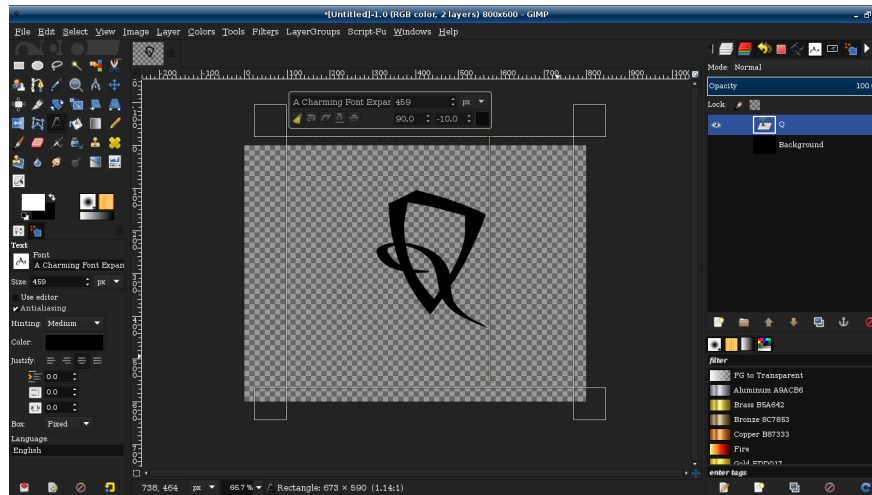


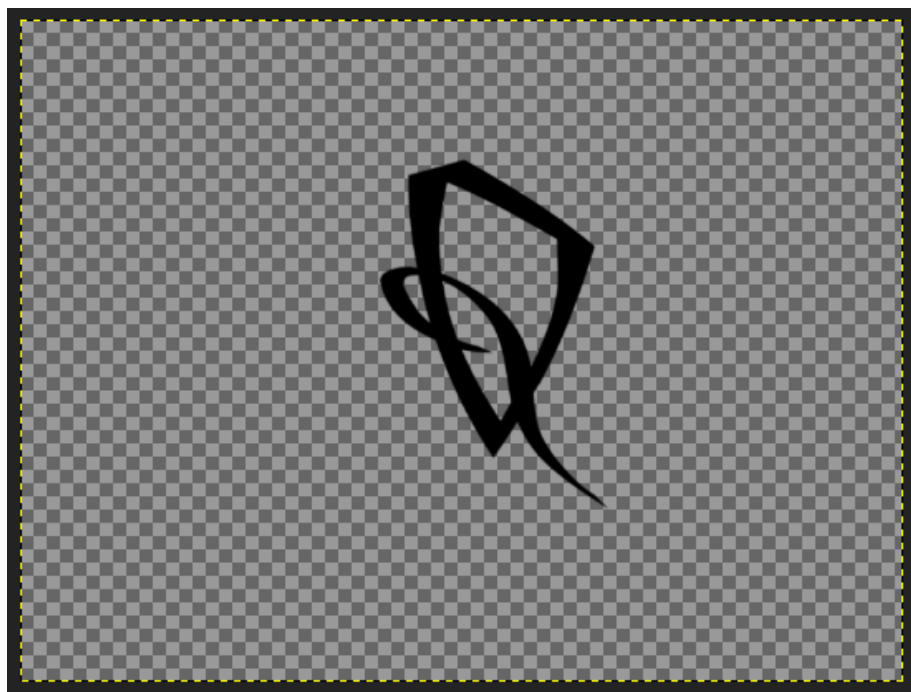
Start by creating the shape you want to transform. For this tutorial, I'm just using a single letter to keep the process simple. It doesn't even have to be text. Just be aware that you will be converting the shape into a path. Elaborate, complex shapes will create elaborate, complex paths for you to deal with.

Here's my starting point, “Q” in “A Charming Font”;



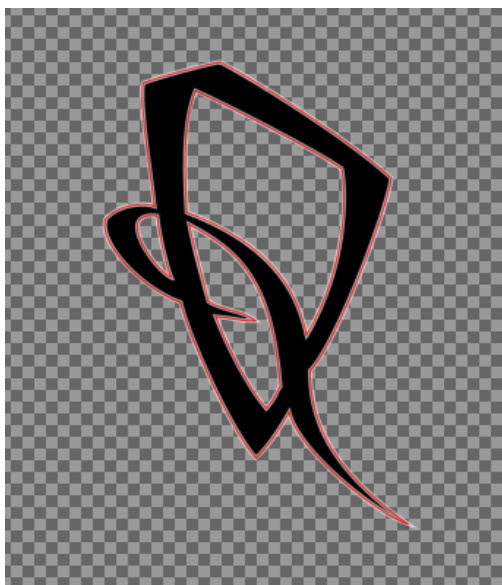
There are a few different ways to get from here to having a path in the perspective we want. In the piece that spawned this tutorial I used saulgoode's “warp text” script to transform the text into paths I had created with the correct perspective. I then had to go through the trouble of cleaning up the text path so that it was sharper and easier for the scripts to work with.

