


www.blender.nl

 this document is online at <http://www.blender.nl/showitem.php?id=20>

Tutorials

2000 08 02

id20

UV Mapping



Samo Korosec



Since Blender 2.0 - 'gameBlender' - was just around the corner, NaN added the UV texturing tool to the blender 1.x series to enable users "prepare" themselves for game content creation, by practicing UV texturing along with low-polygon count modeling. This tutorial was written for 1.8, but everything still works the same in 2.0.

Well, there might have been some confusion about the UV texturing, since Ton suggested to me today, that I should write a UV texturing tutorial :o) Here it is!



Final result.

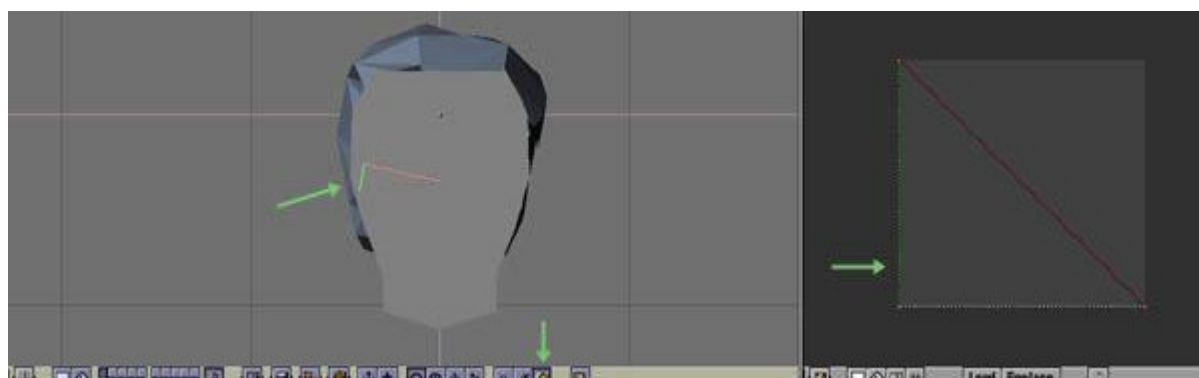


We start with the finished object and the image window opened.

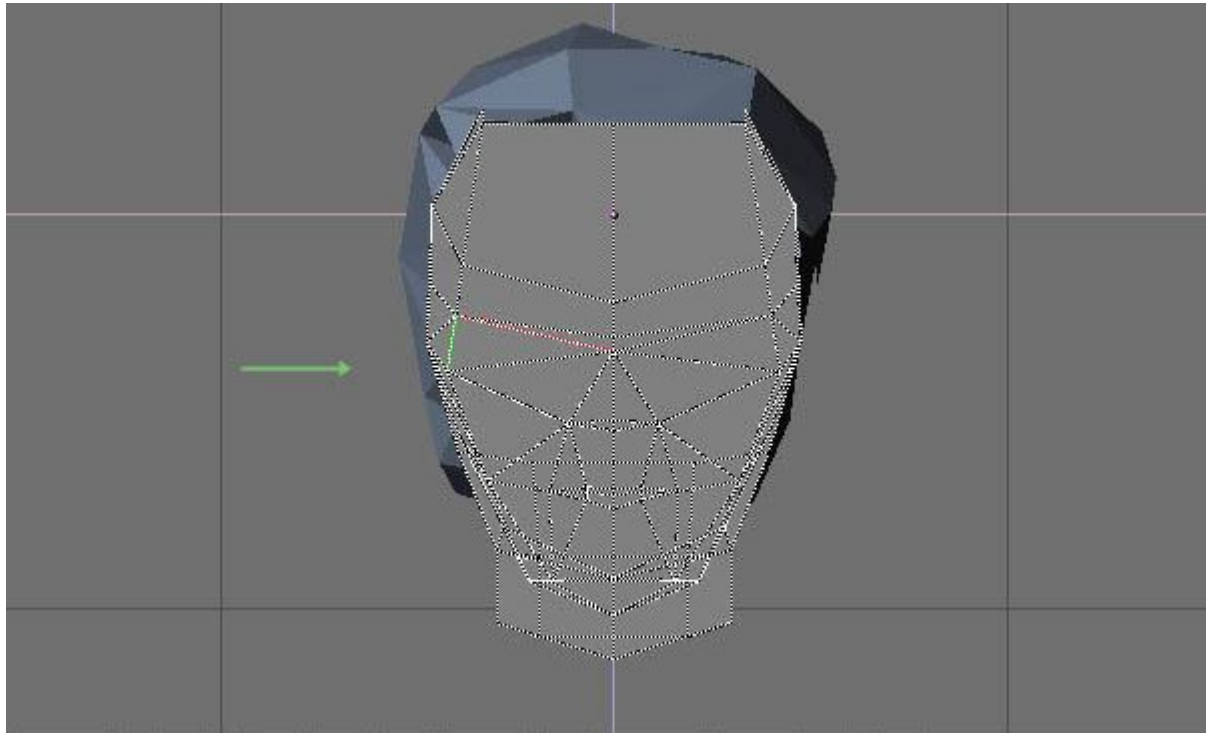
You will start off with a file you created your model in (be it a [face](#), character, vehicle or whatever). Once loaded, you will need two windows to work efficiently - the 3D window (Shift F5) and the image window (Shift F10).

