TEXTURE GRAPH

User Manual

0.1b

OVERVIEW

Texture Graph is a node based editor for creating seamless, high quality textures procedurally, which is built directly into the Unity Editor.

It is heavily focused on creating 2D textures with a non-linear, non-destructive and parametric approach, which mean you can create an image in many different ways, and switch from one to another for experiment without losing any of your work. You can also control any aspect of the image just by adjusting the parameters in the user interface. This way you can have endless creativity within just a single file.

This User Guide is created to help you get started with the tool, including the most basic information about its features and workflow, as well as best practices, tips and tricks from its creator and showcasing amazing work from other successful developers.

To see the release log, please visit this page.

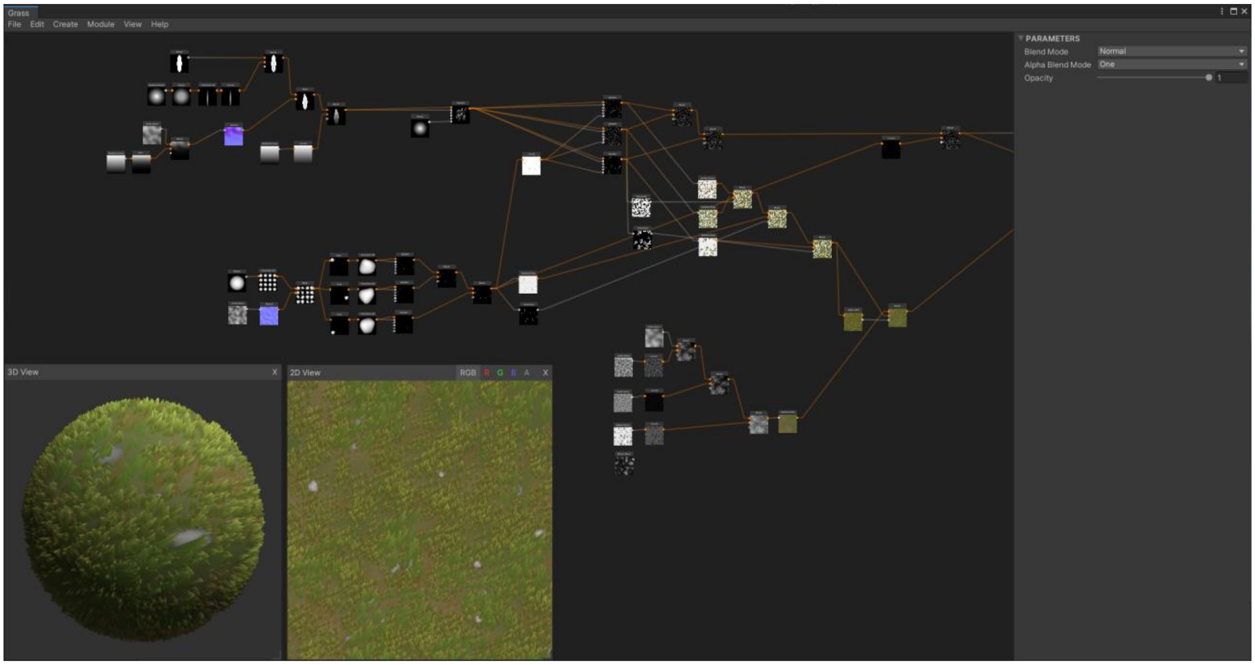
For business or support request, please contact: support@pinwheel.studio

RELEASE LOG

V0.1b

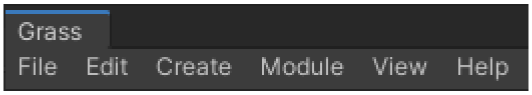
- First release.

USER INTERFACE



PARAMETERS
Blend Mode: Normal
Alpha Blend Mode: One
Opacity: 1

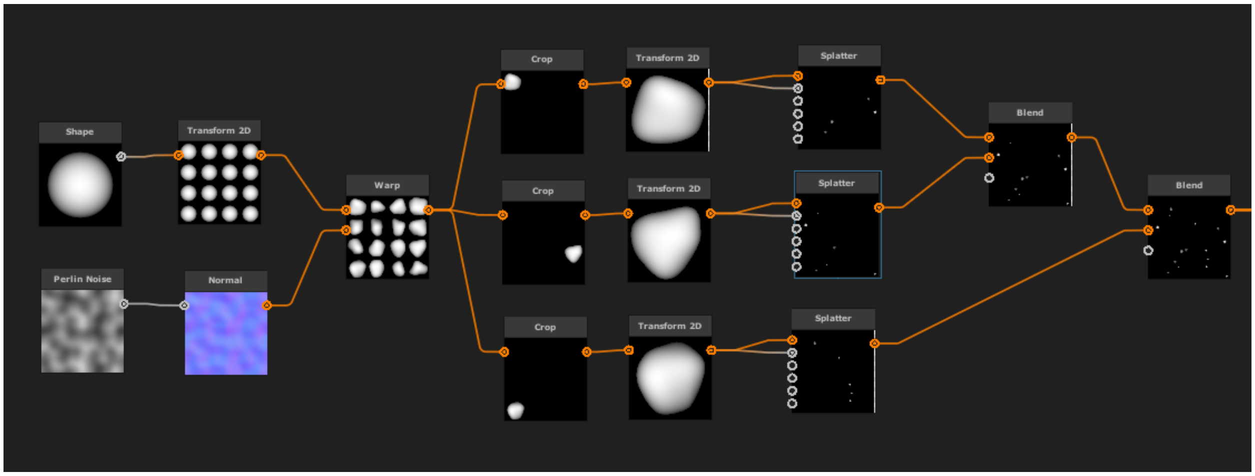
Main Menu



The Main Menu let you do basic operations such as save the graph, opening additional menu such as Export, Graph Settings, etc.

- File: Let you save the graph or export it to result images.
- Edit: Provide Undo/Redo, Cut/Copy/Paste/Duplicate/Delete operation, or open settings for the current graph.
- Create: Creating graph elements such as new node. You can also do this by right click or pressing Spacebar on the graph view.
- Module: You can add more feature to Texture Graph by adding extension modules, this menu will help you to do that.
- View: Open or close additional sub-window such as 3D View, 2D View, etc.
- Help: Accessing additional resources such as documentation, community channels, release log or version info.

Graph View



This is the main part of the editor, where you define the logic to create the texture.

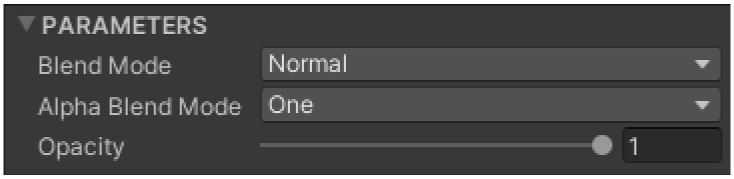
In the graph view, you can create a new node (or other elements) by pressing Spacebar or right click on a blank space and select Create Node.

To connect 2 nodes, simply click and drag the left mouse from one a slot of the first node to the other slot of the second node.

To delete an element, use the left mouse to select it and press Delete.

You can use the right mouse button to display the context menu. Content of the context menu is depend on which elements are selected, such as Cut/Copy/Paste or open Documentation for a specific node.