For this tutorial, I went a much simpler route. First, I used Image>Map>Map Object to map the letter to an angled plane. I don't think this creates true perspective, but it is good enough for small projects.



Then I manually created the path that outlines the letter. I did it manually because using “select to path” or other filters to create the path makes a hideously poor path. The time you spend cleaning up the resulting path is more than the time you'd spend making a path from scratch.

Here's our new “Q” with the path visible. If you aren't comfortable with the paths tool, consider this technique an excuse to practice. You can spend as much time perfecting the path as you like. Name this path “Face” or something similar so you know it's the face of the shape rather than the back.

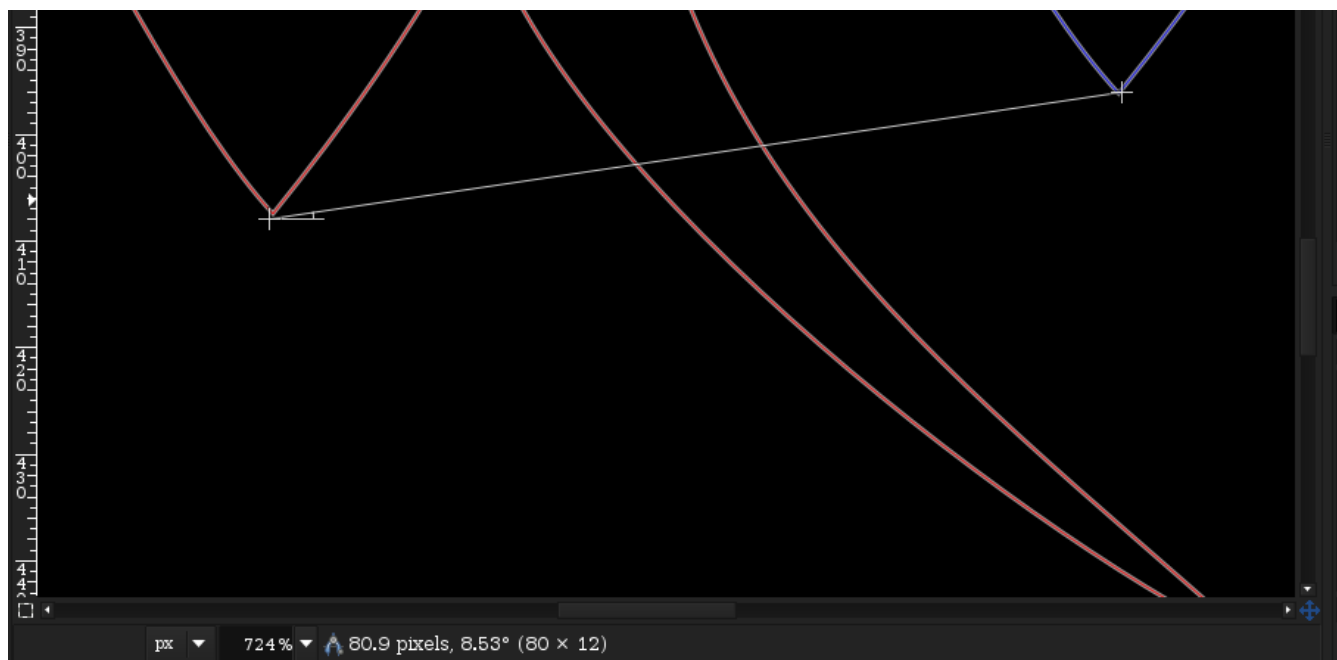


Now, duplicate this path, move the duplicate, scale it down. Rename the duplicate “rear” or something similar. In the original piece I transformed the rear path using vanishing points and guidelines just like you would draw perspective on paper. For this tutorial I'm using this simple method because we don't always need or want perfection. You get to decide how particular you want to be. Here are the two paths we now have to work with;