The 3D window will represent your model, while the image window will show your [textures](#) and allow you manipulating the UV coordinates. The green arrows indicate the buttons that tell you the type of the window, the left being your 3D and the right your image window.

You can download an example .blend file for the [face](#) here.

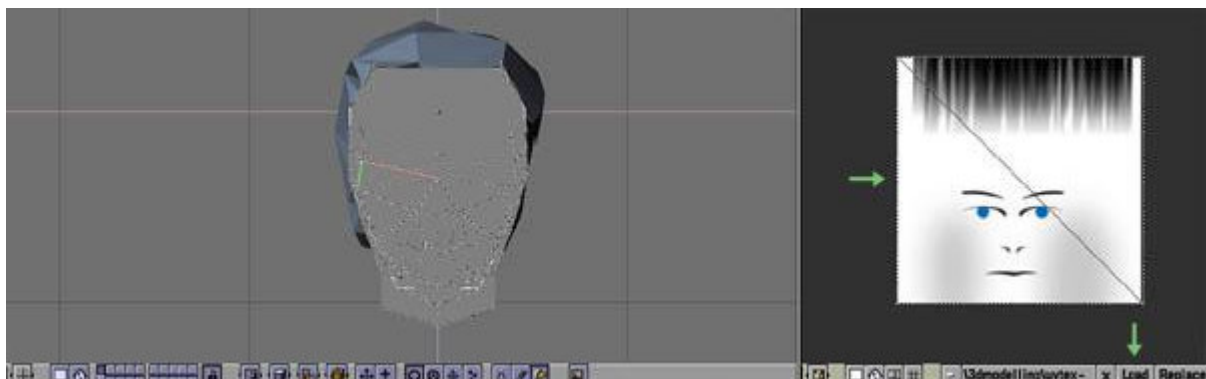
 Download:  [uvtextstart.blend](#)


With your object selected go into face select mode.



Select all vertices.

However we do want to map our whole **face** model and instead of selecting each **face**, we **select** all by pressing **A**. However, a **face** must be selected before that anyway, to appear "active" (see the green arrow), as some other **face** attributes must have a origin - the active **face**.