Detail Panel



▼ PARAMETERS

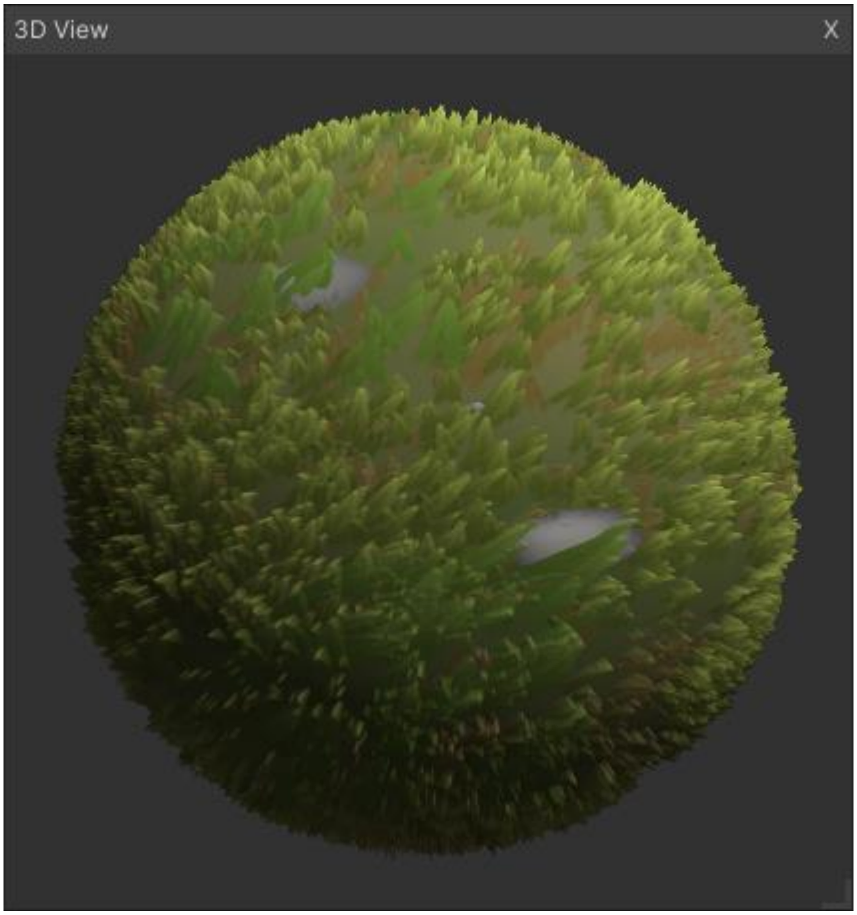
Blend Mode: Normal

Alpha Blend Mode: One

Opacity: 1

The Detail Panel is located on the right side of the editor. This panel will display specific information about the selected element, such as parameters for a node.

3D View



This window displays graph output on a primitive mesh and basic lighting.

To change preview mesh type, right click on the image.

To use a custom mesh, right click on the image, set Mesh to Custom, then drag and drop the desired mesh into the view.

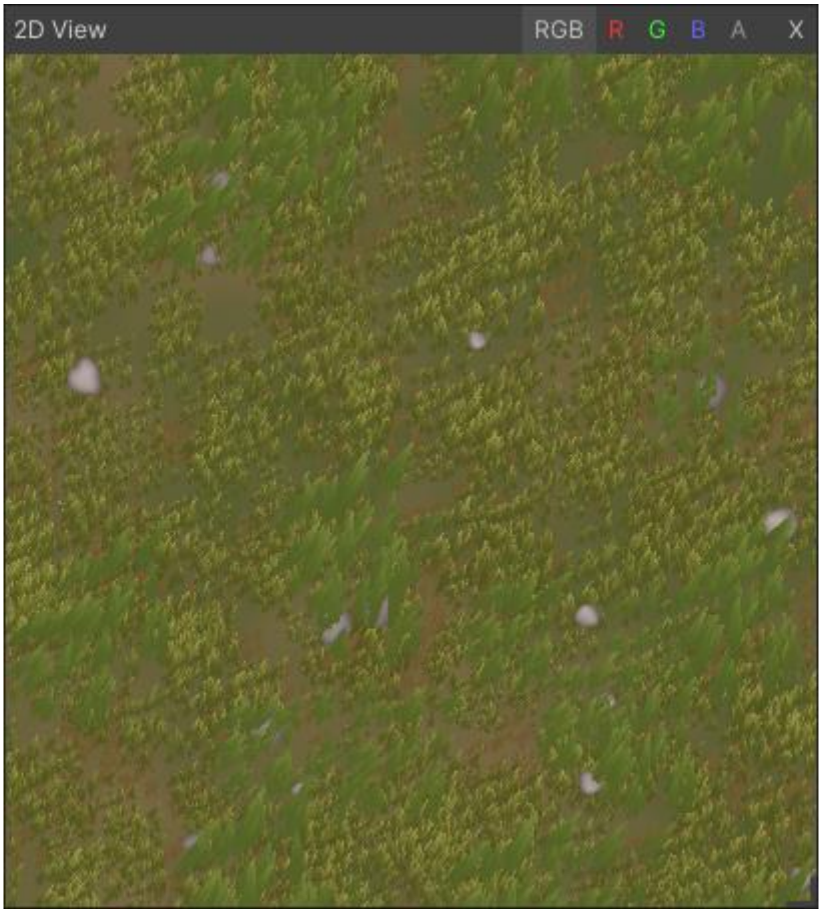
To zoom in/out, use the Scroll Wheel.

To change camera angle, use the Left Mouse.

To change light angle, use Shift + Left Mouse.

To change environment setting such as light color and intensity, tessellation, etc. go to the Graph Settings window (Edit>Graph Settings)

2D View



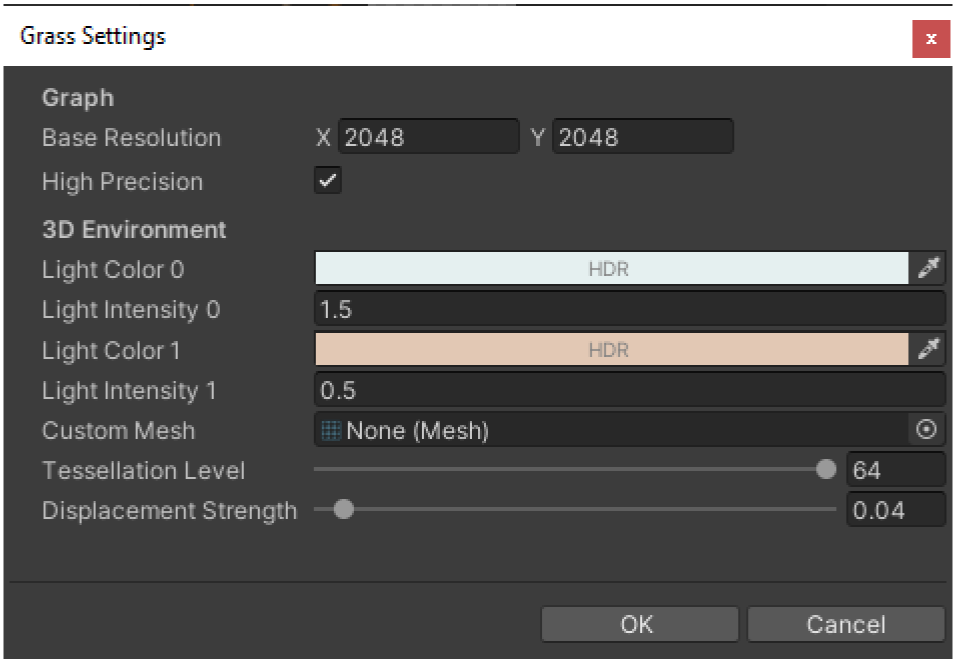
This window displays output image of a node, with custom tiling and offset.

