10.3.4.5 Render - Cycles Render Engine - Nodes - Texture Nodes

Texture Nodes

Image Texture

Image texture from GoodTextures.com

Use an image file as a texture.

Image Data-Block
Image data-block used as the image source. Currently not all images supported by Blender can be used by Cycles. In particular, generated, packed images or animations are not supported currently.

Projection

Projection to use for mapping the textures.

- *Flat* will use the XY coordinates for mapping.
- *Box* will map the image to the 6 sides of a virtual box, based on the normal, using XY, YZ and XYZ coordinates depending on the side.
- *Sphere* will map the image to the sphere using Z axis as central.

- *Tube* will map the tube to the sphere using Z axis as central.

Projection Blend
For Box mapping, the amount to blend between sides of the box, to get rid of sharp transitions between the different sides. Blending is useful to map a procedural-like image texture pattern seamlessly on a model. 0.0 gives no blending; higher values give a smoother transition.

Color Space
Type of data that the image contains, either Color or Non-Color Data. For most color textures the default of Color should be used, but in case of e.g. a bump or alpha map, the pixel values should be interpreted as Non-Color Data, to avoid doing any unwanted color space conversions.

Extension Type

Extension type defines how the image is extrapolated past the original bounds:

- *Repeat* will repeat the image horizontally and vertically giving tiled-looking result.
- *Extend* will extend the image by repeating pixels on it's edges.
- *Clip* will set all the extended pixels values to transparent black.

Vector input
Texture coordinate for texture lookup. If this socket is left unconnected, UV coordinates from the active UV render layer are used.

Color output
RGB color from image. If the image has alpha, the color is premultiplied with alpha if the Alpha output is used, and unpremultiplied or straight if the Alpha output is not used.

Alpha output
Alpha channel from image.

Environment Texture

HDR image from OpenFootage.net

Use an environment map image file as a texture. The environment map is expected to be in Latitude/Longitude or 'latlong' format.

Image Data-Block
Image data-block used as the image source. Currently not all images supported by Blender can be used by Cycles. In particular, generated, packed images or animations are not supported currently.

Color Space
Type of data that the image contains, either Color or Non-Color Data. For most color textures the default of Color should be used, but in case of e.g. a bump or alpha map, the pixel values should be interpreted as Non-Color Data, to avoid doing any unwanted color space conversions.

Vector input
Texture coordinate for texture lookup. If this socket is left unconnected, the image is mapped as

environment with the Z axis as up.

Color output
RGB color from the image. If the image has alpha, the color is premultiplied with alpha if the Alpha output is used, and unpremultiplied if the Alpha output is not used.

Alpha output
Alpha channel from image.

Sky Texture

Sky Texture

Procedural Sky texture.

Sky Type
Sky model to use (Preetham or Hosek / Wilkie).

Sun Direction
Sun direction vector.

Turbidity
Atmospheric turbidity. (2: Arctic like, 3: clear sky, 6: warm/moist day, 10: hazy day)

Ground Albedo
Amount of light reflected from the planet surface back into the atmosphere. (RGB 0,0,0 is black, 1,1,1 is white).

Vector
Texture coordinate to sample texture at; defaults to Generated texture coordinates if the socket is left unconnected.

Color output
Texture color output.

Noise Texture

Noise Texture with high detail

Procedural Perlin noise texture, similar to the Clouds texture in Blender Internal.

Vector input
: Texture coordinate to sample texture at; defaults to Generated texture coordinates if the socket is left unconnected.

Scale input
: Overall texture scale.

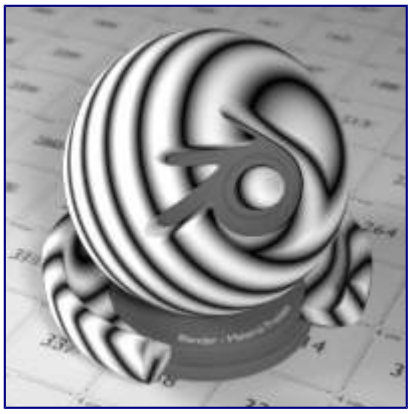
Detail input
: Amount of noise detail.

