Inkscape tutorial: Tracing bitmaps

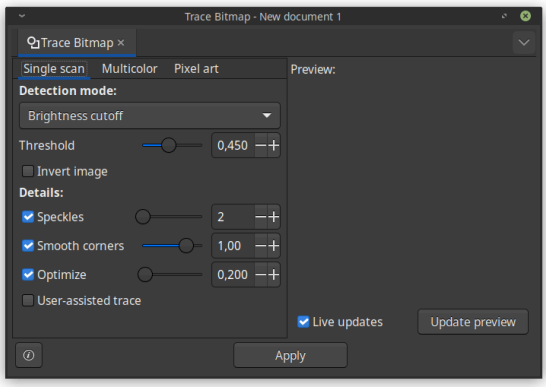
One of the features in Inkscape is a tool for tracing a bitmap image into one or more <path> elements for your SVG drawing. These short notes should help you become acquainted with how it works.

Keep in mind that the Tracer's purpose is not to reproduce an exact duplicate of the original image; nor is it intended to produce a final product. No autotracer can do that. What it does is give you a set of curves which you can use as a resource for your drawing.

Our tracer, derived from the original Potrace library by Peter Selinger, interprets a black and white bitmap, and produces a set of curves. For Potrace, we currently have three types of input filters to convert from the raw image to something that Potrace can use.

Generally the more dark pixels in the intermediate bitmap, the more tracing that Potrace will perform. As the amount of tracing increases, more CPU time will be required, and the <path> element will become much larger. It is suggested that the user experiment with lighter intermediate images first, getting gradually darker to get the desired proportion and complexity of the output path.

To use the tracer, load or import an image, select it, and select the *Path* ⇒ *Trace Bitmap* item, or Shift + Alt + B.



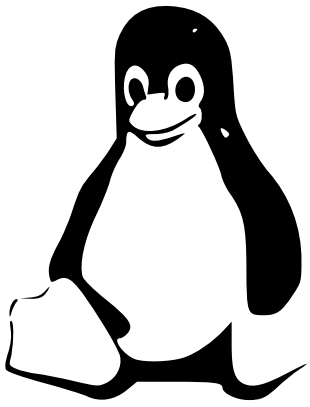
Main options within the Trace dialog

The user will see the five filter options available:

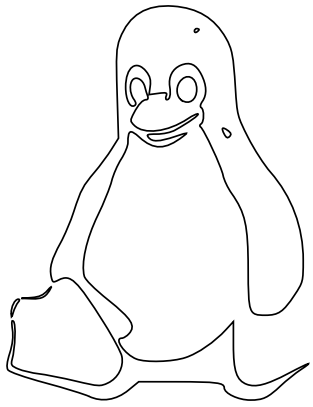
- Brightness Cutoff

This merely uses the sum of the red, green and blue (or shades of gray) of a pixel as an indicator of whether it should be considered black or white. The threshold can be set from 0.0 (black) to 1.0 (white). The higher the threshold setting, the fewer the number pixels that will be considered to be “white”, and the intermediate image with become darker.

Original Image



Brightness Threshold
Fill, no Stroke

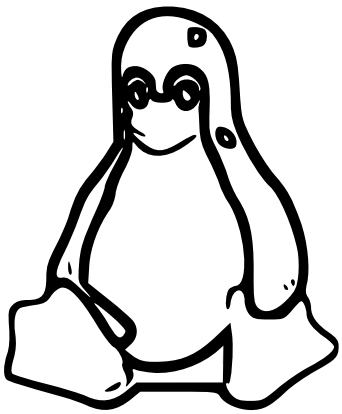


Brightness Threshold
Stroke, no Fill

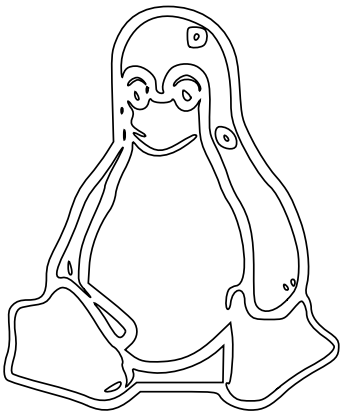
- Edge Detection

This uses the edge detection algorithm devised by J. Canny as a way of quickly finding isoclines of similar contrast. This will produce an intermediate bitmap that will look less like the original image than does the result of Brightness Threshold, but will likely provide curve information that would otherwise be ignored. The threshold setting here (0.0 – 1.0) adjusts the brightness threshold of whether a pixel adjacent to a contrast edge will be included in the output. This setting can adjust the darkness or thickness of the edge in the output.

Original Image



Edge Detected
Fill, no Stroke



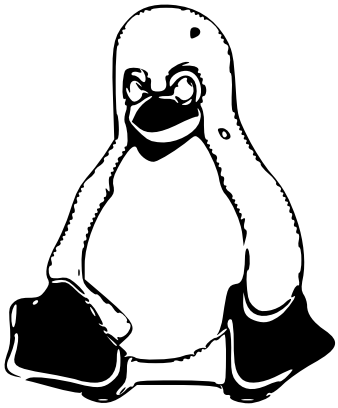
Edge Detected
Stroke, no Fill

- Color Quantization

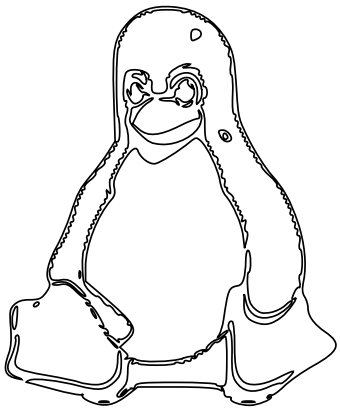
The result of this filter will produce an intermediate image that is very different from the other two, but is very useful indeed. Instead of showing isoclines of brightness or contrast, this will find

edges where colors change, even at equal brightness and contrast. The setting here, Number of Colors, decides how many output colors there would be if the intermediate bitmap were in color. It then decides black/white on whether the color has an even or odd index.

Original Image



Quantization (12 colors)
Fill, no Stroke



Quantization (12 colors)
Stroke, no Fill

The user should try all three filters, and observe the different types of output for different types of input images. There will always be an image where one works better than the others.

After tracing, it is also suggested that the user try ***Path*** ⇒ ***Simplify*** (Ctrl + L) on the output path to reduce the number of nodes. This can make the output of Potrace much easier to edit. For example, here is a typical tracing of the Old Man Playing Guitar:

Original Image

Traced Image / Output Path
(1,551 nodes)

Note the enormous number of nodes in the path. After hitting Ctrl + L, this is a typical result:

Original Image

Traced Image / Output Path - Simplified
(384 nodes)

The representation is a bit more approximate and rough, but the drawing is much simpler and easier to edit. Keep in mind that what you want is not an exact rendering of the image, but a set of curves that you can use in your drawing.

- Autotrace

The Autotrace option uses a different algorithm for tracing and also offers some other parameters to tweak. It may take a little longer to work, but gives you some variety to choose from.

- Centerline tracing (autotrace)

If you would like to vectorize a line drawing, and get strokes that are easy to modify instead of filled areas as a result, use this option. It will attempt to find contiguous lines that make up your drawing.

Authors: Bulia Byak; Jonathan Leighton; Colin Marquardt; Nicolas Dufour; Gellért Gyuris; Maren Hachmann
Header / footer design: Esteban Capella — 2019

Inkscape is Free and Open Source Software licensed under the GPL.
Documents Team Language: 'en'