Linear Workflow (LW) with Autodesk Maya for Vray

In my experiences as a teacher, technical director and CG pipeline developer, most Maya artists don’t understand linear workflow (LW) and would rather ignore it. That would be fine if Maya followed this proper rendering workflow by default, but it does not. Without this rendering and display pipeline you can't make correct lighting decisions, particularly for photo-realistic rendering. Even edge anti-aliasing doesn't look right on a non-gamma boosted image viewer.

Many CG production facilities lock down their color pipeline so new artists are not tempted work without it. At Industrial Light and Magic we called it "gamma wars". Many of the newly hired artists, including myself, didn't like the "look" of a properly boosted display. Renders looked flat and unsaturated. Eventually the veteran color scientists at ILM convinced the protesting newbies with irrefutable facts, until the next batch of new hires started the "gamma wars" all over again.

Linear Workflow (LW) with 3D applications pertains to color managing the viewer, image textures and constant color values to display renders with the correct contrast. Some people call this contrast, gamma. A linear workflow pipeline includes proper display gamma as well as making sure incoming textures and constant colors are in a linear color space. If you want to know more, my favorite color scientist, Charles Poynton’s Color and Gamma website explains the gamma issue in great detail.

This is a step by step tutorial on how to set up Linear Workflow (LW) in Autodesk Maya when rendering with Vray. The content is specific to versions prior to Maya2016 or Maya2015Extension. Color Management which was added to Autodesk Maya 2011, allowed artists to quickly setup a LW for Mentalray. Prior to Maya2011, setting up LW was far more convoluted and varied widely. Unfortuenately, Vray ignores most the built-in color management in Maya and has its own render settings. There is a slight overlap if you view Vray renders with Renderview rather than Vray's native VFB viewer. You should be aware that the Maya software renderer does not support this color management. From this point reader should be familiar with the following Maya interface panels: Vray VFB, RenderView, RenderSettings and Hypershade.

Included with this tutorial is a zipped Maya2014 project with a typical background image and a Maya scene for rendering float (exr) output. This scene has the appropriate color management settings for interactive display and will produce correct output using interactive and batch rendering.

Figure 1 is a composite using Gimp the free paint package of the foreground render blended with the normal mode over the original background. As you can see the render layer blends seamlessly with the background even though the background is an sRGB image and we are working in a linear workflow (LW).

Figure 3

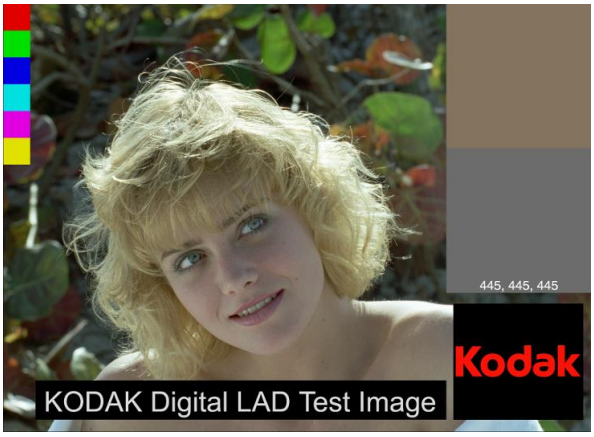

Figure 1

Figure 2

The CG layer (Figure 3) includes six constant color rectangles, two constant color spheres, two constant color rectangles, a gradated ramp rectangle. The matching RGB values can be acquired from the background image (Figure 2) with a paint app like Photoshop or Gimp or the interactive Maya screen color picker. The CG layer in this case is rendered with Mentalray which has a option to un-premult. This un-premult option results in an RGB/Alpha png image that blends with no edge fringing because the RGB channels are actually divided by alpha. Vray, at the time of writing this tutorial, has no such option. This is not required when rendering openexr images since that format assume a pre-multed output. However, fringe free compositing of Vray float openexr renders are only possible with software like Nuke or Aftereffects. Photoshop does

not layer pre-mulitplied openexr images with alpha without using some of its less than ideal fringe removal tricks.

The next two close-up crop images show the difference in the falloff of CG lighting with LW and without LW. It's subtle in this case but still evident. I can tell Figure 4 is the LW render because the light edge is more abrupt matching how light actually reacts to a rounded surface in the real world. Figure 5 is the non LW render. You can see why some artists might prefer the non LW "look" but this is not photo realistic.