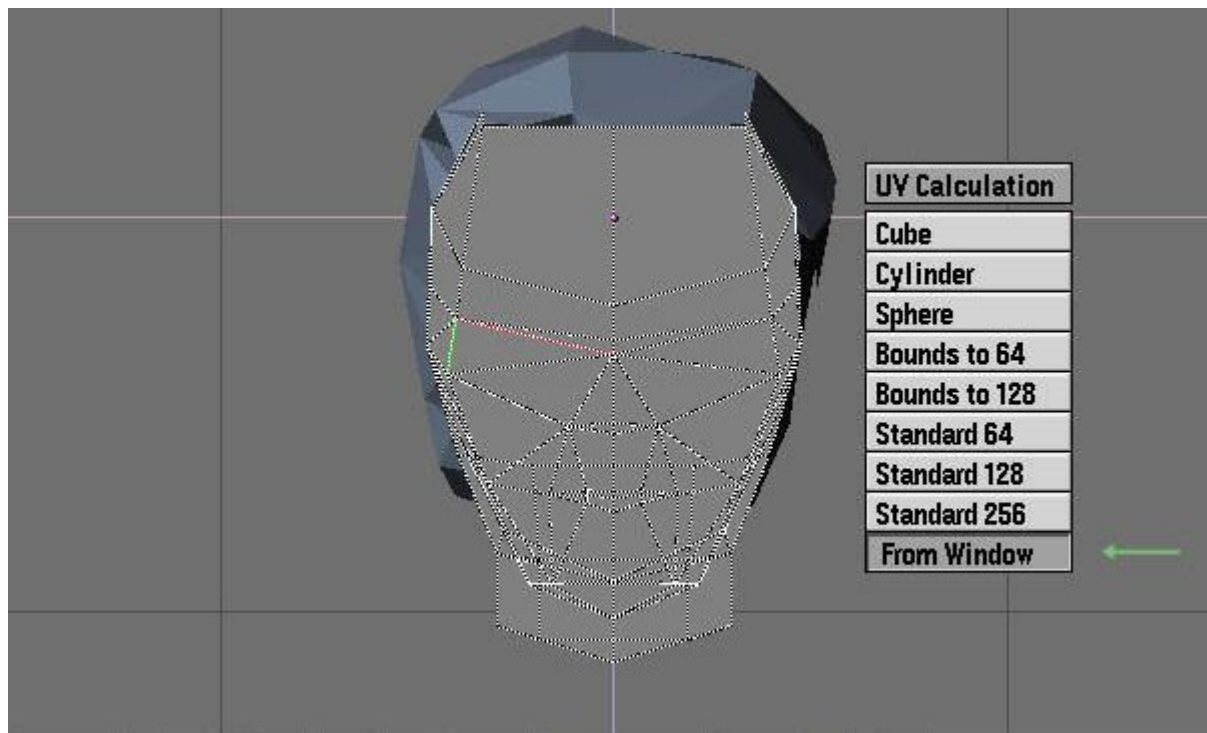
Load an image in the image window.

The next step you need to do is to actually load the **texture** (the green arrows indicate the '**load**' button and the area where the **texture** should appear), where it's width and height have to be a power of two and also have the same dimension. Valid examples would be image sizes like: 8x8, 16x16, 32x32, 64x64, 128x128, 256x256, 512x512,...

This is due to the way **OpenGL** handles textures; you could use a **texture** with its height and width not being the same, however most **OpenGL** hardware (especially games adapters) will then stretch the **texture** to fit in their memory.

(You can download the **texture** here:)

Download:  [uv_map.jpg](#)



Map UV coordinates.

Once you have your **faces** selected, you need to load their UV coordinates into the image window. To do so, you need to press **U** and choose one of the options presented to you. I chose "From Window" - that used the 3d window to map the UV coordinates - hence, the image window contained a "front" view of the objects' coordinates.

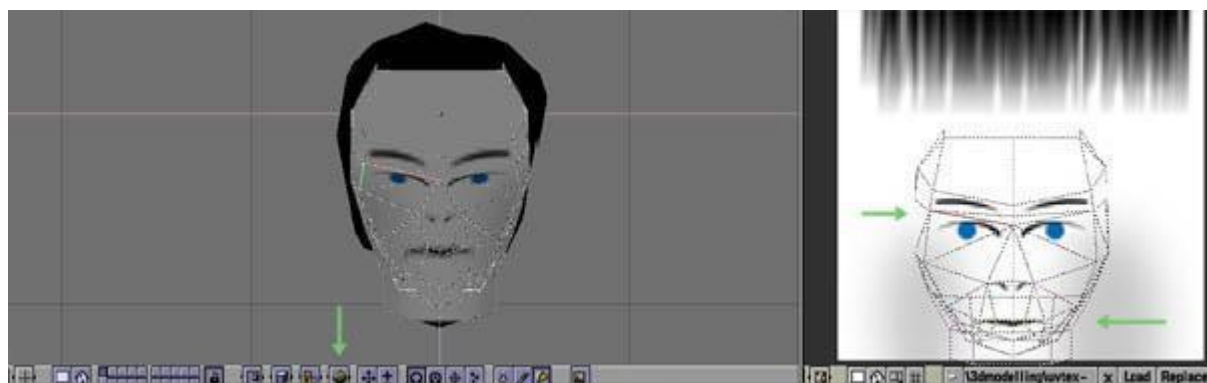
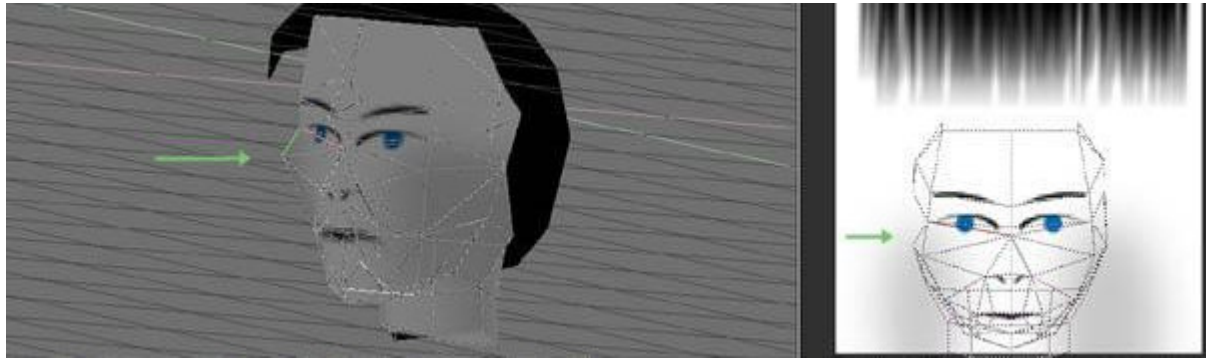


