



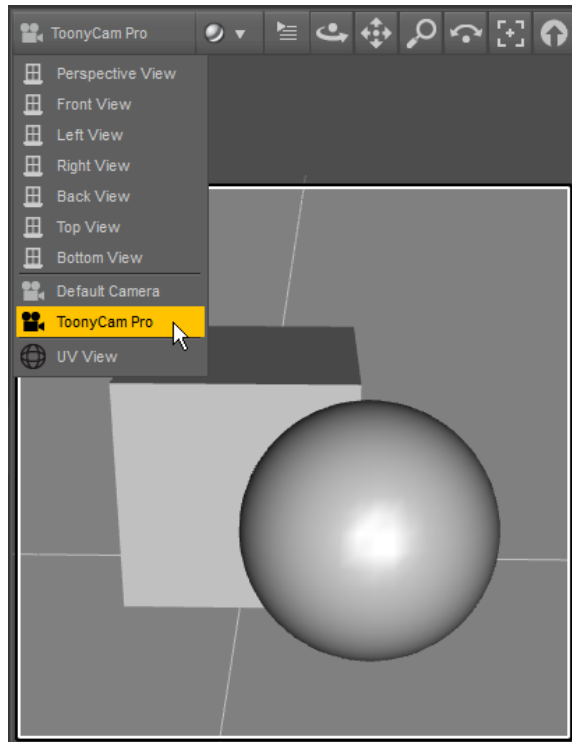
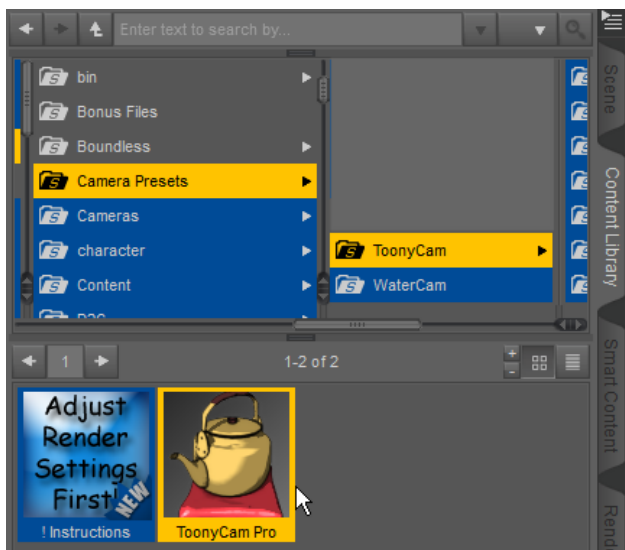
ToonyCam Pro Documentation

© 2014 zigraphix

ToonyCam Pro can be used in a very simple way, or its use can be customized for different special effects.

Simplest operation:

1. Load ToonyCam Pro from the Camera Presets folder
2. After you load the camera, make sure you have selected it as your render camera, i.e. the one you're looking through in the viewport.
3. Use the render settings script **!Render Outlines On**



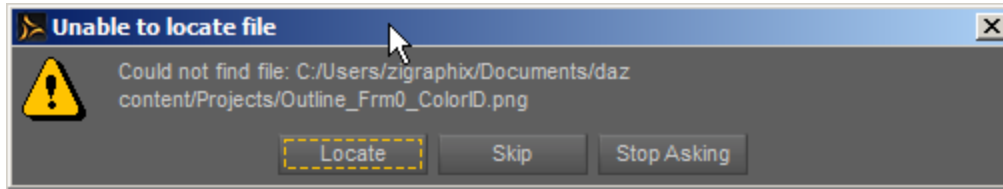


At this point, you can simply render. By default, the outlines will be based on the “Normal” direction of objects in the scene, i.e. where objects curve parallel to the camera, there will be a black line. Also by default, colors will be simplified to three shades of color per object, giving a cell shaded appearance, and the separate shades will blend between them at 25%.



Note: If you save and reload a scene, you will see an error message about missing outline images. This message can be safely ignored. New outline images will be generated and

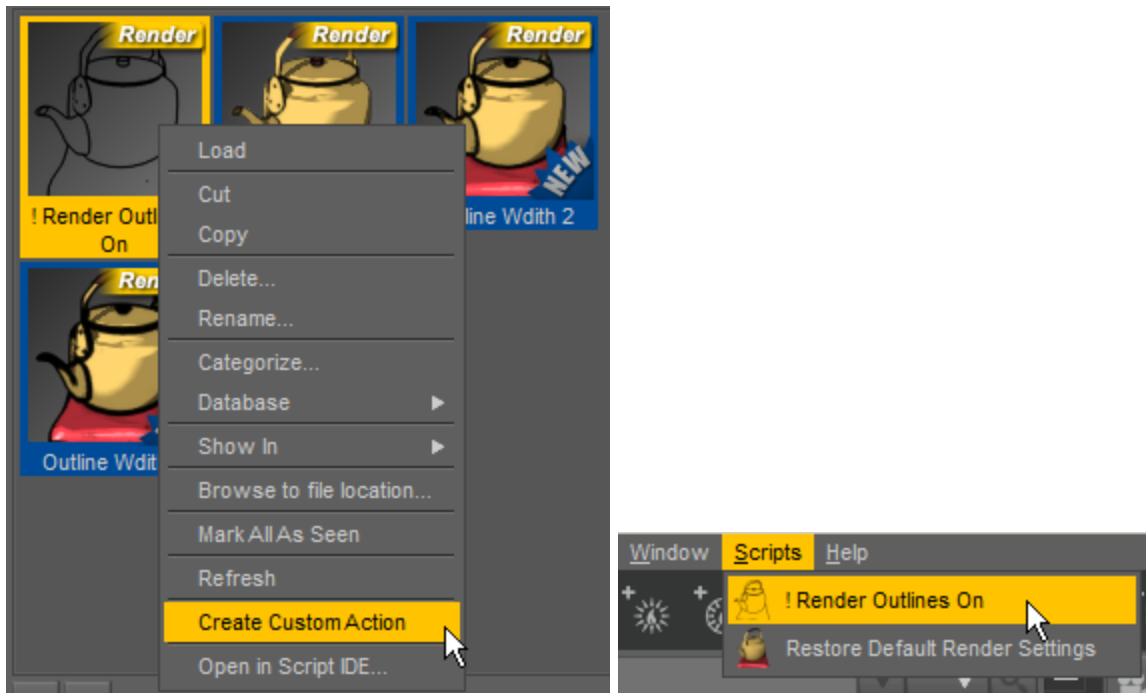
assigned to the camera the next time the scene is rendered.



