

QGIS Application - Bug report #8980

svg fill issue: lines not rendered correctly

2013-11-01 11:57 AM - Giovanni Manghi

Status:	Rejected	
Priority:	Normal	
Assignee:		
Category:	Symbology	
Affected QGIS version:	2.0.1	Regression?: No
Operating System:		Easy fix?: No
Pull Request or Patch supplied:		Resolution:
Crashes QGIS or corrupts data:		Copied to github as #: 17641
Description		
I attach here a svg that is being used to fill polygons.		
The pattern of the svg is made of simple, sharp lines/dots.		
When applied in qgis the result is that not all the pattern is rendered in the same way, see attached screenshot.		

History

#1 - 2013-11-06 01:27 AM - Paolo Cavallini

it would help to have just one repetition of the input pattern in the SVG to avoid the unwanted effect - might not be a bug

#2 - 2013-11-09 06:53 AM - Paolo Cavallini

- Subject changed from *svg fill issue: lines not rendererd corretcly* to *svg fill issue: lines not rendered correctly*

#3 - 2014-01-27 03:28 AM - Radim Blazek

- File *picture.py* added
- File *render.py* added
- File *render.png* added

There are two kind of irregularities with sv11.svg which do apply to any use of that svg including point markers and line markers. See attached a single tile *render.png* rendered by *render.py*

1) The first row of lines is cut to half by top border. That is not surprising, the graphics go out of the SVG border (half of the line width) and thus it is cut of. I consider this a feature. Solution is to design the SVG so that graphics are withing the SVG extent.

2) Even/odd rows (similar for columns but less obvious) are rendered different (1px black x 2px gray). That is due to antialiasing used when the SVG is rendered to an image (depends where line float coordinates fall into image pixels). Qt SVG module has hardcoded antialiasing on as mentione for example here http://qt-project.org/faq/answer/how_to_toggle_antialiasing_for_svg_rendering even not explicitly stated in Qt doc. Strange thing is, that even when the SVG is stored as QImage the picture is still antialiased when rendered on image, see attached *picture.py*. Anyway, we want to use antialiased rendering of SVG files to get smooth symbols.

To be honest, I don't know clean solution for this problem. It would be probably possible to render the tile on bigger image and then scale it down using `QPainter::SmoothPixmapTransform` but that would decrease quality of all SVG symbols.

Is it the SVG pattern a workaround for line pattern fill bug in 2.0?

Files			
sv11.svg	1.82 KB	2013-11-01	Giovanni Manghi
tassello_svg.JPG	9.89 KB	2013-11-01	Giovanni Manghi
55.png	247 KB	2013-11-01	Giovanni Manghi
render.png	323 Bytes	2014-01-27	Radim Blazek
render.py	315 Bytes	2014-01-27	Radim Blazek
picture.py	619 Bytes	2014-01-27	Radim Blazek