

Custom Font Bevel Reflect

Download the filter and extract it, if necessary, before saving it to your user plug-in folder. If you do not have the Bevel Reflect Logo script and Plug-in, they can be downloaded via links to GimpScripts.com.

<https://www.gimpscripts.net/search?q=Bevel+Reflect+Logo>

The accompanying zipped Resources folder contains two tester custom font files, with just three characters, for you to experiment with and a selection of emap image files (in case you have none to hand). Unzip the file and save the resources you want to use to a location of your choice

Sample result with highlights applied to existing decorated characters:



Sample result for characters, with a plain grey face, generated from an emap file:



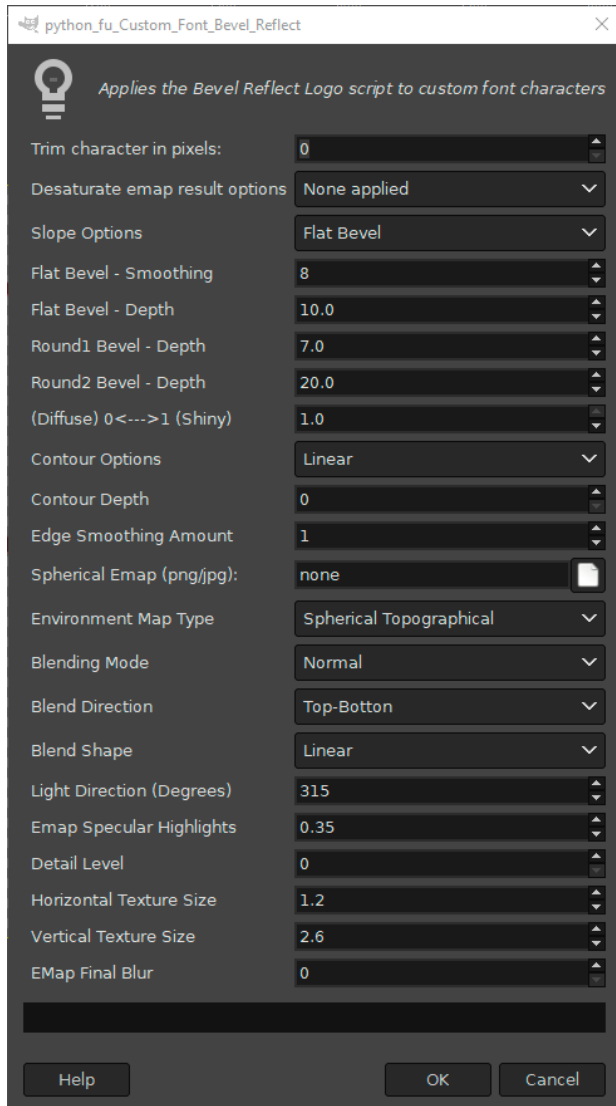
This custom font tool applies the bevel reflect logo script (version 1.7), which must be on your system, to all the characters in a custom font file. All the hard work done by GnuTux, originally in 2013, and laterally by MareroQ can now be used to add those delicious, highlighted bevels to your custom fonts.

One word of caution, however, is that it does take a fair amount of time for the original script to do its work: My machine took about 25 minutes to render 122 character layers. This obviously gives you time to catch up on other non-Gimp tasks, but my advice, as usual, is to experiment with two or three characters before using the filter with a full character set.

The filter dialog can be found at:

Filters>>Custom Font Tools>>Custom Font Bevel Reflect...

The filter dialog mirrors most of the options that you would expect to find in the original Bevel Reflect Logo script, in order to give you as many opportunities as possible to tweak your results. If you have used BRL in the past you should feel perfectly at home.



The first two options are those that I have added.

The trim pixels option allows you to tidy up the character edges, should this be needed, by removing pixels from the character edges. I tend to use a value of 3: This may sound like a drastic amount but if you use Custom Font Facemaker to add the same number of extra pixels, when decorating the original custom font character file, you will end up with the same character thickness.

The second option is really intended for use when adding highlights to an existing set of custom font characters. It basically desaturates the output from BRL before applying it with grain merge mode to each custom font character. This option allows you to choose which type of desaturation to use – depending on how dark the characters are to begin with.

Using the filter with Emap files

Make sure that your custom font characters all have a mid-grey colour for best results.

Open your chosen character file from the File menu.



An emap is simply a file that BRL can use in order to apply colours to, as in our case, a character layer. There are plenty of resources available to download from GimpChat – or you can create your own with any image and the Map Object filter using the sphere option on a transparent background.

Use only png images with a transparent background (or at least with a transparent pixel in the top left corner) unless you want to have a coloured background applied to all your characters.



This is the png emap that I am using which I keep in a folder named Emaps.

Call up the filter dialog:

Filters>>Custom Font Tools>>Custom Font Bevel Reflect...

Set the following options:

Trim option: 3

Slope Options: Round2 Bevel

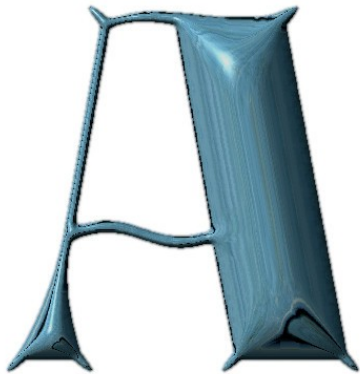
Emap Specular Highlight: 0.50

Detail Level: 3

Click on the Spherical Emap option folder icon and navigate to the location of your saved emap file and click on it.