Intermediate Options:

Creating a Custom Action

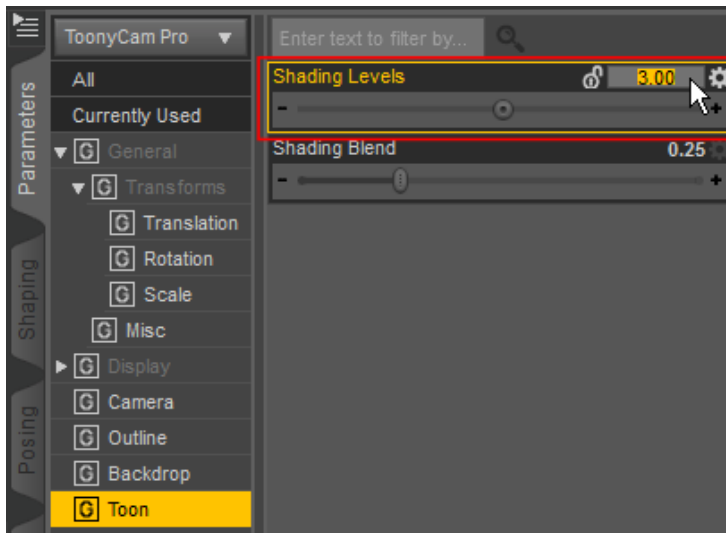
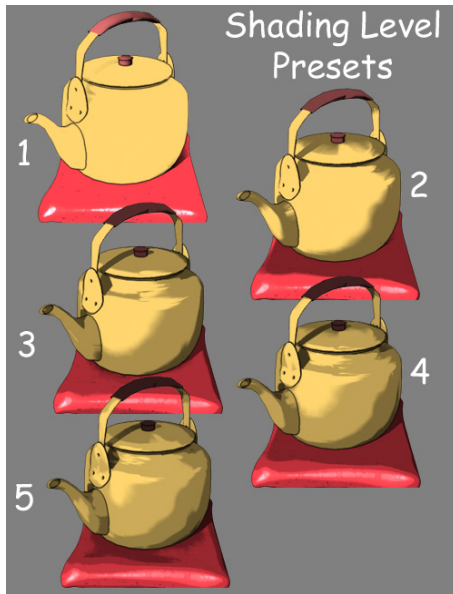
You may find it convenient to create Custom Actions in DAZ Studio for enabling and disabling Outline Rendering. To create a Custom Action, right-click (Command-Click on a Mac) on the **!Render Outlines On** script icon, and select "Create Custom Action". You will now see "!Render Outlines On" as a menu item in DAZ Studio. Similarly, right-click **Restore Default Render Settings** and add this to your Actions menu. You can now easily switch back and forth between the two render modes. Don't forget to switch cameras when you switch render modes!



Shading Levels

If you want to change the number of shades, there are presets in the Shading Levels subfolder, or you can go to the Parameters pane of the ToonyCam Pro camera in the Toon section.

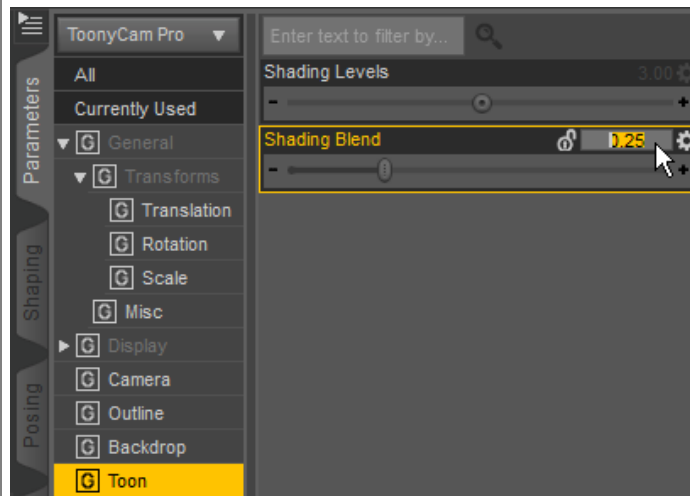
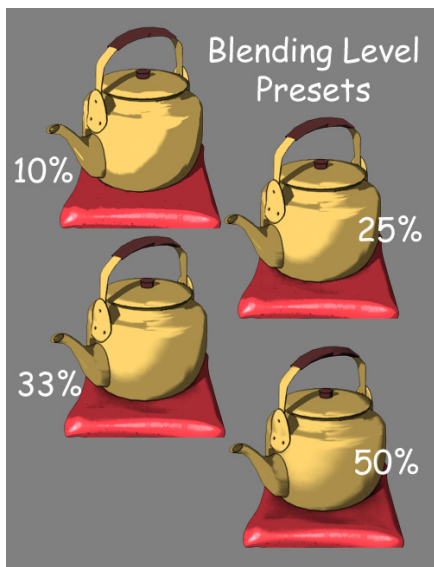
Note: Select ToonyCam Pro Camera in the Scene Tab before using these presets or attempting to edit Parameters.



