

Polybump Plugin for 3DStudio Max

(V2.93, last changes 09/14/2005 MM – FY040112)

Polybump™

© 2001-2005 Crytek GmbH

Abstract:

PolybumpPlugin is the name of a new utility for 3DStudio MAX R4 and higher which uses the Polybump™ technology. This technology allows you to create real-time models with **normal maps**¹. With normal maps you are able to store lighting information in the texels² and use this information for **Dot-product per pixel lighting**. This is different from traditional lighting, where the information is stored per vertex.

With the normal vector information stored in the texture you can have much better looking models with less polygons at all tessellations. Less polygons equals less memory traffic, and faster frame rates.

Hardware support for normal maps has been available for some time, but they were rarely used, partly because of a lack of tools.

Our plug-in creates the normal maps from a given high-resolution model (e.g. 200000 triangles). The plugin is also able to extract further information in order to darken less accessible parts. This results in more natural looking light and less work for artists, as they no longer need to paint shadows.

Normal maps are supported by various graphics hardware (NVidia GeForce1-3, ATI Radeon 7500-8500, ST KyroII, Matrox Parhelia). Even without hardware support, the information can be used to pre-calculate a lighting condition.

¹ A normal map is a simple texture map in which the surface normals are stored as colours.

² A texel is the base element of a texture map (sometimes called pixel).

Installation:

First make sure that you have at least 3DStudioMAX R4.x installed, but not running. Now start the installation application [PolybumpInstaller.exe] and follow the on-screen instructions. Make sure that you specify your 3DStudio MAX folder as the installation directory. When you restart 3DStudio MAX a new plugin should be available through the Utilities tab (the hammer symbol).

Known Limitations:

- This plugin works only with 3DStudio MAX R4 or higher

Quick start:

Prepare the scene: Place the high-resolution model and the low-resolution model in your scene at the same position. Make sure the low-resolution model contains UV coordinates and smoothing groups. Also make sure that the normals of the mesh are outward looking.

Activate the plugin: Switch to the Utilities tab (hammer symbol), add the plugin to your utilities set and activate it.

Select the source: Select the high-resolution model and press “HighPoly”. Now select the low-resolution model and press “LowPoly”. Specify the “Channel No” (in almost all cases #1) and “Material ID” (otherwise the whole object is used).



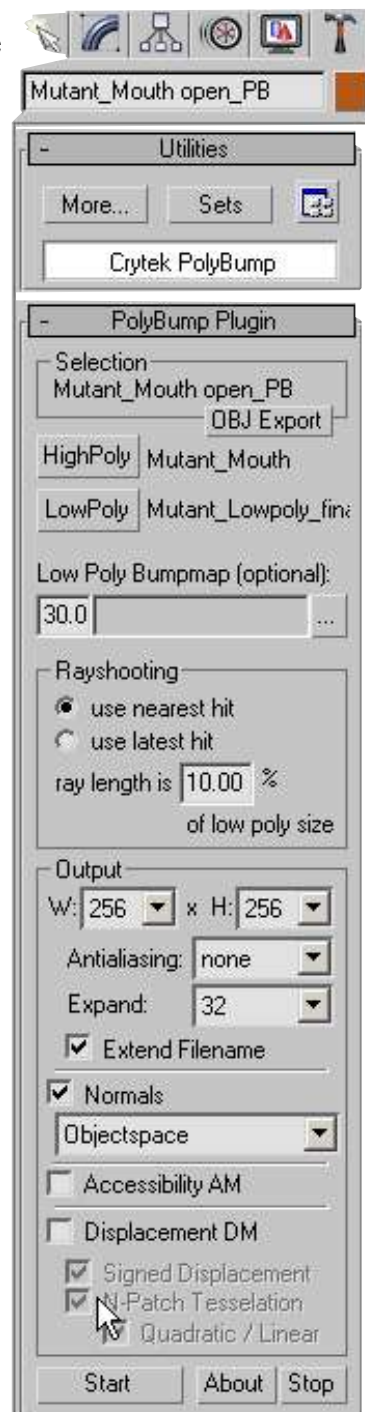
Choose the output quality: Select a texture size and activate “Antialiasing” for increased quality.