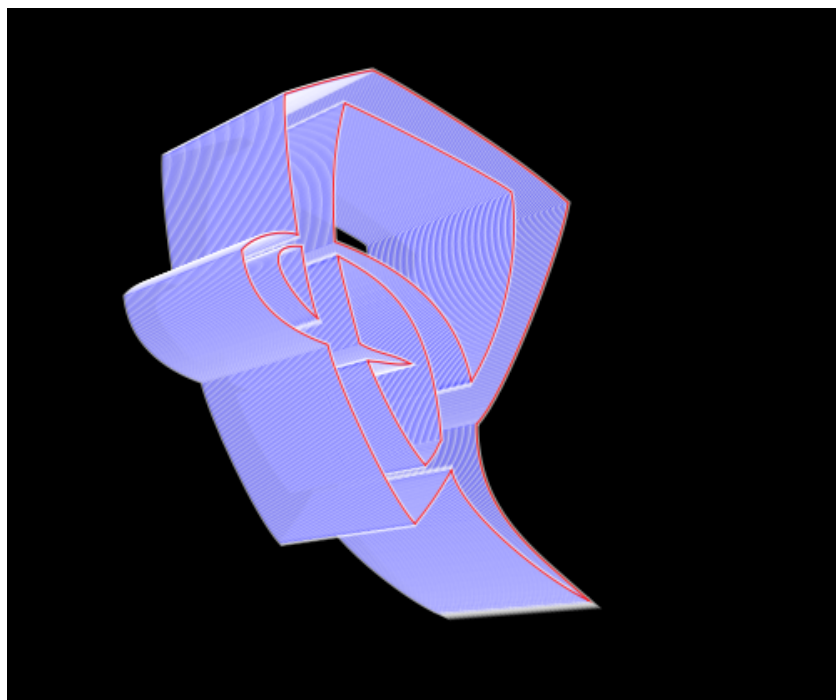


Even with just the two paths you can already see the effect coming together. But now is when we start to have fun. You need two of ofnuts' scripts to complete this project, his intermediate-paths script and his stroke-visible-paths script.

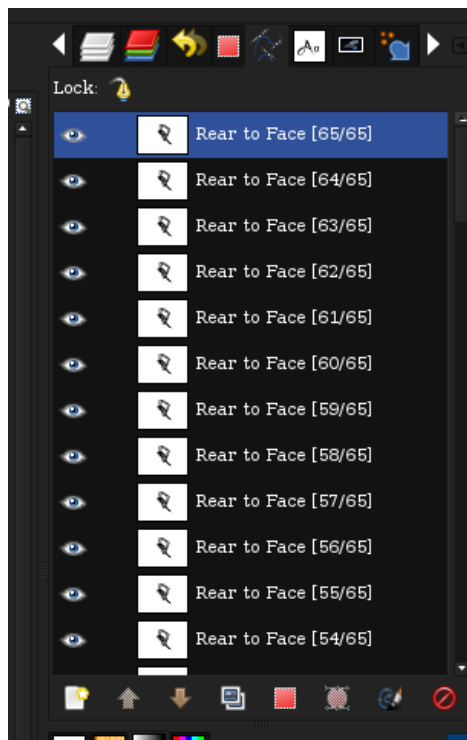
First, use the intermediate-paths script to fill the gap between the face and rear paths. The number of paths needed in the settings depends on how much distance is between the paths. Use the measure tool to measure the gap between two related points on the paths. To get a smooth finish you need the number of paths to be at least 75% of the number of pixels in the gap. In the original piece I had gaps that required hundreds of paths to fill. In this case, I have about 80 pixels to fill. So anything above 60 paths should be adequate.



Here's the result of the script;



Yes, it does generate all those paths;