Shading Blend

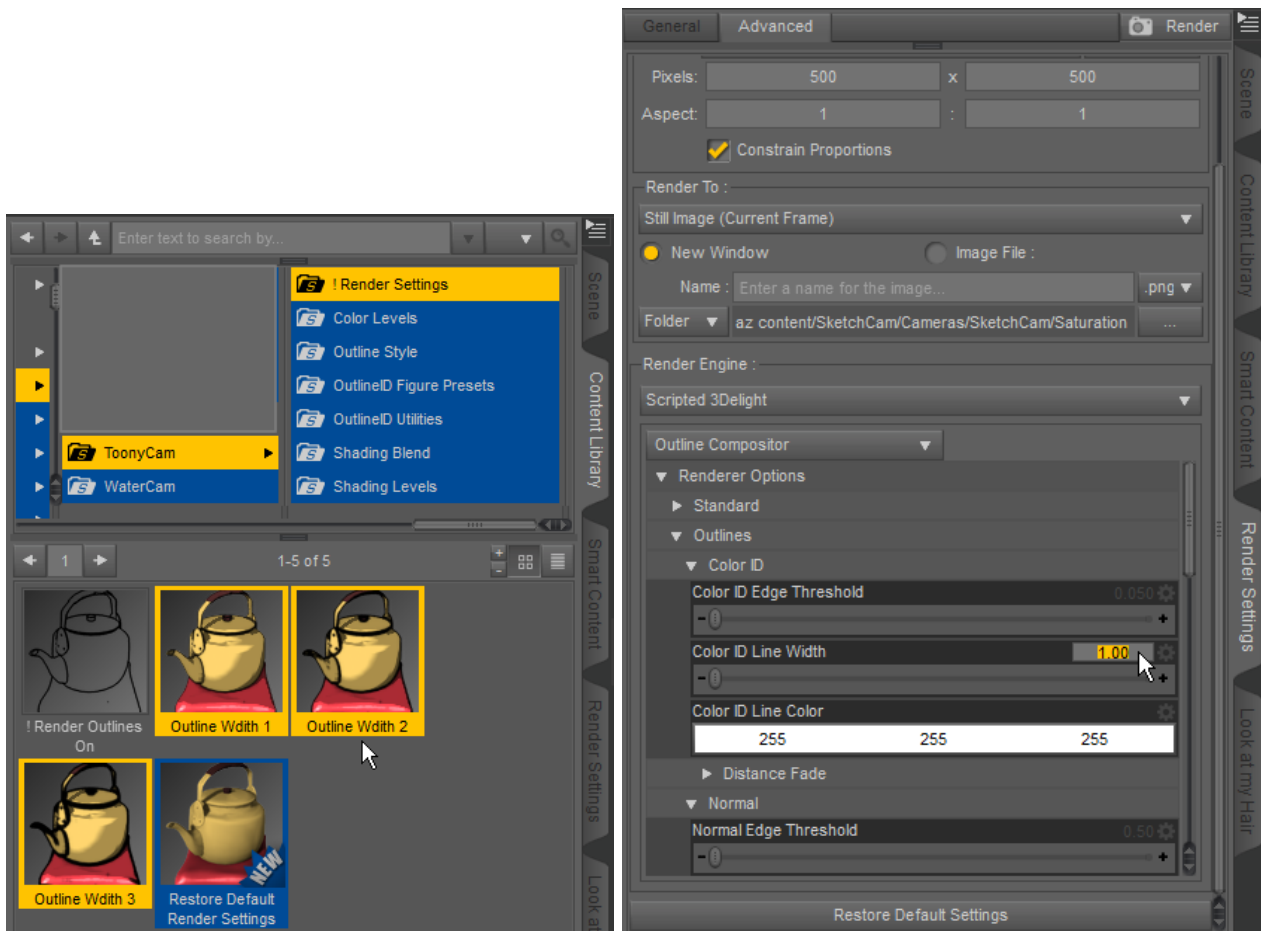
If you want to change the amount the different shading levels blend, there are presets, or again you can adjust this manually in the Properties pane.

Note: Select ToonyCam Pro Camera in the Scene Tab before using these presets or attempting to edit Parameters.



Outline Thickness

If you want to change the thickness of the outlines, you can use the presets in the Render Settings folder, or manually edit the Render Settings pane. The setting for this is a little harder to get to, because it's several levels inside the Scripted Render control area. The presets are much easier to use.



Line widths can be set individually for the different styles of outlines (see below). Line widths are measured in pixels in the final render, so large renders may look better with thicker outlines, and vice versa.

Note: the Line Color settings within the Outline Compositor Render Settings must be left set to White for the camera set to function correctly. Use the Parameters tab for the ToonyCam Pro Camera to change the outline color in the render.

Advanced Options

Outline Styles

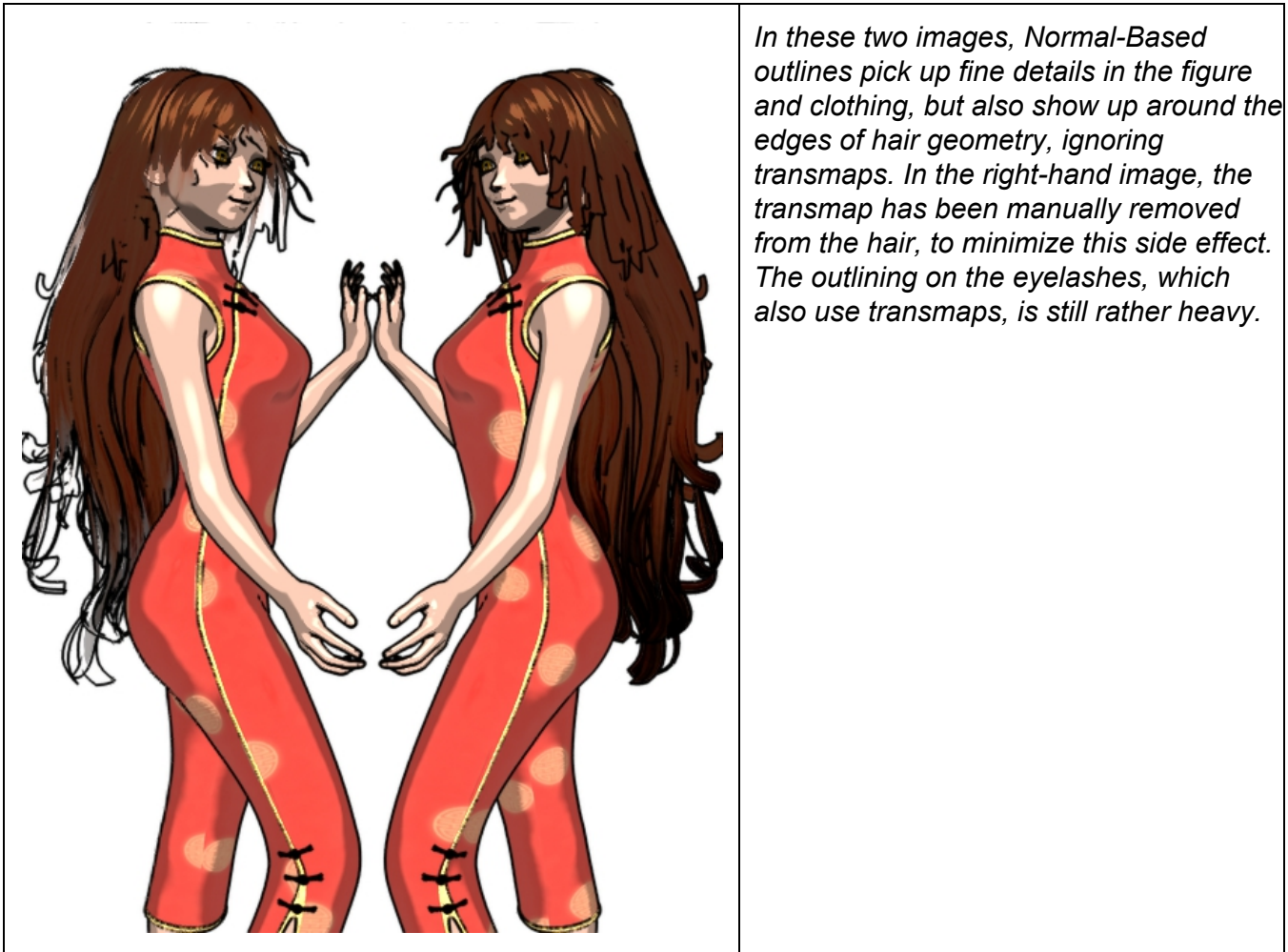
The choice that is most complex is if you decide to use one of the other two methods of outline detection. Again, there are presets to turn the three methods on or off, and you can use any combination in a single render, including no outlines.