Distortion input
: Amount of distortion.

Color output
: Texture color output.

Fac output
: Texture intensity output.

Wave Texture



Default wave texture

Procedural bands or rings texture with noise distortion.

Type
: *Bands* or *Rings* shaped waves.

Vector input
: Texture coordinate to sample texture at; defaults to Generated texture coordinates if the socket is left

unconnected.

Scale input
Overall texture scale.

Distortion input
Amount of distortion of the wave (similar to the Marble texture in Blender Internal).

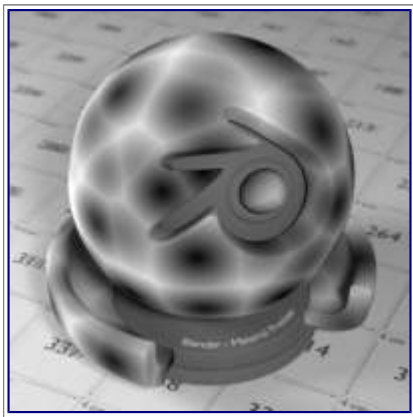
Detail input
Amount of distortion noise detail.

Detail Scale input
Scale of distortion noise.

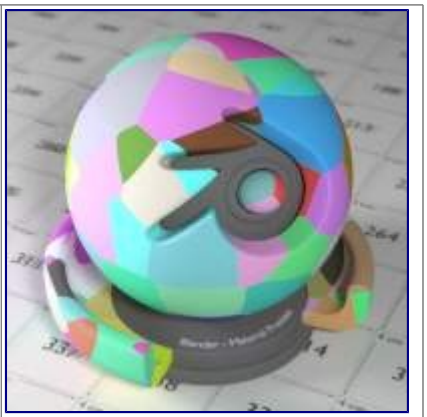
Color output
Texture color output.

Fac output
Texture intensity output.

Voronoi Texture



Voronoi texture, type: Intensity



Voronoi texture, type: Cells

Procedural texture producing Voronoi cells.

Type
Intensity or *Cells* output.

Vector input
Texture coordinate to sample texture at; defaults to Generated texture coordinates if the socket is left unconnected.

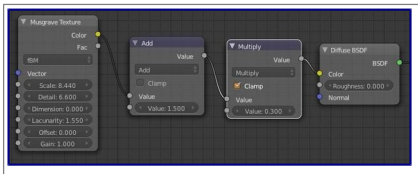
Scale input
Overall texture scale.

Color output
Texture color output.

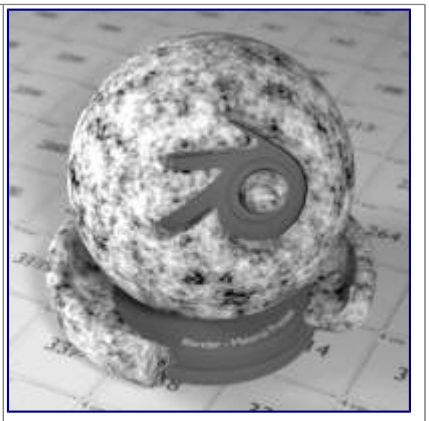
Fac output
Texture intensity output.

Musgrave Texture

Advanced procedural noise texture. Note that it often needs some adjustments (multiplication and addition) in order to see more detail.



Nodes for the image to the right



Remapped Musgrave texture such that most values are visible

Type
: Multifractal, Ridged Multifractal, Hybrid Multifractal, fBM, Hetero Terrain.

Vector input
: Texture coordinate to sample texture at; defaults to Generated texture coordinates if the socket is left unconnected.

Scale input
: Overall texture scale.

Detail input
: Amount of noise detail.

Dimension input
: The highest fractal dimension, specified as the highest scale for the steps of the intensity.

Lacunarity input
: The space of the lacunarity, specified as a frequency factor.

Offset input
: The offset of the fractal, specified between black and white values (Intensity)

Gain input
: A multiplier for the gain input

Color output
: Texture color output.

