

Layered Projection Shader for Mental Ray

website: www.happiestdays.com

Overview

In its simplest form the layered projection shader takes N texture projections and decides which is the best to use for each given shading sample. The shader takes into account texture projection shadowing (if a sample is hidden from a particular projection), texture projection angle and texture projection distance when making this decision.

An interesting point to note (for the new user) is that this shader does not do the actual texture sampling itself. That process is currently left up to the projection utility node. Later versions of the layered projection shader may need to project the actual texture as well to allow for more complex blending.

Version History

Ver 7

- Added support for spherical projections
- Added edge feathering for "Use Best" mode
- Added nearClip attribute
- Renamed to layeredProjectionShader to avoid ambiguity with the MAYA projection node

Ver 6

- Added farClip attribute
- Overhaul of the MEL helper functions (fixed numerous bugs)

Ver 5

- Added ability to choose between general camera direction and "absolute" camera (or ray) direction testing

Ver 4

- Added missing functions in addProjection MEL script (it actually works now)
- Added view dependent projection selection mode
- Added ability to set the maximum number of projections used in blend mode
- Minor performance improvement, global initialisations moved from sample level loop
- Minor performance improvement, occlusion falloff testing

Matte Painting Projections

The layered projection shader was initially developed for a camera projection of a city matte painting, with a second camera projection used to fill in big gaps once the 3D camera move was defined (hint, use a contrasting shadow colour when test rendering to find your gaps easily). It has since been refined for use on projects with multiple matte painting projections as well as generic layered textured projection.

A quick and accurate way of calibrating camera positions from photos is to use image-modelling software like [REALVIZ ImageModeler](#) or [EOS PhotoModeler](#). Although both softwares also allow the building of geometry and extraction of textures, I have found the workflow to be smoother when modelling and texturing (using this shader) in MAYA.

How To Use

The included MEL scripts will assist in the creation of the projection shader and the connection of projections to it.

createLayeredProjectionShader()

Will create a new layeredProjectionShader connected to a surfaceShader. Any selected projections will be added to the new layeredProjectionShader.

addProjection()

Will add the selected projections to a layeredProjectionShader, note that the layeredProjectionShader must be selected first.

Quick Start

1. Create two cameras for projection, lets call them ProjCamA and ProjCamB
2. Create a checkboard texture with "As projection"
3. Create a cloth texture with "As projection"
4. Link the checkboard projection to ProjCamA
5. Link the cloth projection to ProjCamB
6. Select both projection utilites and execute createLayeredProjectionShader()
7. Create a sphere
8. Orient the cameras to point at the sphere from different angles
9. Apply the newly created shader to the sphere
10. Render!

Parameters

These parameters are all editable in the Attribute Editor.

Selection Mode

The method to use when choosing from multiple projections:

Use First (0) - will use the first unshadowed projection (depends on order of the Projections parameter)

Use Best (1)- will use the unshadowed projection with smallest deviation from the normal (or view direction if **View Dependant** is switched on)

Blend Best (2) - involves the blending of all unshadowed projections, weighted by their deviation from the normal (or view direction if **View Dependant** is switched on) *the usefulness of this mode is questionable*

Feather Edge

Amount of feathering when "Use Best" mode is enabled. A value of 0 denotes no feathering while 1 is full feathering. Feathering is calculated as angular deviation from optimum of the two best projections.

Max Blend

When the **SelectionMode** is set to "Blend Best", only the best **MaxBlend** projections will be blended.

View Dependent

When enabled, the best projection is chosen based on viewer angle not surface angle (surface normal or facing ratio from the projection camera). So the projection camera whose direction is closest to the view camera is chosen to texture a particular sample. Helps for texture perspective problems that arise when a recess (e.g. door recessed in frame) is not modelled in the geometry.

Double Side

Ensures normals are always pointing towards the projection when selecting projections in **Use Best** or **Blend Best** mode.

Use Perspective Direction

Only makes sense for camera/perspective projections. When enabled the camera direction (i.e. Z axis in camera space) is used as the test direction in **Use Best** or **Blend Best** mode. Use this option if you have large planar faces exhibiting noticeable projection swapping.

Shadow Color

Is the color to be used when no valid projection is found for the current sample. Can be set to a contrasting color to find gaps for your matte painters to fill. You can also plug a projection or projectionShader into the shadow color.

Projections

.type – projection type (from Maya projection node)

.xform – projection world transform matrix

.nearClip

.farClip

.color

.matte

This is an array of the actual projections that the projection shader has to choose from. Areas not defined in the **matte** are considered invalid and the shader will attempt to find a different projection for the current sample (just as if the sample was shadowed at this location). The transform is the transformation matrix to go from camera to world space (in the Maya the worldMatrix attribute is sufficient). Each projection will only be valid inside its **nearClip** to **farClip** distance.

Output

Out Color

Resulting color of combined input projections.

Out Shadow Matte

Defines areas on geometry which are completely shadowed from all available projections.

Number Available Textures

For each sample, the number of valid projections which the projection shader had available to it (depends on selection mode, e.g. will always be 0 or 1 for Use First)