FIG. 6 - YOU CAN NOW MOVE THE VERTICES IN THE IMAGE WINDOW TO ADJUST MAPPING

You can now move the vertices in the image window to adjust mapping.

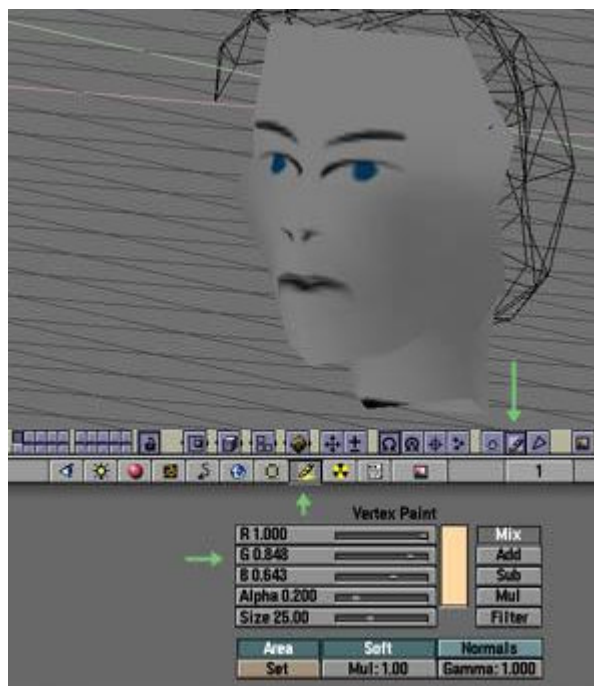
You need to turn on the textured view (left green arrow) now, in order to see the adjustments of the UV coordinates you perform (right arrows). This should give you the first UV texturing results.

Editing the UV coordinates works the same way as editing normal **vertices** in **edit mode**. You can play around by selecting a group of the UV coordinates and scaling, rotating or moving them.



Fine tuning your mapping.

However, you might soon find yourself in a situation, where you are not sure what impact your UV editing actions have - feel free to rotate the 3D view, zoom it in and out etc. You can also move and zoom the image windows, allowing very detailed mapping work.



Go to vertex paint mode.

Once we have our **textures** in place, our friend still looks kind of sick - plain gray is no color for a face!

Exit **face select** mode by pressing **f** and enter **vertex paint** mode by pressing **U** or clicking the button indicated by the right arrow. The other arrows indicate the **vertex paint** buttons button (funny, eh?) and the sliders to choose the right colors.

While in vertex-paint mode, your left mouse button lets you paint on object, while your right mouse button (thanks, Rob!) lets you choose colors beneath the mouse cursor. I tried to get a skin-like color and just painted away.



Vertex painted mesh with UV texturing

After finishing with [vertex](#) painting, we can enjoy our new-textured [head](#). It looks very clean and nice with black hair, however I added a "hair" pattern sort of on top of my [face texture](#) and you can use it..



Repeat the same for the hair and enjoy the result.

..to make a punk! :o)

I hope this little tutorial was helpful.