Normal-Based Outlines

This is the default. Outlines are created based on changes in the curvature of the geometry. Displacement and Normal maps are taken into account. Transparency is *not* taken into account. Normal-based outlines are created for every surface in the scene that includes the “Outline Depth” parameter, set to “True.” By default, this is all surfaces in the

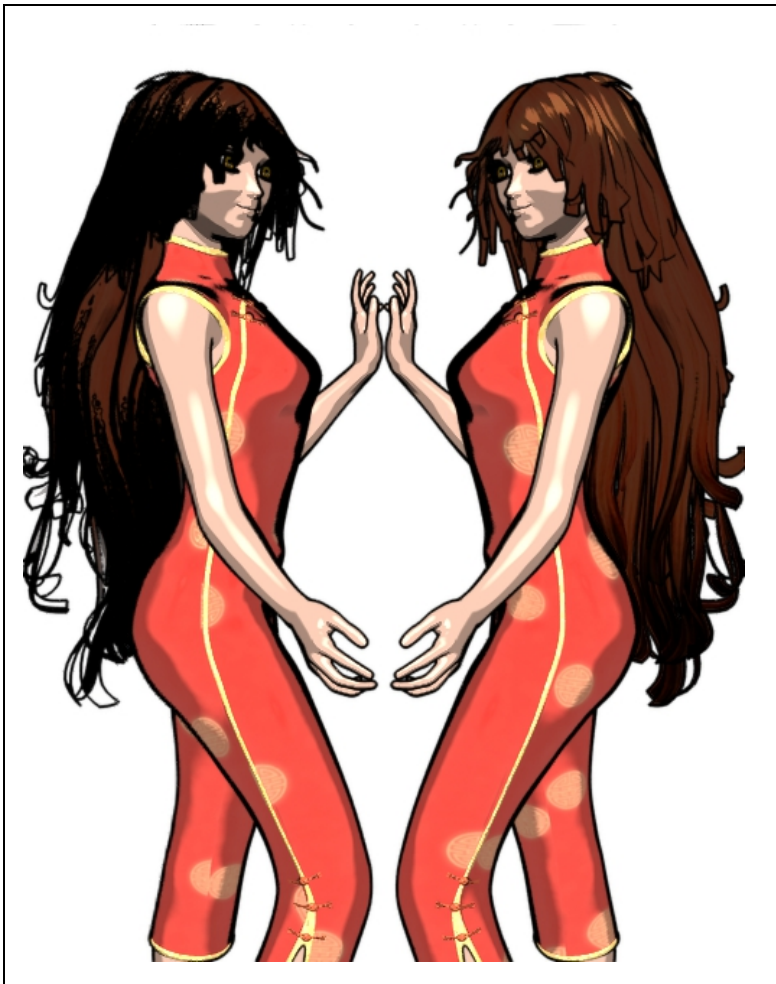
scene.



Depth-Based Outlines



The “Depth” method checks each point in the render to see how far it is from the camera, and if this changes more than a certain amount from the point next to it, an outline is drawn. This method can draw thick outlines near the edges of large curves, and is sensitive to the scale of the image. It is a stylistic choice. You can also go into the Render Settings panel and tweak the sensitivity if you get thick lines or the outlines fade out more than you want. Depth-based outlines are created for every surface in the scene that includes the “Outline Depth” parameter, set to “True.” By default, this is all surfaces in the scene.

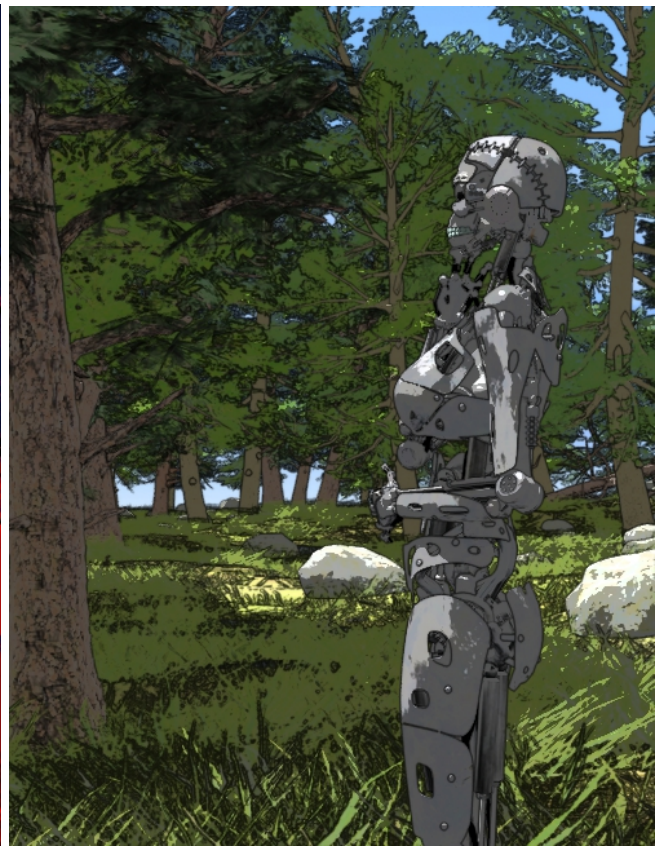


In these two images, Depth-Based outlines accent the curves of the geometry, but ignore transparency mapping on the hair. You may also see heavy black areas on geometry that is nearly perpendicular to the camera plane. (Again, in the right-hand image, the transparency has been removed from the hair, but the outlining on the transmapped eyelashes is fairly heavy.)

