

Stack Overflow is a question and answer site for professional and enthusiast programmers. It's 100% free, no registration required.

[Take the 2-minute tour](#) 

## How to convert a JPEG image into SVG format using ImageMagick?



How to convert a JPEG image into SVG format using ImageMagick?

[imagemagick](#)

edited Sep 26 '12 at 20:26



[Kurt Pfeifle](#)

23.9k 5 56 108

[add comment](#)

asked Jul 15 '09 at 16:55

[jasonb](#)

### 3 Answers

You'll actually need some software or code to vectorize your image in between, as jpg is a raster format, while SVG is a vector format. I don't think imagemagick alone can do that for you.

Here is an [online application](#) built on top of Autotrace, ImageMagick and other components that does what you want.

answered Jul 15 '09 at 16:57



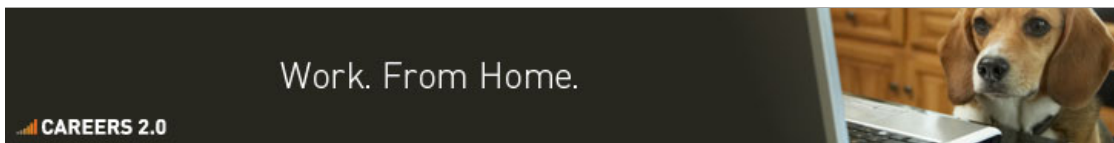
[schnaader](#)

25.9k 2 46 81

3 +1: Turning a bunch of lines into a large matrix of pixels can be done fairly dumbly. Doing the reverse requires some fairly serious smarts. – [T.E.D.](#) Jul 15 '09 at 17:32

That's kind of nifty. – [Snarfblam](#) Jul 15 '09 at 18:01

[add comment](#)



you'll need to use [potrace](#) and convert to a bitmap first.

```
$convert input.jpg output.ppm
$potrace -s output.ppm -o svgout.svg
```

edited Sep 26 '12 at 18:49



[kapa](#)

41.4k 12 72 114

[add comment](#)

answered Sep 26 '12 at 18:45



[iamashm](#)

81 1 1

Actually, with a complete installation of a recent version of ImageMagick it should be as easy as:

```
convert some.jpeg some.svg
```

Of course, ImageMagick cannot do it all by itself -- it uses *delegates* (helper programs) to handle SVG input or output. (This has been pointed out by other answers already.)

To see a (partial) list of all delegates (and their respective commands), run

```
convert -list delegate
```

To see the config file where all the delegate secrets hide, see

```
convert -list delegate | grep delegates.xml
```

To see a (partial) list of SVG handling delegates, run

```
convert -list delegate | grep -i svg
```

However, ImageMagick likes to put some of its external helper utilities into 'stealth' mode and doesn't necessarily reveal their presence when using above commands.

Just look into the `delegates.xml` file itself. On my system it's:

```
grep -i svg /opt/local/etc/ImageMagick/delegates.xml | grep -i --color stealth
```

```
<delegate decode="autotrace" stealth="True" \
  command="&quot;/opt/local/bin/convert&quot; &quot;%i&quot; \
  &quot;pnm:%u&quot;\n\
  &quot;/opt/local/bin/autotrace&quot; \
  -input-format pnm \
  -output-format svg \
  -output-file &quot;%o&quot; &quot;%u&quot;"/>
```

```
<delegate decode="svg:decode" stealth="True" \
  command="&quot;/opt/local/bin/inkscape&quot; &quot;%s&quot; \
  --export-png=&quot;%s&quot; \
  --export-dpi=&quot;%s&quot; \
  --export-background=&quot;%s&quot; \
  --export-background-opacity=&quot;%s&quot; \
  &gt; &quot;%s&quot; 2&gt;&amp;1"/>
```

As you may see, on *my* system the ImageMagick installation automatically uses (amongst others)...

- ... `inkscape` to convert SVG to PNG;
- ... `autotrace` to convert PNM to SVG;

Of course, one could argue the benefits of rather using `autotrace` directly -- but that would require to manually convert the whatever-input-format to PNM first. So for this preliminary step you'd probably use ImageMagick anyway...

answered Sep 26 '12 at 19:20



Kurt Pfeifle

23.9k 5 56 108

[add comment](#)