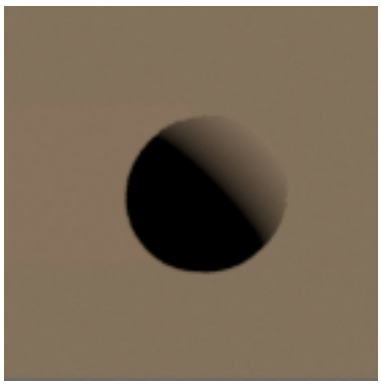

Figure 4

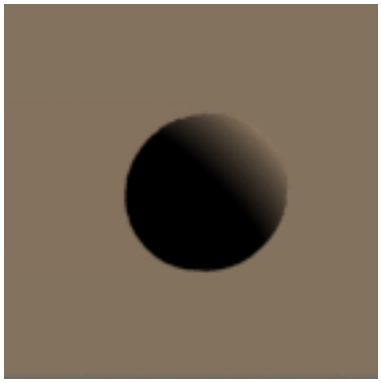

Figure 5

Linear Workflow Render Settings for Vray

When you first start Maya for Vray the defaults are close. I recommend changing the mode to "Dont affect colors, only adaptation"(figure 10a). This is a good option since it doesn't effect the actual final pixel colors but uses gamma boosted values for the adaptive sampling tests. The Color mapping advanced "linear workflow" feature is being phased out and is not recommended by Chaos. It's only effect is to invert the input of the Diffuse Color of Vray standard materials. No other color attributes in the Vray standard materials are affected. Also, none of Maya's built-in material color attributes are affected by this toggle. Specifics for the input image File node are described later in this tutorial

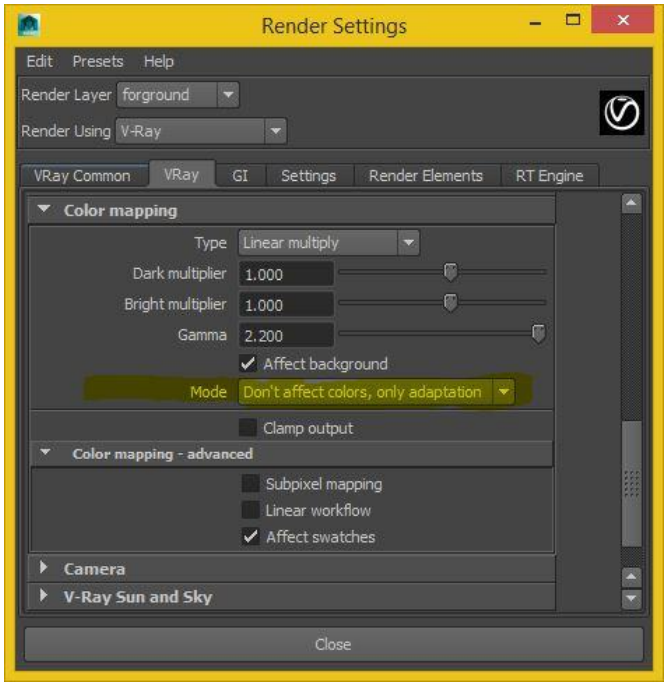

Figure 10a

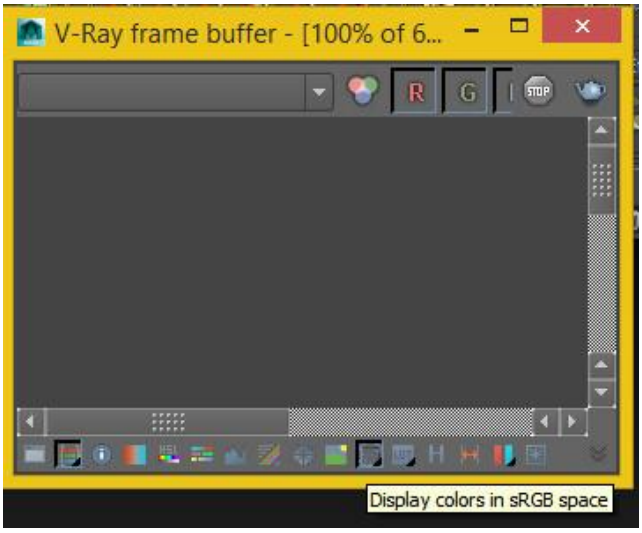

Figure 10b

Vray has its own interactive render viewer called VFB which defaults to a gamma boosted sRGB color profile, this is good(Figure 10b). It is possible to use Maya's Renderview for interactive rendering but setting it up is a bit involved so I recommend using the VFB with Vray.

Float Output with Vray

I do not recommend rendering 8bit images with Vray. At the time of this writing, Vray does not have an option to render 8bit images with a gamma boosted sRGB color profile. You could just apply a gamma 2.2 but this is not the same as sRGB. If you render 8bit images with a un-boosted linear color profile, they will be too dark. If you apply the sRGB profile afterwards, it is very likely that the image will result in gross banding. Also since Vray has no option to un-premult this will make "fringe free" compositing of 8bit render layers difficult.