To display a node output, double click on the node.

To change image tilling/offset, use the Scroll Wheel or Left Mouse.

You can view a specific channel of an image by using the RGB/R/G/B/A button on top of the window.

Graph Settings



Graph

Base Resolution X 2048 Y 2048

High Precision ☑

3D Environment

Light Color 0 HDR

Light Intensity 0 1.5

Light Color 1 HDR

Light Intensity 1 0.5

Custom Mesh None (Mesh)

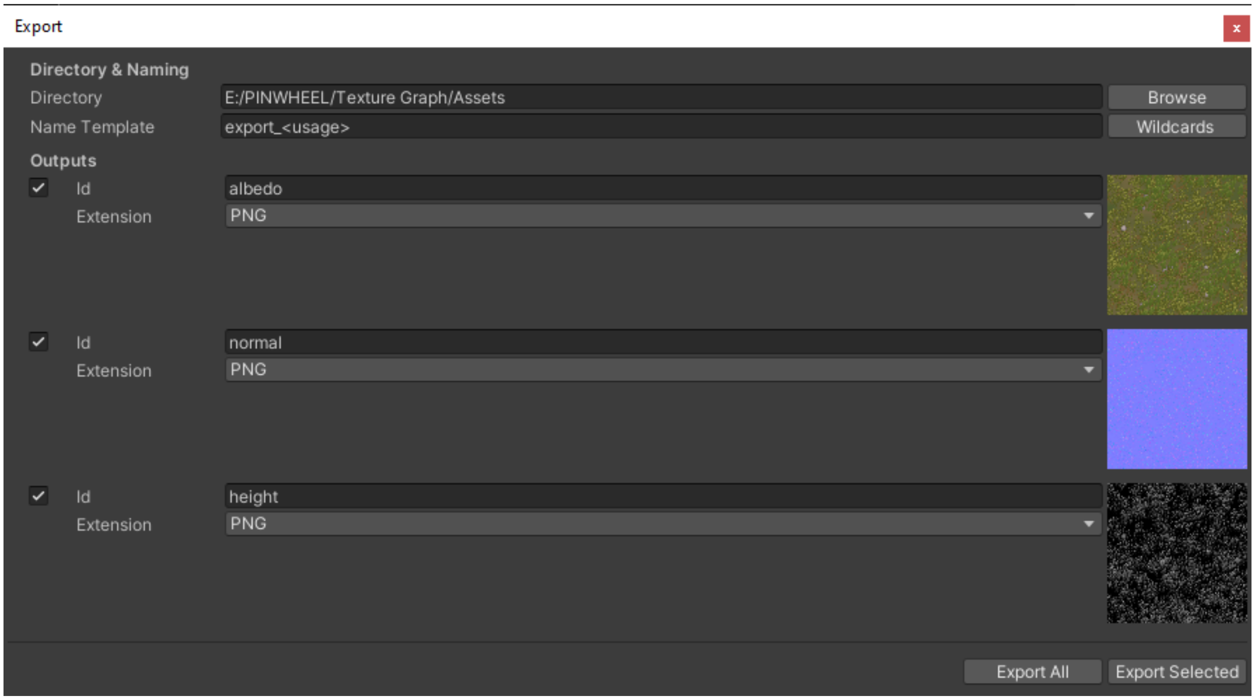
Tessellation Level 64

Displacement Strength 0.04

This window let you edit graph settings and environment parameters for the 3D View.

To open Graph Settings window, go to Edit>Graph Settings.

Export



Export

Directory & Naming

Directory: E:/PINWHEEL/Texture Graph/Assets

Name Template: export_<usage>

Outputs

☑ Id: albedo
Extension: PNG

☑ Id: normal
Extension: PNG

☑ Id: height
Extension: PNG

This window let you export graph outputs to image files.

To open Export window, go to File>Export.

WORKING WITH TEXTURE GRAPH

For most of the time, you creating nodes and connecting them in the graph view to define the graph logic. By using a nodal model instead of stacking of operations, you have a flexible non-linear and non-destructive workflow.

Below are the most often used task you should know when working with Texture Graph.

Creating a new graph asset

Right click on the Project window, then select Create>Texture Graph>New Graph

A new asset will be created in your project, give it a name then press Enter.

Double click on the asset to open the graph editor.

Creating and connecting nodes

There are many way you can use to create a new node:

- In the Main Menu, go to Create>Nodes>...
- Right click on the blank space of the graph view, select Create Node.
- Press Spacebar while hovering your mouse over the graph view.
- Select an existing node and duplicate it by using Copy/Paste/Duplicate menu.

For detail of every node type, please see this section.

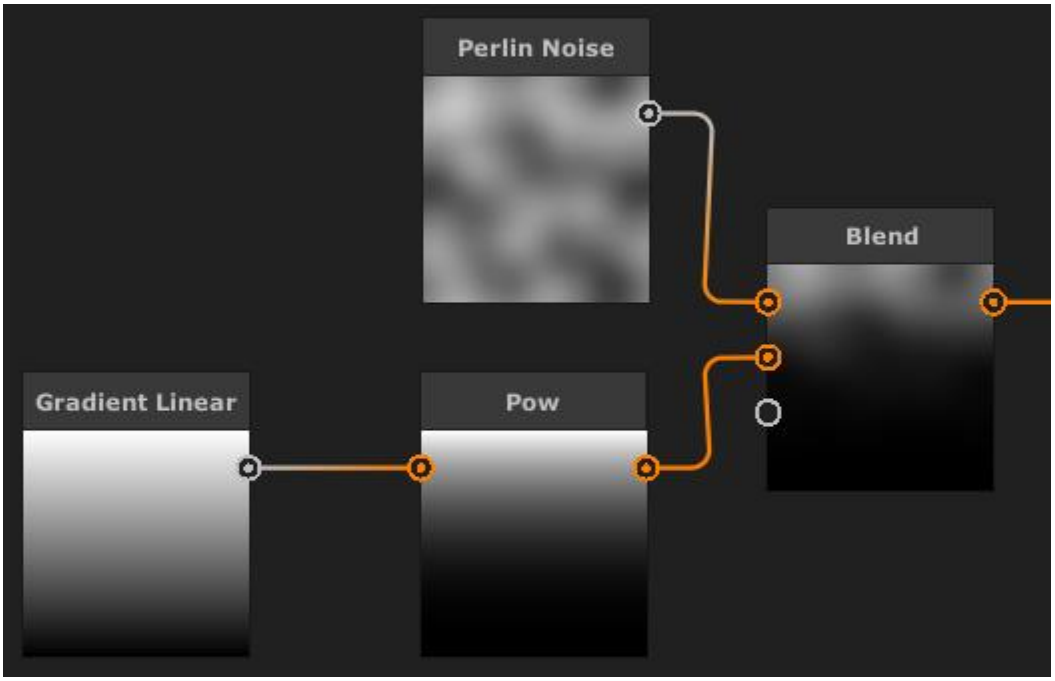
Output of a node can be used as input for other node. To do so, you have to connect the 2 nodes together.