Start the calculation: Press “Start” and choose a path and filename for the normal map. This may be any standard RGB file format.

Wait or work: While the plug-in is working, a progress window is shown. The calculation only

uses free processing power so you can use

MAX as usual while it is running. NB. Ending MAX will stop the plug-in. Use the “Stop” button if you want to stop or restart the calculation.



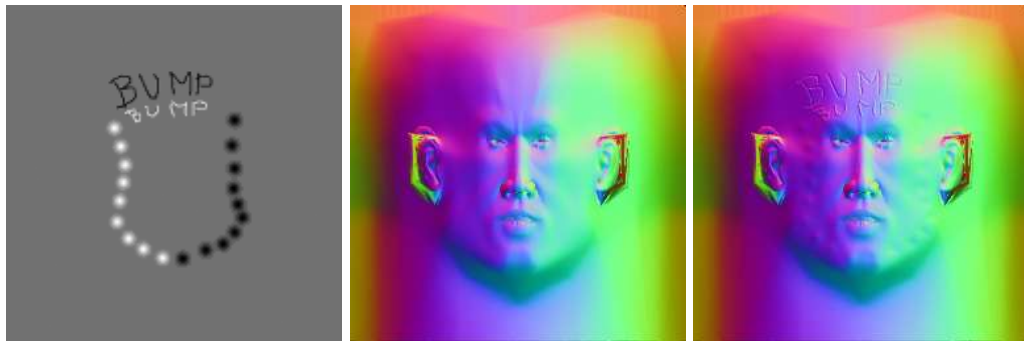
Using the Plugin:

Check the normals: The result is affected by the normal direction of both the low-detail and the high-detail model. Make sure that both models have their normals directed outward.

After mirroring an object in 3DStudio MAX, normals become flipped, even though this anomaly is not visible. Using the Reset X-Form utility will visibly flip the normals on the object if it has been mirrored. You need to flip them back if this happens. If nothing changes after using the Reset X-Form utility then your model is ok.

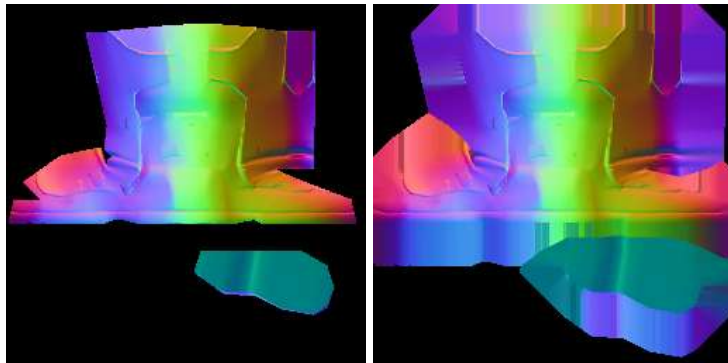
Unique mapping is often necessary: Every surface position should correspond to only one surface position. You are allowed to overlap triangles in the texture only if they would generate the same colours. Using the texture with wraparound is not allowed.

Effects of applying a bump map: The bump map has to be specified as a Targa (TGA) image. The texture coordinates are taken from the low-resolution mesh. Dark pixels on the bump map are treated as pits. Using a bump map allows you to create surface details that are difficult to model. To deselect a bump map, click on the “...” button and press cancel.

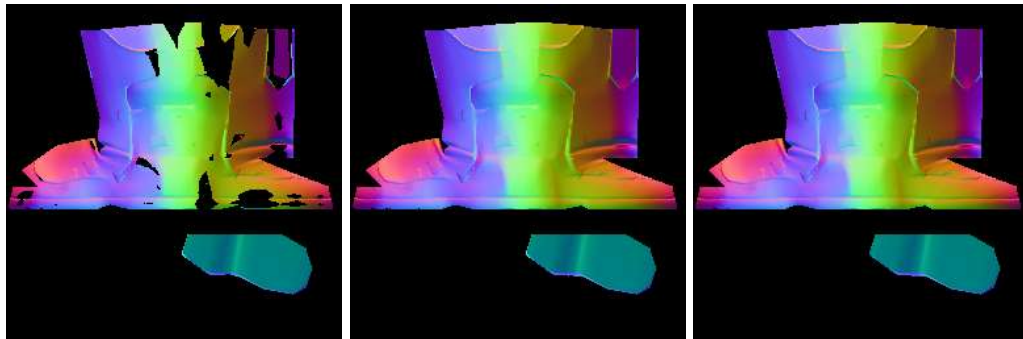


Apply the correct smoothing groups: The smoothing groups of the low-detail mesh affect the direction where the corresponding surface point on the high-detail mesh will be found. Very often one smoothing group will give good results. Depending on the nature of the tangent-space you might use more smoothing groups to get better results.