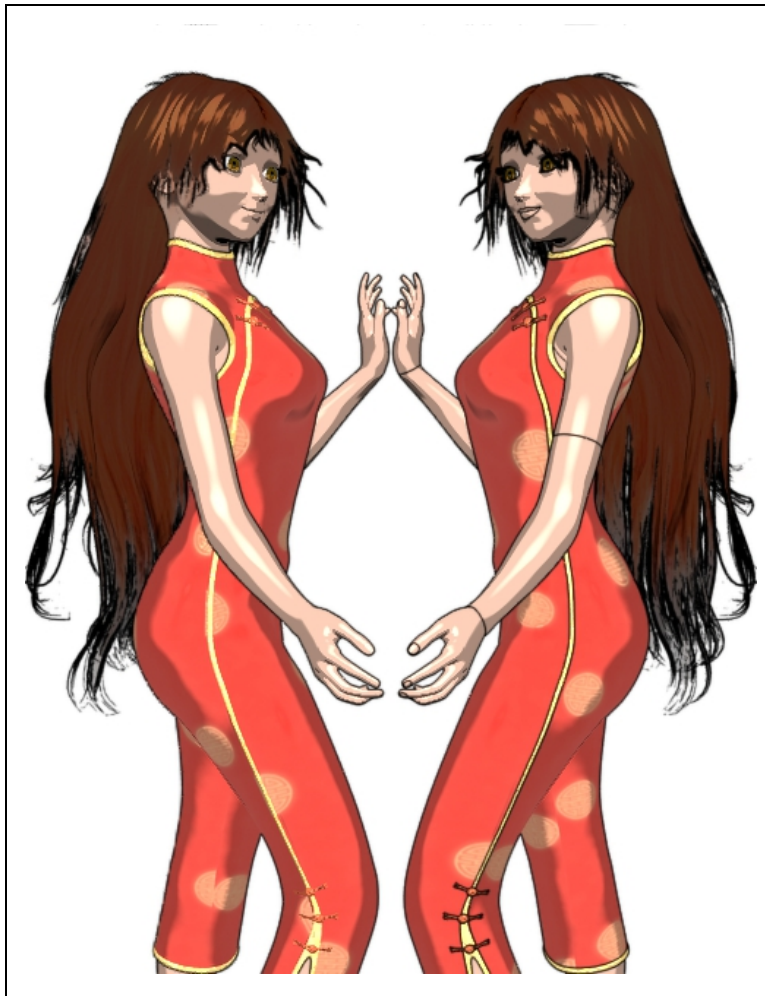


OutlineID-Based Outlines

The OutlineID method offers the most control, but may also require more manual surface parameter editing than other methods. This method of generating outlines first creates a version of your scene in which each object in the scene is rendered using its OutlineID color.



OutlineID is a new parameter added to every surface in your scene when you enable outline rendering. The ToonyCam Pro scripts go through all the objects and material zones in your scene and try to set a unique OutlineID color value to each. These colors do not affect the final render. A first-pass render is then created using these “hidden” colors. On larger images, you may be able to see this before the render replaces this colored image with the final render with outlines. The outlines are generated on the boundaries between different colors in the first-pass render. So if you just use the OutlineID settings on default, you’ll get lines around all the objects in your scenes, and also along the edges of material zones. This works well in many cases, but human figures tend to have material zone boundaries across the limbs in places where outlines are usually not wanted.



In this image, the OutlineID method is used. The figure on the left uses one OutlineID per node (or object) in the scene. The figure on the right uses a unique OutlineID per material zone. Note that more details are picked up in the clothing, but there is also outlining at the wrist, upper arm, lips, etc., wherever the surface material zone changes. Fortunately, this is easily corrected with the provided presets. Also note that with the OutlineID method, the nose is not outlined, as the render engine has no way to distinguish it from the rest of the face.

ToonyCam Pro provides presets that set the OutlineIDs of many common figures to all the same value except for the eyes, which people tend to want to outline. ToonyCam Pro also includes helper scripts to reset the OutlineIDs in the scene to unique values per node, per material zone, or based on the current Diffuse Color or Diffuse Map file name of the surface. Scripts are provided to set the OutlineIDs of all objects in the scene, or only selected objects. You can also go into the Surfaces pane for each object and manually set the OutlineIDs. Make them all the same color (generally black) where you don't want edge detection and outlines. Make them different if you want an outline between those two surfaces.

The OutlineID method is most useful if there is a lot of transparency in the scene, which is especially common with hair and foliage. The Normal and Depth methods of edge detection are not able to take transparency into account. So if you are using transmapped hair, you'll probably want to switch to the OutlineID method, or you'll get outlines where the geometry of the hair is, not where the visible parts are supposed to be. But if there is a visible "scalp" material zone in the hair, you may want to set that material to the same OutlineID as the skin on the figure's face, to avoid an outline between the edge of the scalp and the skin.

Lighting

ToonyCam Pro will work with any lights, but for toon renders, one main light at 100%, at an angle from

the camera, and a fill light at a different angle at 25-30% is usually all you need. ToonyCam Pro does not support caustic or volumetric effects (these require special cameras-- see below).

Surface Shaders

ToonyCam Pro works seamlessly with nearly all surface shaders, including metallics, displacement/fur, subsurface scattering, UberSurface, toon style surface shaders, etc. (see “Garibaldi Hair” in the Troubleshooting section for the only known exception). The toon effects may combine with surface shaders in interesting ways. If you are trying to combine ToonyCam Pro with surface shaders like Visual Style or Manga Style, you need to be familiar with how those shaders work first. Please see “Optimizing Outline Renders” at the end of this document for some suggestions.

Other Effect Cameras

Only one camera can be used per render. To combine ToonyCam Pro with the effects from other cameras, you will need to composite the images manually using an external program.