To connect 2 nodes, drag your left mouse from a slot of the first node and release the mouse on another slot of the second node. A line will appear between 2 slots to indicate that they are connected.

To disconnect 2 nodes, select the line between them and press Delete.

Node data flow and data type

Within a node, data flow from left to right, which mean its input is on the left side and output is on the right side.



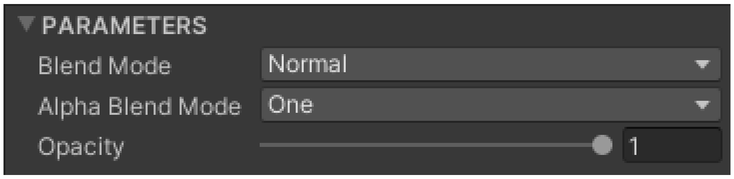
There are 2 type of data: Grayscale which will be displayed as a gray circle, RGBA which will be display as a orange circle.

You can connect 2 slots with different data type, the transferred data will be converted accordingly.

- If a Grayscale output (g) is connected to a RGBA input, it will be converted to RGBA(g,g,g,1)
- If a RGBA output (r,g,b,a) is connected to a Grayscale input, it will be converted to Grayscale ®

Adjusting node parameters

A node may contain several parameters to adjust its output. Select a node in the graph view, then its parameters will be displayed on the Detail Panel on right side of the editor.



PARAMETERS

Blend Mode: Normal

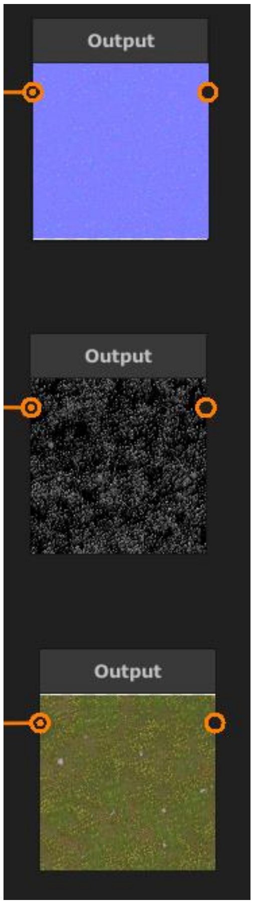
Alpha Blend Mode: One

Opacity: 1

Graph output

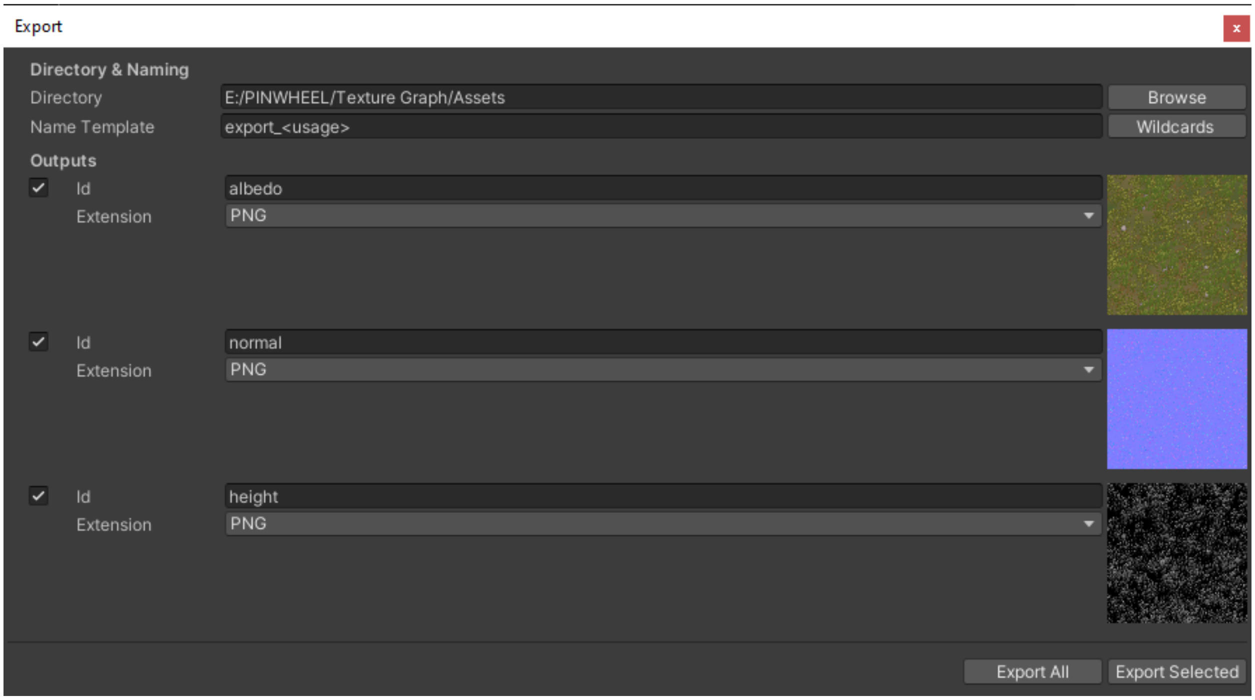
Output is the end points of a graph which determine the textures to be exported to file.

To define an output, you have to create an Output Node in the graph view.



Exporting textures

Go to File>Export to export graph output to textures.



Export

Directory & Naming

Directory: E:/PINWHEEL/Texture Graph/Assets

Name Template: export_<usage>

Outputs

☑ Id: albedo
Extension: PNG

☑ Id: normal
Extension: PNG

☑ Id: height
Extension: PNG

In the Export window, all Output nodes will be displayed along with there result, Id and image extension to export.

There are several parameters to take care of:

- Directory: The folder to save all exported images.
- Name Template: A template for textures' name. You can add wildcards to differentiate them. In the final name, a wildcard will be replaced with appropriate string.
- Id: User-defined identifier for an output. This value should NOT be duplicated between output.
- Extension: Image extension to save the image.

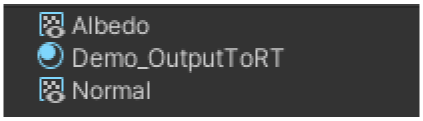
You can export all output at once, or use the check mark in front of each output to export them selectively.

View graph output in the Scene View

Beside of viewing graph output in the 3D View window, you can also do that in the scene view to see how good the result when being used in a scene context, in various lighting and ambient settings, by using the Output To RT node.