Click on the dialog 'OK' button.

My outcome shown below:



You can also use the current gradient to generate a colour map within BRL.

Below I set the foreground and background colours to different reds and selected the FG to BG (RGB) as the gradient and used the same settings and grey character apart from the following options.

Click in the Spherical Emap option and delete the existing text before typing in the word 'none'. Set the Environment Map Type to Spherical Topo (Active Gradient)

My results:

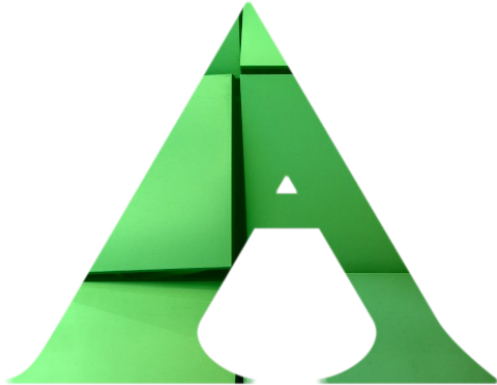


If you want to try out different settings, use Undo (Ctrl + Z) and have a play.

Using the filter with existing decorated characters

Open the file containing the decorated character/s from the File menu.

This is my test character, using an image from Unsplash (reminds me of a door-chime box we used to have) that has been coloured with a green gradient:



Select a light and dark tone colour for your foreground and then for your background colour; I used a light and a dark orange.

Select the foreground – background RGB gradient (FG to BG RGB) for the purpose of this guide. You could choose any gradient but the results could vary considerably from my outcomes.

Open the filter dialog once more.

If you have already used the filter with an emap, click on the Spherical Emap option folder icon, delete the text and replace it with the word 'none'.

Use the following settings:

Trim option: 3
Emap Specular Highlight: 0.50
Detail Level: 3

Select 'Desaturate Average' from the 'Desaturate emap result options'.

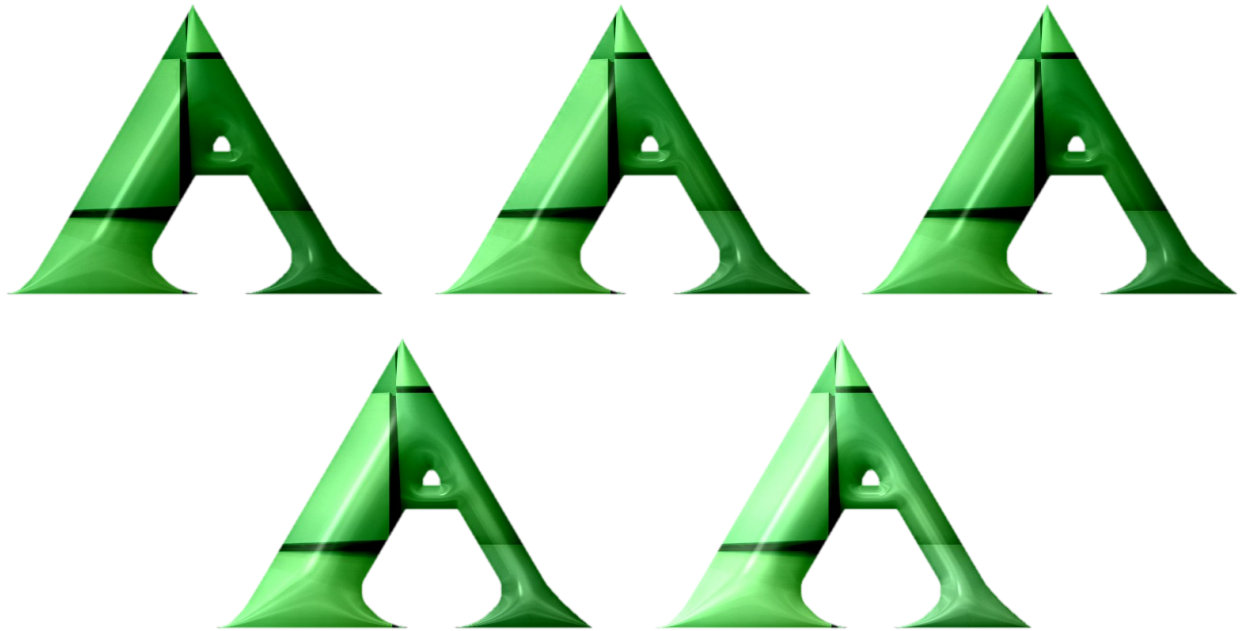
Select whichever of the 'Stroke Options' you prefer and click on the dialog 'OK' button.

The three slope options produced the results, in their order, shown below:



If you decide to keep the 'Desaturate map result options' as 'None applied' then the colours from your selected gradient will be grain-merged with the colours in the existing character.

Here are the results that I obtained with each of the desaturation options (in order):



This options provides a method of slightly lightening or darkening the effect on the images used by different custom font files.

There are plenty of other options to play with; 'Environment map type' and 'Contour Options' can make quite a difference to your outcomes.

Have GIMP? Have fun!

Thank you GnuTux and MareroQ.