# SDF Texture Generator Documentation

A distance field is field that contains a measure of distance from some something. A signed distance field (SDF) measures the distance from some contour in both directions, inward and outward. In the case of SDF textures, the distance is stored per pixel as a value in the 0-1 range. Typically, 0.5 represents zero distance from the contour. A value of 1 corresponds with fully inside and a value of 0 means fully outside, though sometimes it's the other way around.
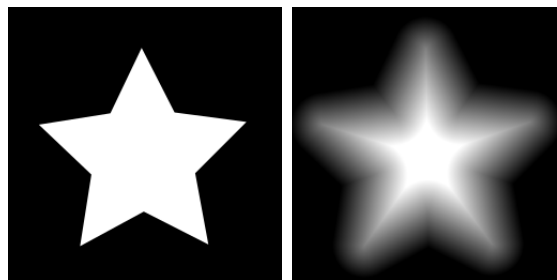

Original and SDF (alpha replaced with black).

SDF textures are often created by starting with a very-high resolution aliased bitmap, then sampling it to created a lower-resolution SDF. This brute-force approach is slow and uses a lot of memory.

In contrast, this tool works with an anti-aliased source image and quickly produces an SDF of the same resolution. You need single-pixel anti-aliasing, no multi-pixel blurring. Typically, you can directly use the output from any image-editing program or bitmap font exporter.

Generating an SDF Texture

To create an SDF texture inside the Unity editor, open the generator via the *Window / SDF Texture Generator* menu item. If you have a texture selected while doing so, it will be automatically used as the source texture.
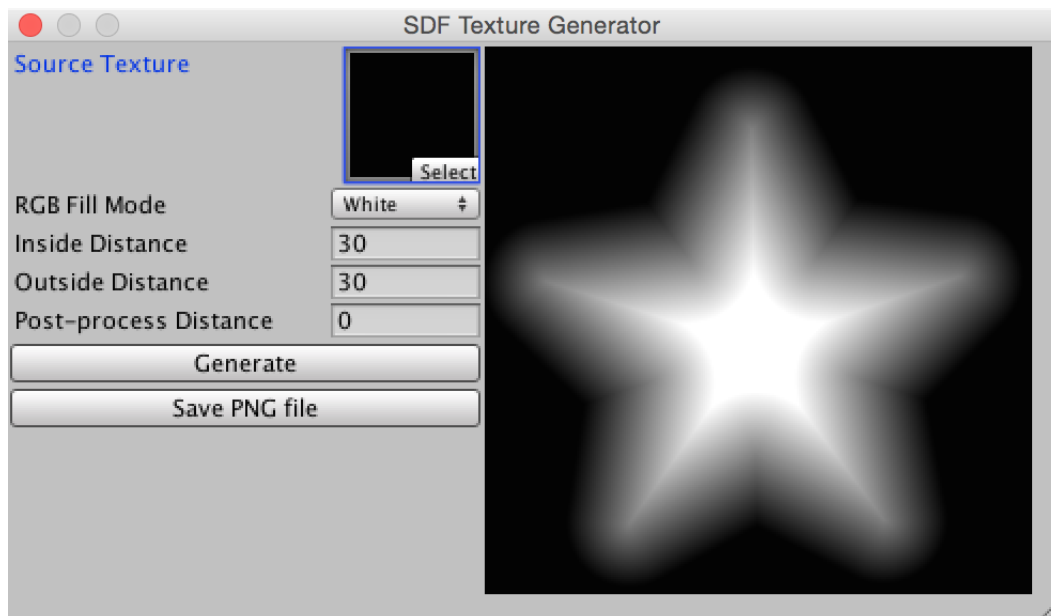
SDF Texture Generator Window.

Source Texture is any texture with an alpha channel. If it's an alpha-only texture then the source preview will appear solid black. You don't need to worry about texture import settings, they will be temporarily changed if needed.

RGB Fill Mode controls what's put into the RGB channels of the generated SDF texture. Your choices are solid white, solid black, the SDF, and the source RGB. The alpha channel always contains the SDF.

Inside Distance determines how far the field will penetrate inside the shape. Likewise for **Outside Distance**, but in the other direction. These distances are in pixels.

Pixels fully inside the shape will get an alpha value of 1, while pixels fully outside the shape will get an alpha of 0. Typically both distance settings will be greater than zero, and pixels that are exactly on the original contour will become 0.5. But if you set one of the distances to zero, then the gradient will start as either 0 or 1 at the contour.

Inside/outside distances set to 20/20, 0/20, and 20/0.

Post-process Distance is another pixel distance. When larger than zero, pixels close to the contour will be evaluated again in a brute-force fashion to possibly get a more accurate result. Only use this when you get bad results. Personally, I've never used it.

Generate creates the SDF texture. It should typically do this quickly, though it might take seconds for large textures, especially if texture import settings need to be temporarily changed.

Save PNG file opens the save dialog so you can store the SDF texture. The default location is the same as that of the source texture. When saved inside the Unity project, the texture will have the default import settings. You'd typically change this to Alpha 8 format and make some other tweaks.

Except for the source texture, all configuration options of the generator are persistent.

Using the API

You can also generate SDF textures through code. The `CatlikeCoding.SDFToolkit` namespace contains the `SDFTextureGenerator` class. It has a static `Generate` method that has parameters corresponding to the configuration options described above. Check the code documentation for details.

While it is currently not handy or optimized to do so, it is possible to use the generator at runtime in your applications.

Acknowledgement and License

The generator is based on the Anti-aliased Euclidean distance transform described by Stefan Gustavson and Robin Strand. The algorithm it uses is an adapted version of Stefan Gustavson's code and falls under the permissive MIT License. This means that you can bundle it with your commercial products.

The MIT license applies only to the `SDFTextureGenerator.cs` class file. It does not cover the rest of the SDF Toolkit.

made by Jasper Flick

copyright belongs to Catlike Coding