PxrProjectionLayer

A specialized version of PxrTexture to project textures, designed to work with PxrProjectionStack.

It has additional functionalities to optimize texturing and apply multiple textures at once.

Input Parameters

Filename

The filename of the texture. This parameter must be supplied.

> This parameters shouldn't be left empty, otherwise the missing color will be projected instead.

First Channel Offset

First channel offset to be looked up for filename.

Filter

Selects different reconstruction filters that can be used during texture lookup. The filters available for PxrProjectionLayer are:

Value	Filter
0	Nearest
1	Box
2	Bilinear
3	Bspline
6	Gaussian
7	Lagrangian

Blur

Specifies how much to blur the image retrieved from the texture file.

Mip Interpolate

Selects whether to interpolate between adjacent resolutions in the multi-resolution texture, resulting in smoother transitions between levels.

Missing Color

If there is an error opening the texture, use this color.

Missing Alpha

If there is an error opening the texture, use this alpha.

Linearize

Apply the reverse sRGB transform your texture. If you are painting textures in sRGB space (default for most paint packages) but viewing your data in data linear space, your textures will look washed out. This will apply the sRGB transform to your texture, which should make it appear visually linear again.

PreMultiply

Enable this if you want the final color to be multiplied by the alpha.

Mask

A black and white mask can be used to avoid reading texture on points that won't be visible. PxrProjectionLayer is providing such an output (resultMask) to avoid reading texels that are either occluded or outside of the projection window.

Channels Filenames

Channels are additionnal RGB textures that can be projected and output to other nodes in the graph or to AOVs. This is typically used to project additional material properties like specular color, roughness, etc. Channels will be combined and output by PxrProjectionStack. Note that only PxrProjectionStack knows how to process and route channels to outputs. You can project as many channels as you want but bear in mind that a large number of channels will have an impact on memory and render time. Channel textures should always be in the same order if you are combining multiple projections with PxrProjectionStack, otherwise your inputs will be incorrect.

Channels Linearize
Specifies which channel needs to be linearized.

Channels Missing Color

Specifies a missing color for each channel.

Channels Input Black

Define the black point of the channel's color. This is working exactly like Photoshop's Levels : every value below the black point will become black.

Channels Input Gamma

Define the gamma of the channel's color. This is working exactly like Photoshop's Levels.

Channels Input White

Define the white point of the channel's color. This is working exactly like Photoshop's Levels : any color above the white point will become white.

Channels Output Black

Define the black output level. This is working exactly like Photoshop's Levels : this will define the darkest output color.

Channels Output White

Define the white output level. This is working exactly like Photoshop's Levels : this will define the lightest output color.

Manifold

Provides the range over which to apply the texture. Plug a PxrProjector here to provide mapping coordinates to the texture function. Defaults to s,t.

MipBias

Bias mip selection chosen for rendering to either a higher (negative integers) or lower resolution (positive integers) selection.

MaxResolution

Clamp mip selection to a preferred maximum level. This may prevent the highest resolutions from being rendered depending on your selection.

OptimizeIndirect

Sets texture filter to Box (inexpensive) on indirect hits instead of the selected filter used for other visible effects.

Output Parameters

resultRGB
The filtered color result. Note all results are looked up starting at the startChannel offset.

resultR

The R channel result

resultG

The G channel result

resultB

The B channel result

resultA

The Alpha channel, which takes into account the mask provided through the mask input.

outNumChannels

The number of channels output by this layer. This is the number of textures defined in channelsFilenames. This should always be connected to PxrProjectionStack's layersNumChannels parameter so it may unpack the channel data.

outChannelsRGB

The RGB channel data, packed in a single output. PxrProjectionStack needs to know the number of available channels through a connection with outNumChannels to be able to unpack the data.