Technical Details

The following are details about the functionality of DAZ Studio itself that may help to clarify how this product works.

There are three different aspects of DAZ Studio interacting to produce ToonyCam Pro renders:

Custom Camera

ToonyCam Pro provides a custom camera. This is very different from a camera preset. Camera preset products create one or more standard DAZ Studio cameras at different points in your scene, possibly with some DOF settings, but usually only “custom” in the sense that they are at particular positions and point in particular directions, and you can switch between them.

The code used for the ToonyCam Pro camera provides additional options not present in a standard DAZ Studio camera. The advantage of using this method to change the render, compared to “toon” surface shaders, is that you only have to make the changes in one place, the camera parameters, and the changes affect your entire scene. The disadvantage is that the changes affect your entire scene, you can’t just use one setting in part of your scene and a different setting in a different part. (See below for some ways around this limitation.)

ToonyCam Pro is implemented both as a “Scene Subset” in DAZ Studio, and as a Camera Preset. Loading a Camera Preset selects the new Camera in the viewport as the rendering camera automatically, but also deletes all other cameras in the scene, because this is the default behavior of DAZ Studio. ToonyCam Pro is also implemented as a Scene Subset. The advantage with a Scene Subset is that existing cameras aren’t deleted from the scene. The disadvantage is that the new camera isn’t automatically enabled when it is loaded.

The ToonyCam Pro Camera **must** be used only after running the included Render Script. It will render black otherwise.

Scripted Rendering

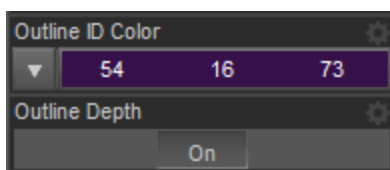
This product also involves a feature in DAZ Studio called “Scripted Rendering.” This makes changes on the Render Settings tab, primarily switching from “3Delight” to “Scripted 3Delight,” and selecting a script to control the render. Again, this is different from most other DAZ Studio products. By default, DAZ Studio uses the same default “script” every time it renders—it calculates shadow maps (if you are using them), then renders the scene using default parameters that can be adjusted somewhat from the Render Settings tab. Scripted Rendering allows changes to the parameters are on the Render Settings tab, as well as altering other functions of the 3Delight render. ToonyCam Pro uses customized code based on resources included with DAZ Studio 4.x and the scripting SDK (Software Development Kit) provided by DAZ 3D (used by permission) to create outline render passes, and then sets the custom camera to locate those images and composite them into the final render.

Render settings affect every camera in the scene. Again, this has the advantage that if you adjust the render settings, you don’t have to also adjust individual cameras, and the disadvantage that you can’t adjust individual cameras to have different render settings in the same scene. To make the presets for the render settings, a special script is used with some hand editing, but the camera is designed so that it isn’t necessary to change render settings very much—the outline thickness is the most commonly changed Render Settings parameter, and presets are provided.

The Render Script **must** be used with a camera that supports this mode, e.g. ToonyCam Pro. You will see an error with any other camera. The render will still complete, but no outlines will be included.

Surface Parameters

Finally, the custom camera and the render script make use of parameters on the surfaces tab. Since the whole point of using a camera is to avoid having to manually edit every surface, a few scripts are included to handle this for you, and the setup scripts automatically try to make smart choices for you in preparing surfaces for the render—without changing any existing surface settings. The two new parameters are **Outline ID Color** and **Outline Depth**.



OutlineID

The user is able to adjust OutlineID to selectively outline objects or material zones in the scene. As explained above, this is implemented as a color, and a separate pass render creates an image that is used to determine where to draw outlines on the basis of node and material zone values.

Outline Depth

This parameter can be used to selectively disable outlining around surfaces in Normal and Depth outlining modes. However, if an object is excluded from outlining, the outlines of other objects behind

that object will appear.

Every surface in the scene **must** have these two parameters included to be outlined. If you add new objects to your scene after enabling outline rendering, select each new object and use one of the included presets to set these parameters on the selected object. This will add the missing parameters. If the parameters are not added, ToonyCam Pro will treat those surfaces as though the OutlineID Color is black (0, 0, 0) and the Outline Depth is set to "False." This will usually result in those surfaces not being outlined, regardless of other camera or render settings.



Optimizing Outline Renders

Partial Outlining

You can easily disable outlining on all objects in the scene and only enable it selectively, with a few caveats. The procedure differs slightly depending on which Outline Style you will use.

OutlineID

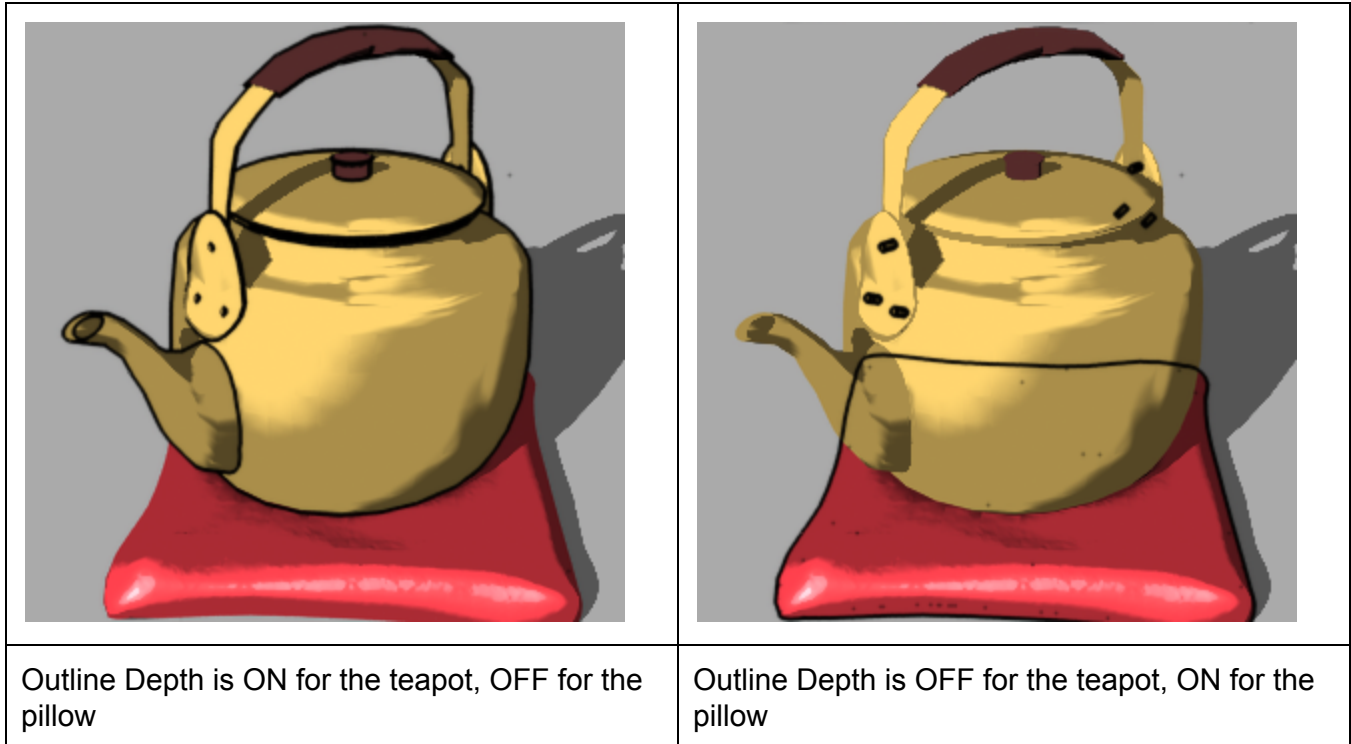
With the OutlineID method turned on in the camera, you can set the OutlineID of all surfaces in the scene to (0, 0, 0) (Black) using the included "OutlineIDs OFF" preset. Since all objects and surfaces in the scene will then have the same OutlineID, there will be no distinctions between surfaces in the OutlineID pass render, so there will be no outlines drawn. You can then select specific objects in the scene and use "Selected OutlineIDs..." presets to enable outlines for just those objects and surfaces, or you can manually edit the OutlineIDs in the Surfaces tab.