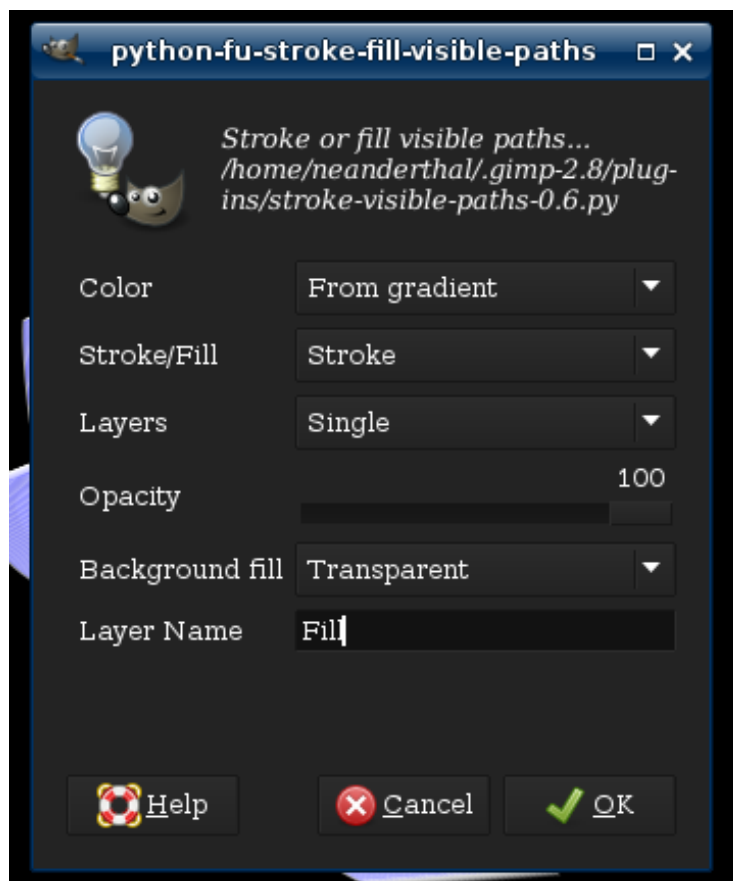
But don't worry. Those paths are the only visible paths. So when you are done with them it's very easy to simply “merge visible paths” so they become one single path in the dialog. Then you can delete that massive path or keep it to decompose it later if you think you might need it again. Personally, unless the path is large and takes a great deal of time for the script to create the intermediates I think it's just as simple to run the intermediate-paths script again if you need these paths later. So I merge and delete these.

Now we need the stroke-visible-paths script to turn all these paths into an object we can work with. The translucent look of the original piece was a happy accident. For this tutorial, I'm going directly to that point to save time.

****VERY IMPORTANT****

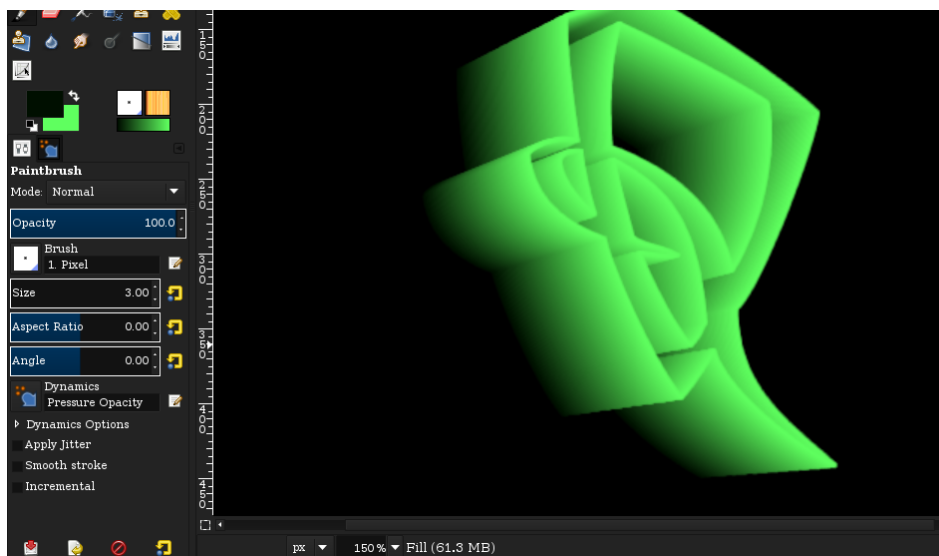
Before using the stroke-visible-paths script make sure you set the brush size in the brush dialog. It has to be the Pixel brush at its native 3x3 size. Anything else will produce a very messy result.

These are the settings I use;