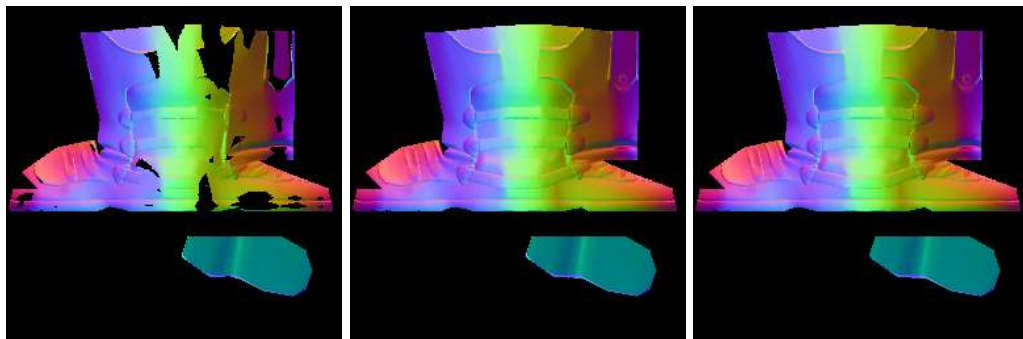
How “Expand” affects the result: In order to avoid seams, the result can be automatically expanded by 30 pixels. Normally 1 or 2 pixels will remove seams, but further expansion will avoid the same problems for mip-mapping.



How the ray length and “latest hit” / “nearest hit” affects the result: On the following pictures you can see the difference between the use of different ray lengths.



“use nearest hit” with 2%, 5% and 50%



“use latest hit” with 2%, 5% and 50%

If you choose “use latest hit” you may end up with values that are too large. You have to set the value carefully, you may want to set different values for different parts of the model (use Material ID). In this case you will have to combine the textures yourself.

Using the accessibility texture: With the accessibility map, you are able to improve your lighting even further. This texture is white for fully accessible surfaces (within the range given by the ray length) and grey to black for less accessible surfaces. You could use this information:

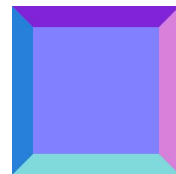
- As a good starting point for your texture mapping, because it makes it easy to see where the details are.
- For dirt shading (use the information to mix between two materials).
- To modify the normal map (add a 50% grey layer, copy the accessibility map to the selection, add layer mask – hide selection). This will darken less accessible surfaces a little and give you a simple hemisphere/radiosity lighting approximation.

How the normal is coded as colour (objectspace): Each colour in the normal map represents a normal vector. The three-colour components are directly associated with the three components of the vector.

Positive x-axis:	100% Blue
Positive y-axis:	100% Green
Positive z-axis:	100% Red
Negative x-axis:	50% Green, 50% Red
Negative y-axis:	50% Blue, 50% Red
Negative z-axis:	50% Green, 50% Blue

How the normal is coded as colour (tangentspace): Each colour in the normal map represents the modification of the low-resolution normal vector. The three-colour components are directly associated with the three components of the vector.

No modification:	100% Blue, 50% Green, 50% Red
Rotate to the left:	dark cyan
Rotate to the right:	violet
Rotate upward:	lilac
Rotate downward:	bright cyan



This is a raised button in tangent-space

How the normal is coded as colour (clonespace): The colour in clonespace is encoded the same way as in objectspace. The colour is calculated for a specific surface point on the low-resolution mesh. The necessary base vectors can be calculated for scaled/rotated/translated/mirrored triangles with information in the .

CLN file. As a registered user you will get a technical document where this process is described in detail.