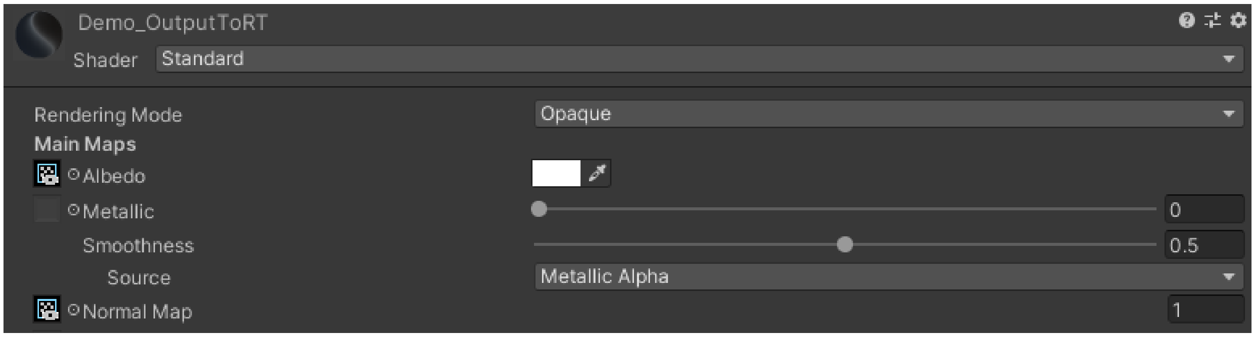
First, create some Custom Render Texture assets by right clicking on the Project window, then select Create>Custom Render Texture. Select them and set an appropriate resolution in the Inspector.

Then, create a new material for your game object, Standard material for example.



Albedo
Demo_OutputToRT
Normal

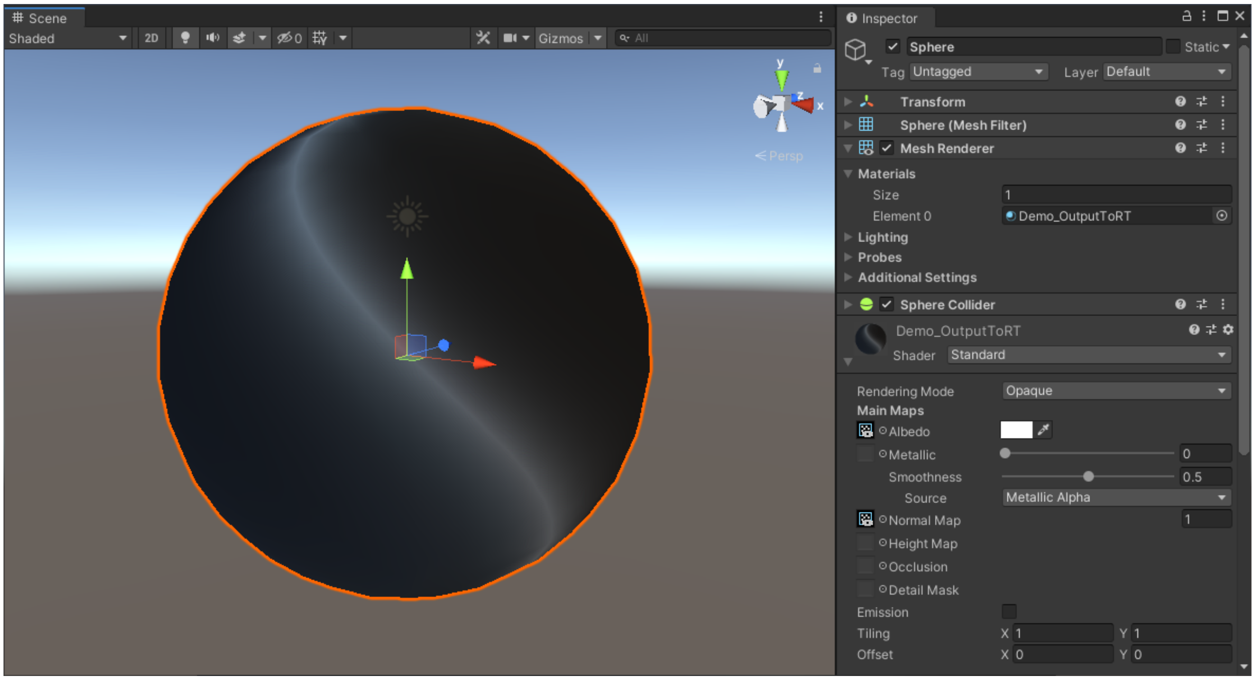
Assign the Custom Render Texture to appropriate slot in the material.



Demo_OutputToRT
Shader: Standard

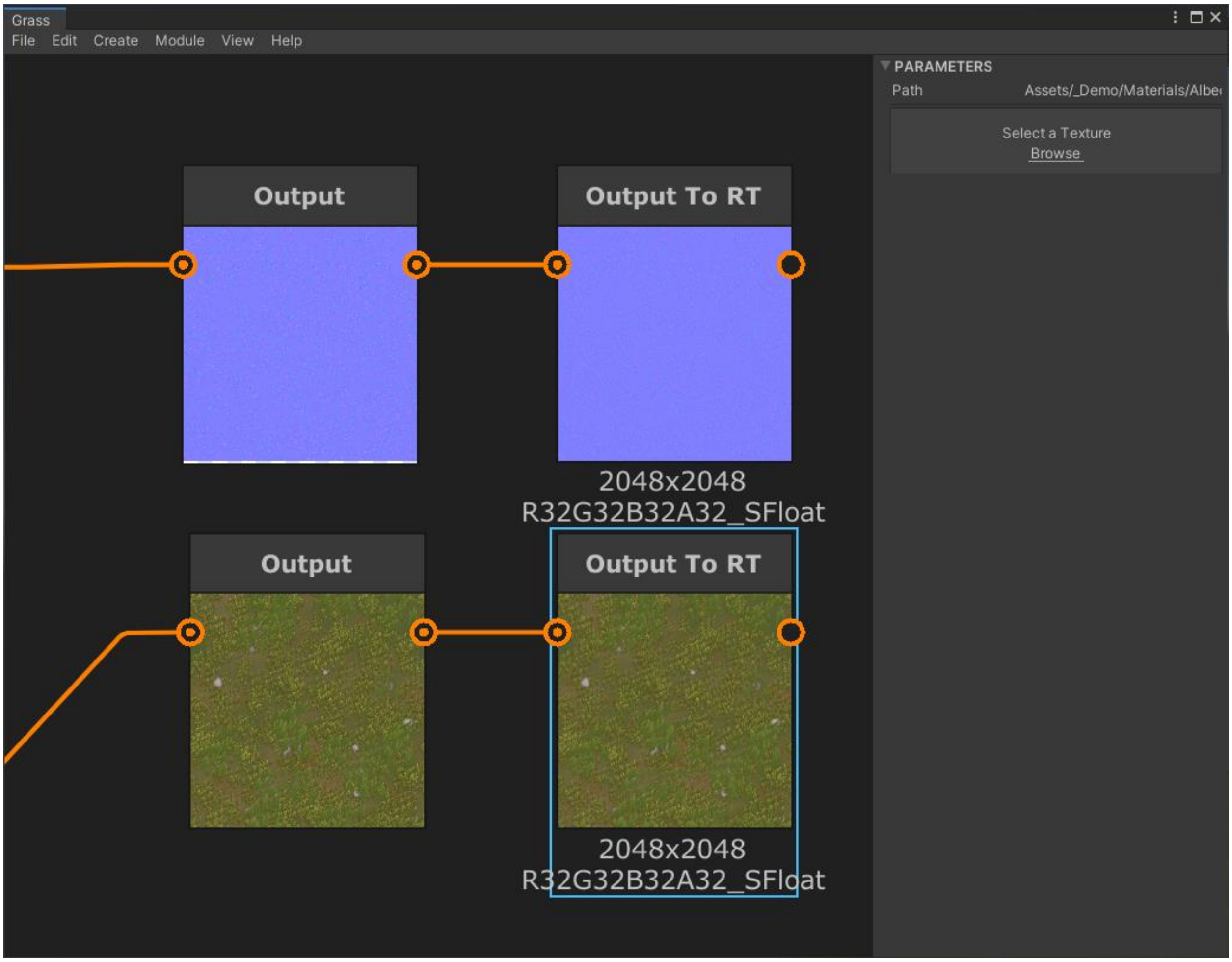
Rendering Mode: Opaque
Main Maps
Albedo
Metallic: 0
Smoothness: 0.5
Source: Metallic Alpha
Normal Map: 1

Assign the material to the desired game object.