Now let's describe the LW settings required for rendering a float image format. Set the Image format exr(Figure 11a). I also usually set the Image compression to PIZ and the bits per channel to 16 bits(half-float) which results in smaller file sizes(Figure 11b). This is done via the Vray image format options interface which pops up when you press Image format options. As an aside, I recommend using the handy File Name Prefix keywords <Scene>_<Layer> which tells Maya to embed the name of the Maya scene in the output image name.

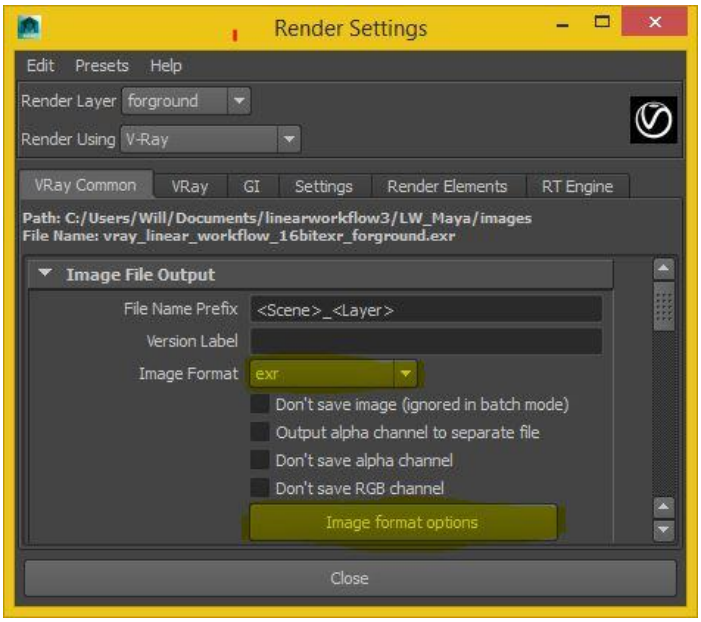

Figure 11a

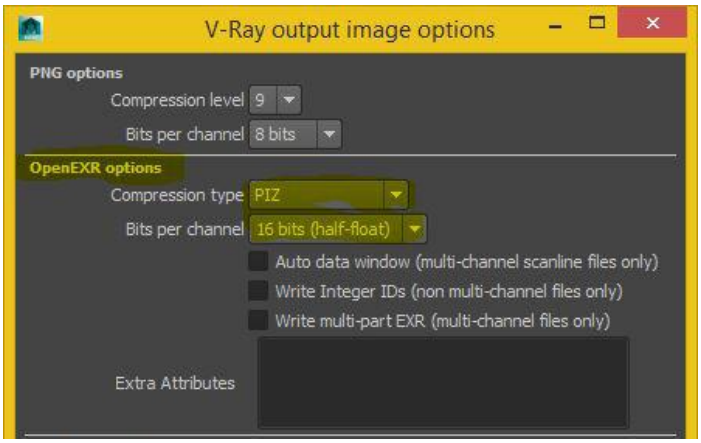

Figure 11b

Material settings for Vray

Next, let's cover some of the hypershade nodes in these scenes. The background texture (a sRGB gamma boosted png file of the famous Kodak Marcie image widely used in color grading) is a surfaceShader materials with a file node attached to the color attribute.

File Node settings for Vray

By default the File node Color Profile is set to "Use Default Input Profile", which is a Maya feature that enables global settings of color profiles for File nodes(Figure 12a). Unfortunately Vray ignores the Color Profile pull-down choices of the file node. Vray for Maya has its own way of establishing the color profile used in inverting incoming texture images.