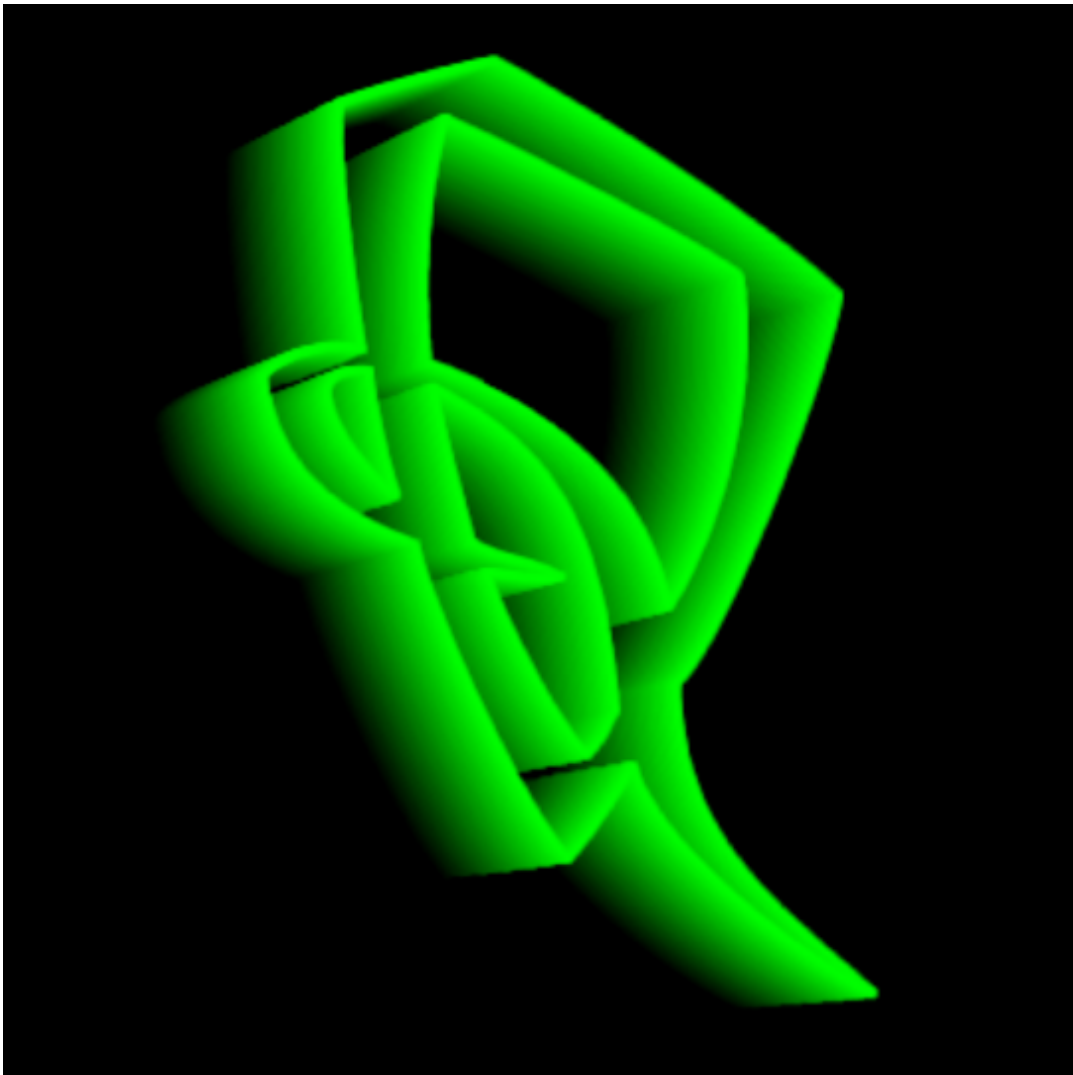
The stroke/fill setting should be obvious since we are going for the outline. The gradient setting, again, should be obvious since we are wanting to fade the shape into the background. The background fill needs to be transparent so you can see each layer through the others. At least, these are the right settings for this particular technique. You could very easily change the settings to use this technique to make a gif where the letter faded into or out of existence by making each layer opaque and playing them in sequence. Just a thought.

Here's the result of the script;