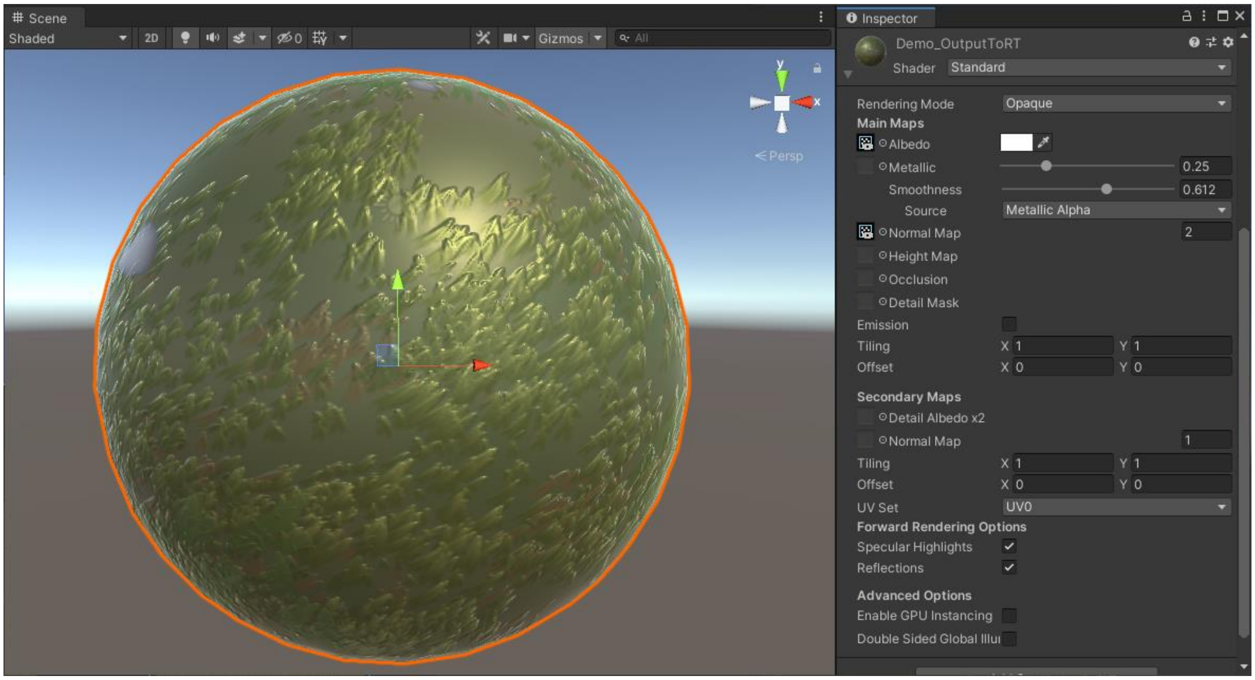
Inspector
Sphere — Static
Tag: Untagged — Layer: Default
Transform
Sphere (Mesh Filter)
Mesh Renderer
Materials
Size: 1
Element 0: Demo_OutputToRT
Lighting
Probes
Additional Settings
Sphere Collider
Demo_OutputToRT
Shader: Standard
Rendering Mode: Opaque
Main Maps
Albedo
Metallic: 0
Smoothness: 0.5
Source: Metallic Alpha
Normal Map: 1
Height Map
Occlusion
Detail Mask
Emission
Tiling: X 1 Y 1
Offset: X 0 Y 0

Open the graph, create some Output To RT node, select the target texture in the Detail Panel, and connect its input slot.



Note that you can connect the Output To RT input slot to any node, not just the Output one.

Back to the Scene View, you can see the texture rendered there.



Demo_OutputToRT

Shader Standard

Rendering Mode Opaque

Main Maps

Albedo

Metallic 0.25

Smoothness 0.612

Source Metallic Alpha

Normal Map 2

Height Map

Occlusion

Detail Mask

Emission □

Tiling X 1 Y 1

Offset X 0 Y 0

Secondary Maps

Detail Albedo x2

Normal Map 1

Tiling X 1 Y 1

Offset X 0 Y 0

UV Set UV0

Forward Rendering Options

Specular Highlights ☑

Reflections ☑

Advanced Options

Enable GPU Instancing □

Double Sided Global Illu □

NODE LIBRARY

Core Module

Alpha Merge

Description

Add alpha to a RGB texture. Useful for masking or alter the alpha channel of an image.

Input

- RGB: Color input.
- A: Input for the A channel.

Output

- RGBA: The packed texture.

Alpha Split

Description

Split a RGBA texture to RGB and A.

Input

- RGBA: The input texture

Output

- RGB: Output for the RGB channel.
- A: Output for the A channel.

Billow Noise

Description

Generate a seamless billow noise (A variant of the traditional Perlin noise).

Output

- Noise Map: The generated noise texture.

Parameters

- Scale: Scale/Density of the noise map.
- Seed: An integer to randomize output.
- Variant: This value is used to creating a slightly different output by rotating the noise random vectors.

Blend

Description

Blend two textures together using a blend operation.

Input

- Background: The background texture. Default is black.
- Foreground: The foreground texture. Default is black.
- Mask: A grayscale map to interpolate between Background and blend result. Default is white.

Output

- Output: The blend result.

Parameters

- Blend Mode: Blend operations for RGB channel. For detail of how each mode work, please refer to this page.
- Alpha Blend Mode: Blend operation for A channel.
- Opacity: Define the strength of blend operations. Opacity will be multiply with Mask value, then used as a factor to interpolate between Background and blend result.

Blend Src Dst

Description

Blend two textures together in a Source Factor and Dest Factor fashion. This is exactly the same way how object are rendered onto the screen in Unity render engine.

Input

- Background: The background texture. Default is black.
- Foreground: The foreground texture. Default is black.

Output

- Output: The blend result.

Parameters

- Src Color: Blend factor for foreground color.
- Dest Color: Blend factor for background color.
- Src Alpha: Blend factor for foreground alpha.
- Dest Alpha: Blend factor for background alpha.
- Color Ops: Blend operation for color.
- Alpha Ops: Blend operation for alpha.

For more information on how these factors work, please refer to this page.

Blur

Description

Perform a simple box blur on an image. A higher quality variance is Gaussian Blur, which trade off performance for quality.

Input

- Input: The image to be blurred.

Output

- Output: Blur result.

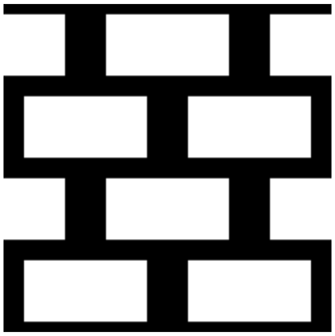
Parameters

- Radius: Radius of the blur. Higher value gives more blurry result, but cost more time to compute.