Fac output
: Texture intensity output.

Gradient Texture

Gradient texture using object coordinates

A gradient texture.

Type
: The gradient can be *Linear*, *Quadratic*, *Easing*, *Diagonal*, *Spherical*, *Quadratic Sphere* or *Radial*.

Vector input
: Texture coordinate to sample texture at; defaults to Generated texture coordinates if the socket is left unconnected.

Color output
: Texture color output.

Fac output
: Texture intensity output.

Magic Texture

Magic texture: Depth 10, Distortion 2.0

Psychedelic color texture.

Depth
: Number of iterations.

Vector input
: Texture coordinate to sample texture at; defaults to Generated texture coordinates if the socket is left unconnected.

Distortion input
: Amount of distortion.

Color output
: Texture color output.

Fac output
: Texture intensity output.

Checker Texture

Default Checker texture

Checkerboard texture.

Vector input
: Texture coordinate to sample texture at; defaults to Generated texture coordinates if the socket is left unconnected.

Color1/2 input
: Color of the checkers.

Scale input
: Overall texture scale. The scale is a factor of the bounding box of the face divided by the scale. For example, a scale of 15 will result in 15 alternate patterns over the overall UV bounding box. Different patterns could be achieved using other nodes to give different input patterns to this socket. For example, using the Math Node.

Color output
: Texture color output.

Fac output
: Checker 1 mask (1 = Checker 1).

Brick Texture

Brick texture: Colors changed, Squash 0.62, Squash Frequency 3.

Procedural texture producing Bricks.

Options

Offset

Determines the brick offset of the various rows.

Frequency
Determines the offset frequency. A value of 2 gives a even/uneven pattern of rows.

Squash
Amount of brick squashing.

Frequency
Brick squashing frequency.

Sockets

Color 1/2 and Mortar
Color of the bricks and mortar.

Scale
Overall texture scale.

Mortar Size
The Mortar size; 0 means no Mortar.

Bias
The color variation between Brick color 1 / 2. Values of -1 and 1 only use one of the two colors; values in between mix the colors.

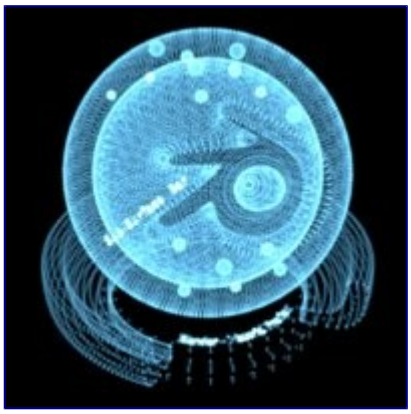
Brick Width
The width of the bricks.

Row Height
The height of the brick rows.

Color output
Texture color output.

Fac output
Mortar mask (1 = mortar).

Point Density



Domain object with Point Density texture using vertices from ball as points.

Used to add volumetric points for each particle or vertex of another object.

Options

Point Data

Where to get points from.

Particle System

Use each particle position from the specified particle system.

Object Vertices
Use each vertex position from the specified object.

Object
Which object's vertices or particle system will be used.

Particle System
Particle positions from this system will be used.

Space

The coordinate system for mapping points.

World Space
Map each point exactly where the source particle/vertex is.

Object Space
Fit the points from the source particles/vertices inside the bounding box of the object with the point density texture. .. TODO As far as I can tell this is how it works, but should be checked with a developer.

Radius
Radius from the shaded sample to look for points within. .. TODO Same as tooltip, this does not make much sense to me.

Interpolation

Texel filtering type.

Closest
No interpolation, use nearest texel. Produces blocky looking points.

Linear
Interpolate linearly between texels, producing soft, round points.

Cubic
Use cubic falloff, producing very soft points. Useful when points are very densely packed.

Resolution
The dimensions of the texture holding the point data.

Color Source
Which attribute of the particle system is used to color the output.

Sockets

Vector
Texture coordinate to sample texture at; defaults to global position (Position output of Geometry node) if the socket is left unconnected.

Color output
Texture color output.

Density output
Density of volume.