	
OutlineIDs are non-black and unique for teapot, black for pillow	OutlineIDs are black for teapot, non-black for pillow

Normal

With the Normal outline method turned on in the camera, you can disable outlines for all objects and surfaces in the scene by using the “All Outlines OFF” preset in the “Outlines On-Off” folder. Then, select objects and use the “Selected Outlines ON” preset, or manually edit the “Outline Depth” parameter in the Surfaces tab.

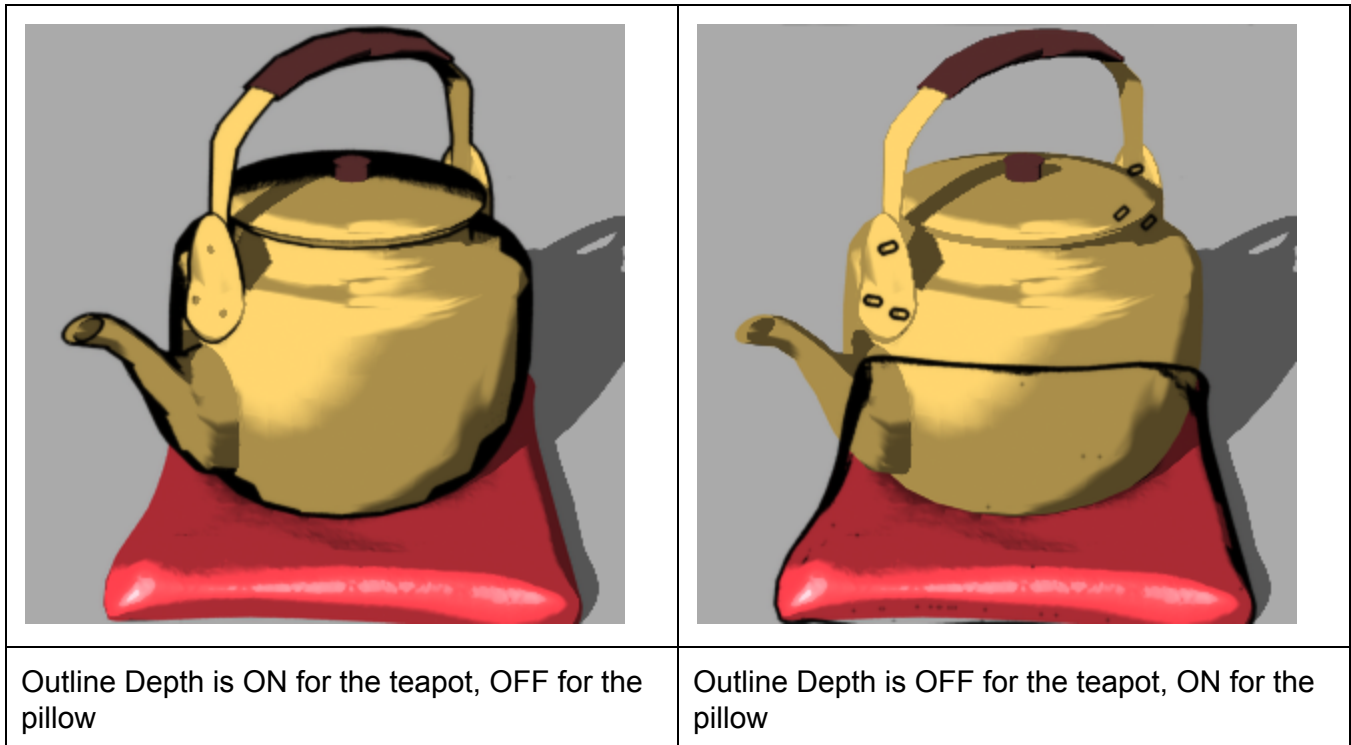
Note: Outlines in the back of the scene (farthest from the camera) may show through non-outlined objects closer to the camera.



Depth

With the Depth outline method turned on in the camera, you can disable outlines for all objects and surfaces in the scene by using the “All Outlines OFF” preset in the “Outlines On-Off” folder. Then, select objects and use the “Selected Outlines ON” preset, or manually edit the “Outline Depth” parameter in the Surfaces tab.

Note: Outlines in the back of the scene (farthest from the camera) may show through non-outlined objects closer to the camera.



Mixed Outline Method Scenes

By carefully combining the OutlineID method with either Normal or Depth based outlines, it is possible to have the advantages of both. Here is a process to help you get started.