Hints:

- It is not necessary to put the entire low-resolution model into the high-resolution model and vice versa
- Calculating the texture at a higher resolution, then down sampling to the required size could improve your results. You should use the “Texture Expand” feature to avoid filtering with black pixels.
- Tiny **gaps** on the mesh should not be a problem if you use the “Expand” feature
- Ray shooting **nearest hit** means the plugin gets the normal from the high-detail object at the nearest position measured from the corresponding low-detail mesh
- Ray shooting **latest hit** means that the plugin gets the normal from the latest hit at the high-resolution model. For this mode the **ray shooting length percentage** is very important
- The **ray shooting length percentage** refers to the bounding sphere of used triangles (Material ID) of the low detail mesh (=100%)

Version History:

07/16/2002 V 2.00

First public build, Evaluation version on Crytek website

08/02/2002 V 2.10

Integrate displacement mapping output, changed colour scheme for tangentspace (Red => Blue)

09/05/2002 V 2.50

changed colour scheme for also for clonespace and objectspace (Red <=> Blue), major speed improvements

10/08/2002 V 2.52

fixes rarely occurring endless loop (this could also cause artefacts)

11/12/2002 V 2.53

fixed a bug that caused tiny holes in the output

11/21/2002 V2.6

accessibility calculation is now affected by the specified bump map, fixed bug that removes the mesh assignment, multiple selections for the high poly mesh

02/19/2003 V2.76

additional bump map is now also affecting the accessibility calculation
adjustable antialiasing and expand
improved antialiasing, tangent space calculation and raycasting speed
new feature: export as OBJ, multiple selection for the high poly mesh
Fixed bug that removed the mesh assignment
Fixed bug in loading the greyscale tga files

04/29/2003 V2.9

improved tangentspace calculation (mirror seams are now almost invisible)

04/29/2003 V2.92

minor internal changes, cleaned up info dialog
fixed non-uniform scaling (normal was wrong)
default setting is now tangentspace

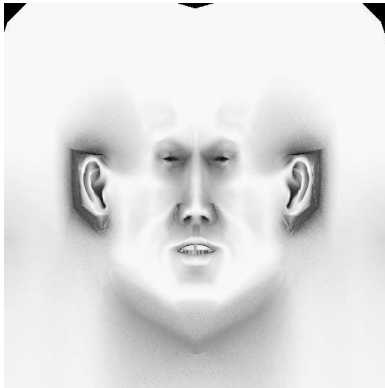
09/14/2005 V2.93

minor internal changes, cleaned up installation dialog

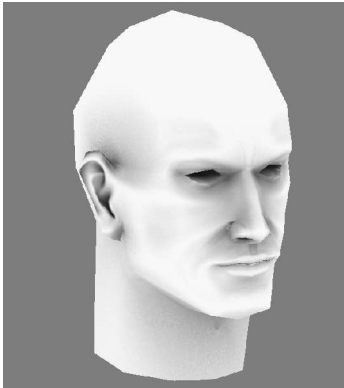
Examples:



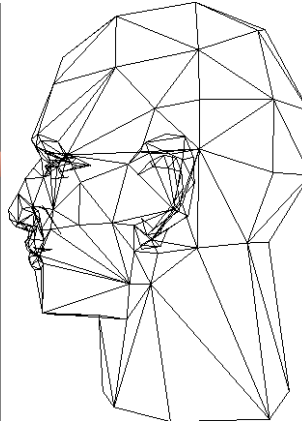
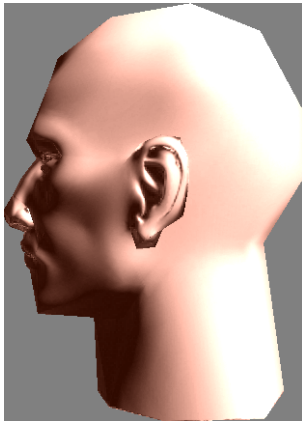
normal map



accessibility map



accessibility map applied to the low-resolution model



How to contact Crytek:

Crytek GmbH
Rosenauer Str. 16
96450 Coburg
Germany

www.crytek.com

If you have any question regarding Polybump please contact polybump@crytek.de