcome. Defaults to 0.1 .
<code>--no-overlap</code>	<code>-n</code>	✗	Guarantees that stipple points will not overlap each other at the expense of some tone preservation. Defaults to OFF .
<code>--fixed-radius</code>	<code>-f</code>	✗	The radius of all stipple points will be equal regardless of the intensity of the input image. Defaults to OFF .
<code>--sizing-factor</code>	<code>-z</code>	✓	The radius of each stipple point will be multiplied by this factor. Use this parameter to control the tone preservation quality of the output image. Defaults to 1.0 .
<code>--subpixels</code>	<code>-p</code>	✓	Controls the tile size of centroid computations. Theoretically, higher values produce more faithful tone preservation at the expense of rendering speed, but real world outcomes are highly variable. Defaults to 5 .
<code>--log</code>	<code>-l</code>	✗	Determines whether the utility will produce an output log or not. If there is a problem with the utility, the output log may prove useful in diagnosing the problem. Defaults to OFF .