Set the OutlineID of all items in the scene to Black using the included preset (OutlineIDs OFF). Set Outlines ON using "Outline Depth" for all objects in the scene. Enable both OutlineID and (for example) Normal outlines in the camera. At this point, all surfaces will be outlined using the Normal method, and NOT using the OutlineID method. Then, select surfaces in your scene to switch from Normal outlining to OutlineID. For those surfaces only, turn "Outline Depth" OFF and set the OutlineID colors to unique values other than Black. Those surfaces only will be outlined by OutlineID rather than by Normal. Usually you will select objects that need transparency to outline using OutlineID rather than Normal or Depth, but any Outline Depth objects will show outlines through OutlineID-only objects closer to the camera. See the examples below.



Both OutlineID and Depth based outlines are turned on in the camera.

The teapot has unique OutlineIDs set for each surface material zone, but Outline Depth set to OFF.

The pillow has Outline Depth set to ON, but the OutlineID color is black. The varied line thickness is interesting, but shows through the teapot.

The flowers and leaves have Outline Depth set to OFF and the OutlineID to Black, producing no outlines. This works well for simplifying backgrounds and drawing attention to the foreground objects.



OutlineID and Normal outlines are turned on in the camera.

The teapot has OutlineIDs set to black, and Outline Depth set to ON.

The pillow has Outline Depth set to OFF, but the OutlineID color is non-black.

The Normal method picks up more detail on the teapot than the OutlineID method, which depends on individual material zones.

Using OutlineID on the flowers and leaves allows the transparency maps to be supported.

This provides the highest level of outline detail for all objects in this scene.

Setting Solid Diffuse Colors

Although ToonyCam Pro will work with existing Diffuse Color Maps, you may prefer to set the Diffuse channel of your surfaces to solid colors. This is set in the Surfaces tab. Select surfaces to edit, and reset the Image Map of the Diffuse Channel to “None,” then set the surface Diffuse color to a solid color.

Troubleshooting

My image is all black!

Make sure you have remembered to turn on Outline Rendering. See Step 3 in the “Simplest Operation” section at the start of this document.

My renders don't have outlines!

There are many things that can cause this. Most likely you are rendering through a camera other than ToonyCam Pro. Be sure to make the ToonyCam Pro camera active in the viewport you are using to render. See Step 2 in the "Simplest Operation" section at the start of this document.

Also, make sure you have turned on at least one Outline Style, and depending on your Outline Style, that you have not disabled outlines in all the objects in your scene. If in doubt, use the "All OutlineIDs per Node" and "All Outlines ON" preset scripts in your scene.

Some elements of my scene don't have outlines, but some do.

If you add new objects to your scene after enabling outline rendering, they won't have the two new required surface parameters, OutlineID and Outline Depth. Select each new object and use one of the included presets to set these parameters on the selected object. This will add the missing parameters.

Outlines don't align with the image in my render!

Are you using spot render? If so, be sure you select a square area (or as close as possible). Otherwise, the dimensions of the first render pass, which creates the outlines, won't be overlaid on the final pass render correctly.

This doesn't work with Garibaldi hair!

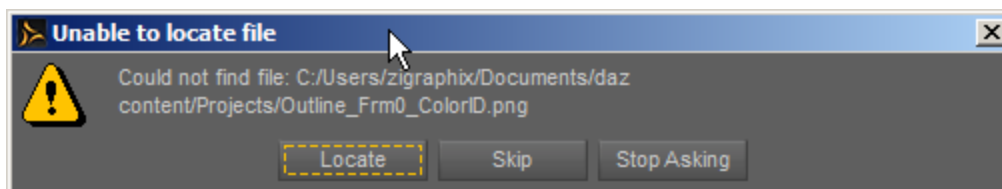
The Garibaldi hair shader does not seem to allow the new parameters in hair surfaces. To use Garibaldi hair, turn rendering on *before* adding the Garibaldi hair, and do not use the presets to add the missing parameters to the new surfaces. Garibaldi hair will not be outlined, but will otherwise render correctly. Another workaround is to use a different shader on the hair.

I have outlines around the edges of the hair, across the face!

This happens because the hair model has a "scalp" material zone with transparency mapping. You may need to manually edit settings in the Surfaces tab to correct this. In Normal or Depth outline renders, try finding the Scalp material zone and turn off Outline Depth. In OutlineID renders, set the OutlineID for the Scalp material zone to match the OutlineID of the skin on the face. If you are using the included Outline ID Figure Presets, the skin color is set to (128, 255, 255).

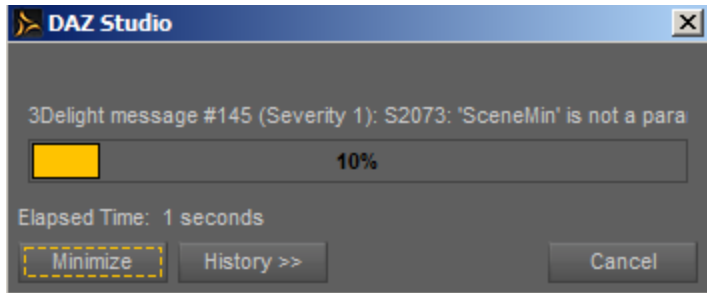
I see errors when I load a scene I saved earlier!

If you save and reload a scene, you will see an error message about missing outline images. This message can be safely ignored (Click "Skip" or "Stop Asking"). New outline images will be generated and assigned to the camera the next time the scene is rendered.



I see errors during render!

In DAZ Studio 4.6.2.118 and earlier (at least), rendering using Shader Mixer based cameras generates an error. This error can be safely ignored. The bug has been reported and confirmed by DAZ 3D, and will hopefully be fixed in a future release of DAZ Studio.



If I clear the current scene and load a new scene, my new scene won't render.

When using outline rendering, please quit and restart DAZ Studio between scenes. An unknown bug causes the Outline Render method to fail after the scene is cleared.

My Manga Style renders look wrong when I use ToonyCam Pro!

Manga Style shaders require a very low value (significantly less than 1.00) in the Shading Rate render parameter. Because the Shading Rate has to be manually implemented as part of a Scripted Renderer setting, it will change when Scripted Rendering is enabled. The default value in Scripted Render is 1.0, and resetting the outline width will also reset this value to 1.0.

More information about Shading Rate and other shading parameters can be found here:
<http://renderman.pixar.com/view/overview-of-the-render-settings>

The attached screen shot shows where to reset it.