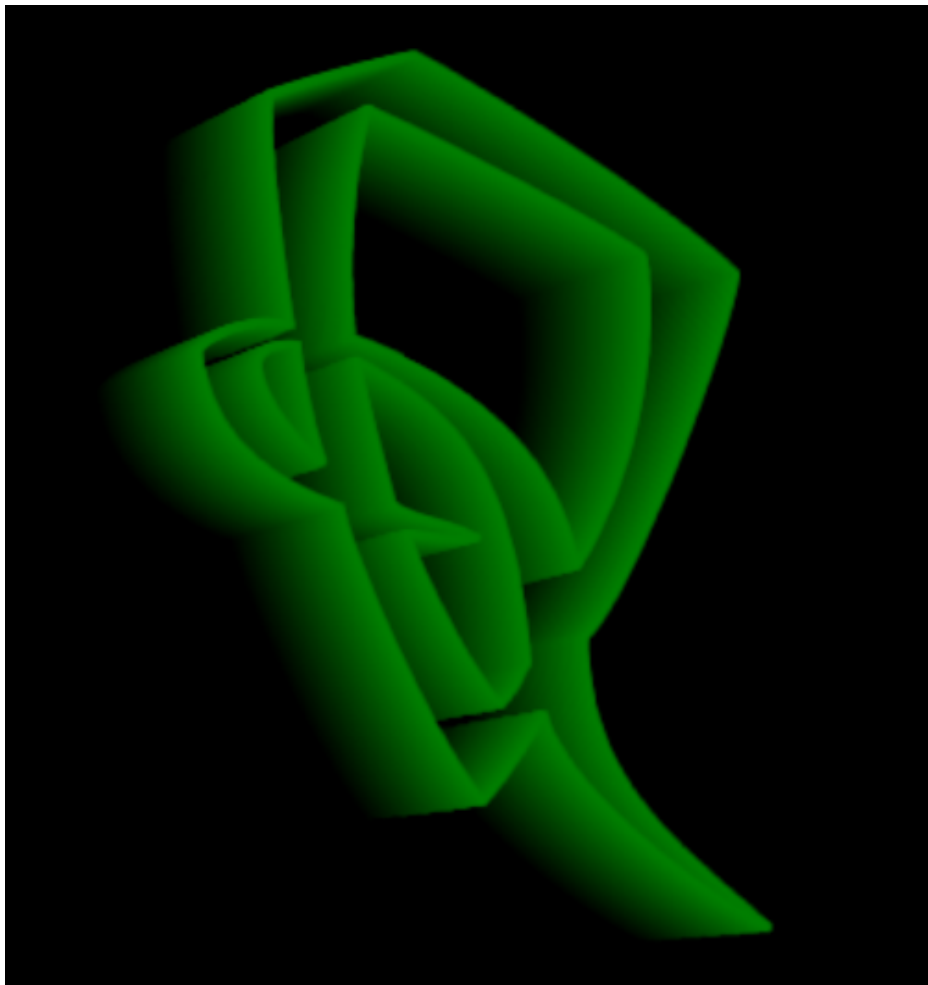


Note that when I ran the intermediate-paths script I ran it from the rear to the face. So the rear path is first in the stack. The stroke-visible-paths script starts at the “first” path and goes from there. And it chooses the gradient color from foreground to background as it progresses. So in order to get the rear of the “Q” to be dark I had to set the foreground as the darker color. The colors I chose were simply a very dark, pure green and a medium, pure green. That gives a very nice, smooth transition.

And here's the result with the layer mode set to “Hard Light”;



“Grain Merge” mode produces a softer result;



Other layer modes may be more useful to you depending on what your background image is or what effect you are going for. Also, the color you choose and the overall darkness of the strokes play a role in what mode is best for your piece.

I also need to mention that this tutorial is a bit simplified compared to the original piece because it is focused solely on producing the effect. In the original, each letter was its own layer and had been built from its own paths. That way I could adjust the opacity of each layer and shade between them to enhance the illusion of depth. With the single letter in this tutorial there's no way to show all that. And all that tweaking would vary greatly from one piece to the next, so it wouldn't necessarily be helpful in a tutorial anyway. So if you do use this on whole words or multiple shapes just keep each on its own layer and then tweak them until the overall image is what you want.

One last thing to finish out this tutorial. On the original piece I had a “neon” glow to the face of the letters. I used a script called “lightsaber 1.3.0” to do the glow. I tried ofnuts’ “neon” script, which creates a neon tube along a path, but it creates what looks like a neon tube. That isn’t quite the effect I want here, which is more like edge-lit plexiglass. So I’m going again with the “lightsaber” script. There are probably other scripts to do the same thing. There are certainly manual techniques that would work as well.

The lightsaber script works on a selection. So we have to first have something to select. I’m going after a sharp, clean look with clearly defined angles as though the edge of the letter is the source of the light. So the selected area has to be small. The easy way to do this is to stroke the “Face” path with a small (3 pixel) line on a new layer. Stroking the path with a line instead of a brush will give sharper angles. Then “alpha to selection” that layer to turn the stroke into a selection. Then simply run the lightsaber script on that selection and voila!



In closing, I want to be clear that this tutorial is just an overview of the technique. There are many ways to tweak this to produce different results. Different layer modes, different gradients, different gaps between face and rear layers, different effects on the face, different opacities... you know the routine. This isn't exactly the way I first did it. But with the exception of the perspective it is the way I'd do it next time I wanted this effect.