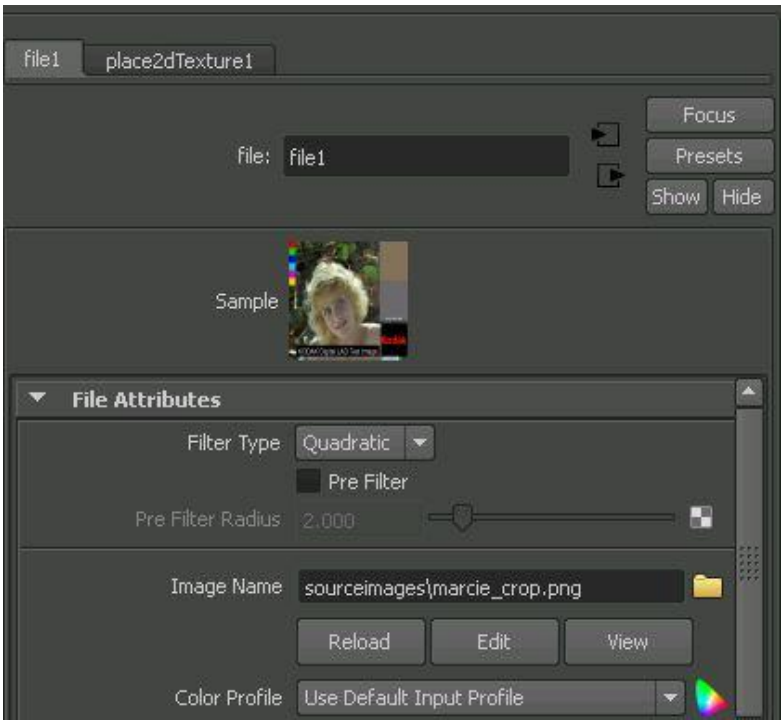

Figure 12a

Although the "linear workflow" toggle mentioned earlier does something its only limited to the Diffuse Color inputs of Vray materials. All the other color attributes that use sRGB texture images also need to be inverted. Add to this that Chaos has been warning that this toggle is being deprecated, results in my recommendation that each File nodes needs to have the special Vray attributes created. This is pretty inconvenient if you have may File nodes. Figure 12b is a snapshot of the Hypershade File node showing how you create the special Vray attribute required to control color profile of incoming texture images. Under the Attribute pulldown click on Vray then Texture input gamma.

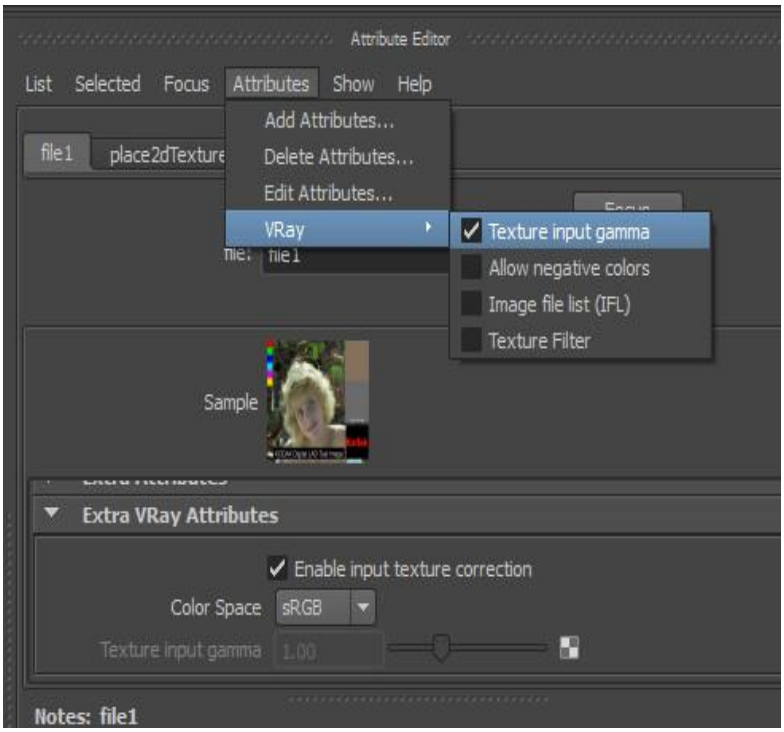

Figure 12b

Once the extra attribute is created turn on Enable input texture correction and choose sRGB for Color Space. This will result in Vray internally preproessing the texture based on an inverted sRGB profile, which will makes the image's color values linear.

Hypershade solid color RGB values for Vray

Taking a look at the hypershade material network for the brown sphere which has a Maya Phong material attached, we notice there is a gamma node directly connected to the color attribute of the phongShader. The RGB values of the color attribute of the gamma node were eye drop picked from the original Marcie png image in a paint program. Those RGB values were typed into the Value attribute of the gamma node. We then set all three Gamma values to 0.4545. This inverts the specific RGB color so that it fits into our LW setup and matches the background color. This would be even more accurate if there were a hypershade color node that included the means to specify sRGB, but its close enough for most eyes. The grey sphere is done in the same way. For single colors this is a more efficient approach than using a ramp node and a gamma node. For ramp nodes you have to attach a gamma node to the output in order to invert the gamma. It's unfortunate that Maya does not have a Color Profile pull-down in these nodes just like the file node. If they had, this manual aspect of setting up LW in Maya would be unnecessary.

Figure 42

Hypershade network ramp RGB values Vray

For ramp nodes you have to attach a gamma node to the output in order to invert the gamma. Again it's unfortunate that Maya does not have a Color Profile pull-down in these nodes just like the file node. If they had, this manual aspect of setting up LW in Maya would be unnecessary.

Figure 13