Brick

Description

Generate a simple brick pattern.



Output

- Output: The output image.

Parameters

- Tiling: Tile the pattern multiple times.
- Gap Size (0-1): Size of the gap between blocks.
- Inner Size (0-1): Size of the inner part of each block, to create the feel of extrusion.

Cartesian To Polar

Description

Convert an image from Cartesian (x,y) to Polar (radius, angle) coordinate.

Input

- Input: The input texture.

Output

- Output: The output texture

Channels Shuffle

Description

Combine the RGBA channel of 2 input textures in any order to create a new texture.

Input

- Input 0: The first input image.
- Input 1: The second input image.

Output

- Output: Combine result.

Parameters

- Red/Green/Blue/Alpha Source: Select the data channel to put into the result.

Checkerboard

Description

Generate a seamless checkerboard pattern.

Output

- Output: The generated pattern.

Parameters

- Scale: Scale/Density of the pattern.
- Color 0: The first color (or odd color) of the pattern.
- Color 1: The second color (or even color) of the pattern.

Clamp

Description

Clamp the image data within a Min and Max value.

Input

- Input: The input image.

Output

- Output: The output image.

Parameters

- Min/Max: Value range.
- Apply To Alpha: Determine if it should apply the effect to alpha channel.

Colorspace Conversion

Description

Convert and color input from Gamma (sRGB) to Linear and vice versa.

Input

- Input: The input image.

Output

- Output: The output image.

Parameters

- Mode: The conversion mode, Gamma To Linear or Linear To Gamma.

Contrast & Brightness

Description

Adjust contrast & brightness level of an input image

Input

- Input: The input image.

Output

- Output: The output image.

Parameters

- Contrast: Adjust the distinction of the image color.
- Brightness: Adjust the image light level.

Crop

Description

This node let you select an area on an image with a rectangle and result with the unselected region discarded.

For ease of editing, you can interact with the canvas on the parameter panel.

Input

- Input: The input image.

Output

- Output: The output image.

Parameters

- Rect Center: Center position of the cropping rectangle, in percentage.
- Rect Rotation: Rotation of the cropping rectangle, in degree.
- Rect Size: Size of the cropping rectangle, in percentage.
- Background Color: Color to fill in the discarded pixels.

Curve

Description

Use curves to remap texture data.

Input

- Input: The input image.

Output

- Output: The output image.

Parameters

- RGB: The curve to apply to RGB channel. This curve will be applied **after** per-channel curves.
- R: The curve to apply to R channel.
- G: The curve to apply to G channel.
- B: The curve to apply to B channel.
- A: The curve to apply to A channel.

Emboss

Description

Perform a simple 2D lighting effect on an input image with a heightmap. Useful when you want to create a 3D-like look for the texture or simple light baking.

Input

- Main Texture: The input image.

- Height Map: A grayscale map to extract the shape direction/normal vector.

Output

- Output: The output image.

Parameters

- Light Angle: Angle of the light source, in degree.
- Intensity: Intensity of the effect.
- Highlight Color: Tint color for the area that face the light source.
- Shadow Color: Tint color for the area that away from the light source.

Gaussian Blur

Description

Perform a high quality Gaussian blur on an image. A simpler variance is Blur, which trade off quality for performance.

Input

- Input: The image to be blurred.

Output

- Output: Blur result.

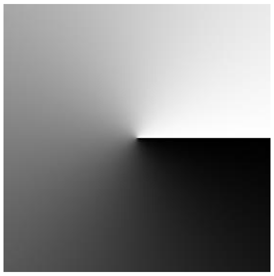
Parameters

- Radius: Radius of the blur. Higher value gives more blurry result, but cost more time to compute.

Gradient Angular

Description

Generate a grayscale angular gradient defined by a center point and an endpoint.



Output

- Output: The output image.

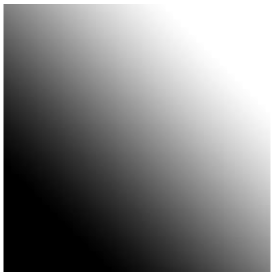
Parameters

- Center Point: The point where the gradient begins(full white).
- End Point: The point where the gradient ends (full black).

Gradient Axial

Description

Generate a grayscale gradient that follow a line defined by 2 points. The gradient can be made to be reflected at the center of the line.



Output

- Output: The output image.

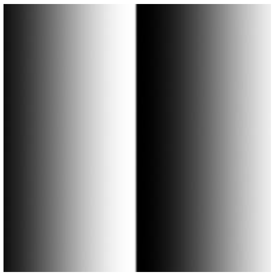
Parameters

- Start Point: The point where the gradient begins(full black).
- End Point: The point where the gradient ends (full white).

Gradient Linear

Description

Generate a linear gradient. The gradient can be repeated or rotated on a predefined set of angles.



Output

- Output: The output image.

Parameters

- Rotation: Rotation of the gradient, in degree (0, 90, 180, 270)
- Scale: How many times the gradient should be repeated.
- Mid Point: The point on each strip where the gradient getting reflected.

Gradient Map

Description

Use a gradient to remap a grayscale image to a colored one.

Input

- Input: The input image.

Output

- Output: The output image.

Parameters

- Gradient: The gradient to apply.
- Scale: How many times to repeat the gradient.
- Wrap Mode: How to repeat the gradient.

Gradient Map Dynamic

Description

Remap a grayscale image to a colored one using a gradient texture. Similar to Gradient Map node but the gradient comes from another node.

Input

- Main Texture: The input grayscale image.
- Gradient: The color texture which will be sampled as a gradient

Output

- Output: The output image.

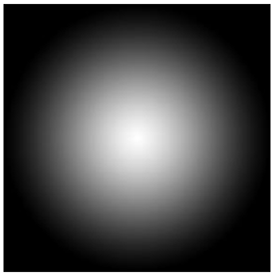
Parameters

- Axis: The axis to sample the gradient texture.
- Slice: Position of the gradient line/column in UV space.

Gradient Radial

Description

Generate a grayscale radial gradient defined by a center point where it is full white, and an end point where it becomes full black.



Output

- Output: The output image.

Parameters

- Center Point: The point where the gradient begins(full white).
- End Point: The point where the gradient ends (full black).

Grayscale

Description

Convert a color texture to grayscale.

Input

- Color: The color input image.

Output

- Gray: The grayscale output image.

HSL

Description

Allow you to adjust Hue, Saturation and Lightness of an input image.

Input

- Input: The input image.

Output

- Output: The output image.

Parameters

- Hue: Shift the image color.
- Saturation: Adjust color intensity.
- Lightness: Adjust color lightness.

Invert

Description

Invert color of a texture (1-color).

Input

- Input: The color input image.

Output

- Output: The output image.

Levels

Description

This node allow you to correct the tone of an image by adjusting its shadow, midtones and highlight. An editor is also provided to make it easier to work with this node, in the parameters panel.

This node is similar to Photoshop Levels Adjustment, please take a look at this page for more detail information.

Input

- Input: The color input image.

Output

- Output: The output image.

Load Texture

Description

Load a texture from asset. The texture only being linked with the graph, not embedded.

Output

- Texture: The linked image.

Parameters

- Path: The path to the texture. This can be under Assets/ folder or Resources/ folder. To load a texture at runtime, provide a Resources/ base path.

Mirror

Description

Mirror the input image over an axis, a quick way to get symmetrical effect.

Input

- Input: The input image.

Output

- Output: The output image.

Parameters

- Mirror X/Y: Should it mirror the input over X/Y axis?
- Flip X/Y: Choose a side to mirror from.
- Offset X/Y: Define the mirror X/Y axis.

Normal

Description

Generate a tangent space normal map based on an input height map.

Input

- Input: The input height map.

Output

- Output: The output normal map.

Parameters

- Strength: Normal map strength, this value will get multiplied with height map input for normal vector calculation.

Output

Description

This node does no texture modification. It serve the purpose of the end point of your graph, which has some settings to determine how texture are exported from the graph.

Input

- Texture: The texture to output.

Parameters

- Output Id: A user-defined ID which will be used to identify several output of a single graph. This value should not be empty or duplicated.

Output To RT

Description

Render the graph onto a Custom Render Texture asset.

Input

- Texture: The texture to render out.

Parameters

- Path: The path to the Custom Render Texture asset. This can be under Assets/ folder or Resources/ folder. For runtime usage, provide a Resources/ base path.

Perlin Noise

Description

Generate a seamless Perlin noise.

Output

- Noise Map: The generated noise texture.

Parameters

- Scale: Scale/Density of the noise map.
- Seed: An integer to randomize output.
- Variant: This value is used to creating a slightly different output by rotating the noise random vectors.

Polar To Cartesian

Description

Convert an image from Polar (radius, angle) to Cartesian (x,y) coordinate.

Input

- Input: The input texture.

Output

- Output: The output texture

Pow

Description

Apply a power function over texture data.

Input

- Input: The input texture.

Output

- Output: The output texture.

Parameters

- Power: Exponent value.

Quad Transform

Description

Transform an input image using 4 anchor points. You can interact with the canvas on the Parameters panel for ease of editing.

Input

- Input: The input texture.

Output

- Output: The output texture.

Parameters

- Point 0/1/2/3: Position of the 4 anchor points in UV space.
- Background Color: Color of the pixels that are not covered by the quad.
- Cull Mode: The culling mode.
- Flip Order: Flip the render order of the 2 triangles.

RGBA Merge

Description

Pack 4 grayscale inputs into a single RGBA texture. Useful for texture packing.

Input

- R: Input for the R channel.
- G: Input for the G channel.
- B: Input for the B channel.
- A: Input for the A channel.

Output

- RGBA: The packed texture

RGBA Split

Description

Split a RGBA texture to 4 grayscale output respectively to its channels.

Input

- RGBA: The input texture

Output

- R: Output for the R channel.
- G: Output for the G channel.
- B: Output for the B channel.
- A: Output for the A channel.

Ridged Noise

Description

Generate a seamless ridged noise (A variant of traditional Perlin noise).

Output

- Noise Map: The generated noise texture.

Parameters

- Scale: Scale/Density of the noise map.

- Seed: An integer to randomize output.
- Variant: This value is used to creating a slightly different output by rotating the noise random vectors.

Shape

Description

Generate grayscale primitive shapes, including: Square, Disc, Hemisphere, Cone, Paraboloid, Bell, Thorn, Pyramid, Brick, Torus

Output

- Output: The generated texture.

Parameters

- Shape: The shape to be generated.
- Scale: Scale of the shape.
- Inner Size: Size of the inner part of some shape (Brick, Torus)

Sharpen

Description

Apply a sharpen effect on the input image.

Input

- Input: The input image.

Output

- Output: The output image.

Parameters

- Intensity: Strength of the effect.

Skew

Description

Skew an input image.

Input

- Input: The input image.

Output

- Output: The output image.

Parameters

- Axis: The skew axis.
- Amount: The skew amount [-1, 1].

Solid Color

Description

Output a solid color.

Output

- Output Color: The output color.

Parameters

- Color: The color to output.

Splatter

Description

This node allows you to to create a randomly distributed patterns using a source shape texture and make sure the result is seamless.

Input

- Shape: The shape to scatter.
- Shape Alpha: Additional alpha multiplier to mask out the shape.
- Offset Map: A grayscale map to control the random position. Black is no random offset.
- Rotation Map: A grayscale map to control the random rotation. Black is no random rotation.
- Scale Map: A grayscale map to control the random scale. Black is no random scale.
- Mask Map: A grayscale map to strip off instances in particular regions.

Output

- Output: The output texture.

Parameters

- Tile X/Y: Number of tiles on X/Y axis.
- Instance Per Tile: Number of instance to draw at each tile.
- Base Offset: Offset the whole texture. This value is independent from Offset Map, Offset Min/Max and Offset Multiplier.
- Offset Min/Max X/Y: Minimum and maximum random offset amount.
- Offset Multiplier: Control the random offset intensity.
- Offset Random Seed: A number to randomize the offset direction.

- Base Rotation: Rotate every tiles uniformly, regardless of Rotation Map, Rotation Min/Max and Rotation Multiplier.
- Rotation Min/Max: Minimum and maximum random rotation angle.
- Rotation Multiplier: Control the random rotation intensity.
- Rotation Random Seed: A number to randomize the rotation angle.
- Base Scale: Scale each tile uniformly, regardless of Scale Map, Scale Min/Max and Scale Multiplier.
- Scale Min/Max X/Y: Minimum and maximum random scale amount.
- Scale Multiplier: Control the random scale intensity.
- Scale Random Seed: A number to randomize the scale.
- Background Color: Color of the background.
- Hue/Saturation/Lightness Variation: Create a different color for each tile.
- HSL Random Seed: Randomize the color variation.
- Mask Threshold: Apply threshold filter to the mask map without the need of using an additional node, to create sharper mask.
- Mask Random Seed: Randomize the masking.

Step

Description

Apply a quantize/step function over texture data. This node is mainly to create a banding effect on an image.

Input

- Input: The input texture.

Output

- Output: The output texture.

Parameters

- Step: How many step to adjust texture data. (0-256)

Threshold

Description

Turn a pixel to full white if its value is greater (or equal) a threshold value.

Input

- Input: The input texture.

Output

- Output: The output texture.

Parameters

- Threshold: The threshold value.
- Mode: Comparison mode.

Transform 2D

Description

Translate, rotate and scale an input image. The result can be looped over image boundaries.

Input

- Input: The input texture.

Output

- Output: The output texture.

Parameters

- Tiling Mode: How to tile the image if it cross the boundaries.
- Offset: Offset the input image, in percentage.
- Rotation: Rotate the input image, in degree.
- Scale: Scale the input image, in percentage.
- Background Color: A color to replace the pixel if it is out of boundaries, only when Tiling is not enabled for an axis.

Value Noise

Description

Generate a seamless value noise.

Output

- Noise Map: The generated noise texture.

Parameters

- Scale: Scale/Density of the noise map.
- Seed: An integer to randomize output.

Warp

Description

Distort an input image based on a vector map. Only X and Y channel of the vector are in used. You can use the Normal node to generate a normal/vector map based on a grayscale image.

Input

- Main Texture: The image to be distorted.
- Vector Map: The vector map which is used to distort the main image.

Output

- Output: The distorted texture.

Parameters

- Intensity: Strength of the effect.