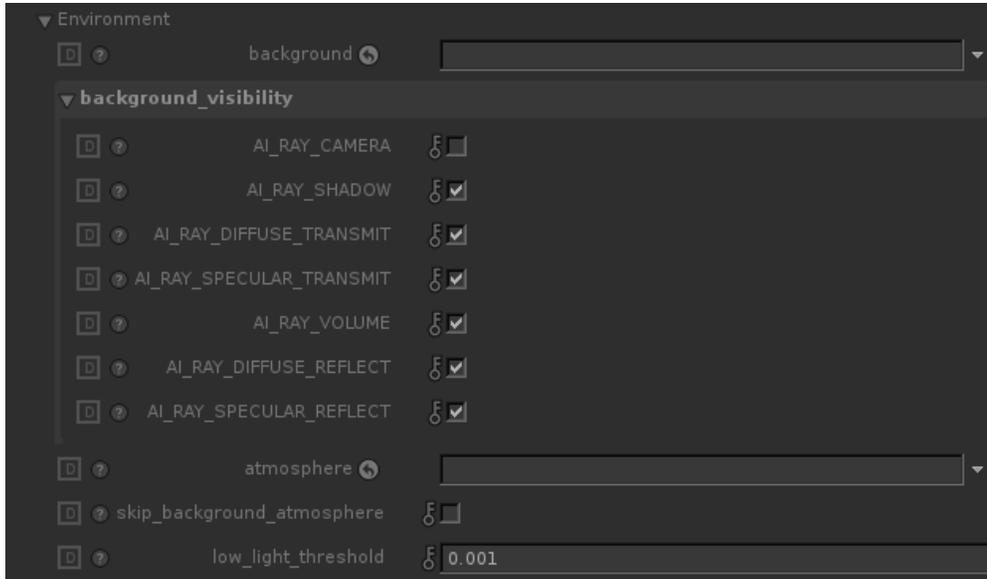


Environment



background

The name of the shader to be evaluated as background for rays that don't hit any geometry.

You can drag a material location to this parameter, which material defines the shader to be run when rays hit the background.

Note that it is preferred instead to use a [skydome_light](#) instead to generate a background, and there is an additional shader slot for background rays there.

background_visibility

Defines the Background visibility mask.

Note that KtoA defaults to a background that is invisible to camera rays, as that is the typical workflow for images with compositing as their destination.

atmosphere

The name of the shader to be evaluated for the global atmosphere evaluation.

You can drag a material location to this parameter, which material defines the shader to be run when rays evaluate the atmosphere volumetrics.

skip_background_atmosphere

It doesn't apply the atmosphere shader in the background.

low_light_threshold

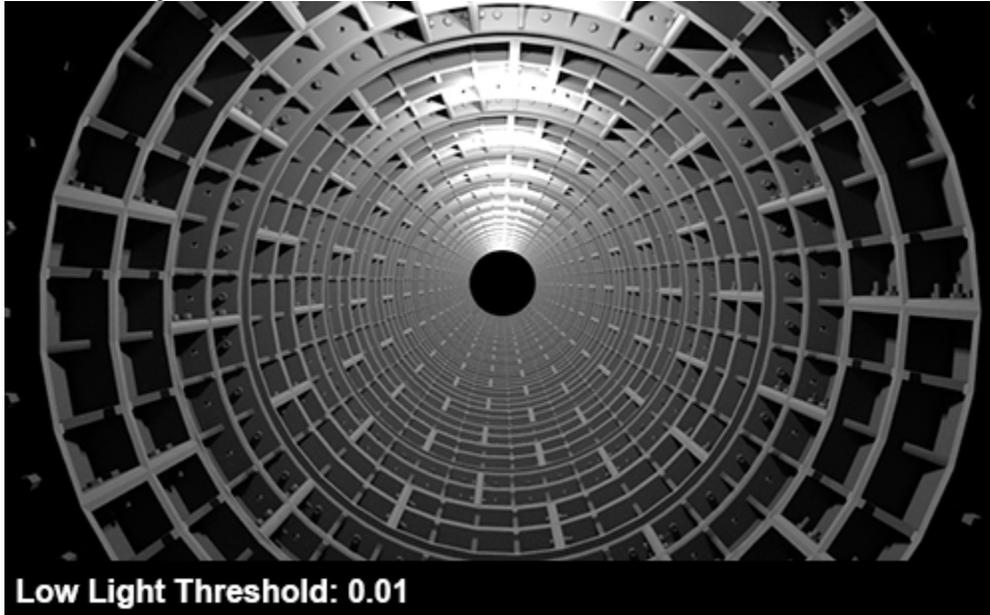
Raising this value can speed up rendering by allowing Arnold to ignore tracing a shadow ray for light samples whose light contribution is below a certain value. The point of *Low Light Threshold* is to save casting shadow rays when Arnold knows that the error from not casting that ray is below a certain amount. This makes sense because below a certain threshold there will be no perceptible difference between shadowed and unshadowed areas.

Roughly, what is taken into account to discard a shadow ray is its maximum possible incoming irradiance, which is linearly related to the light's intensity and decreases with distance. The specific number will vary depending on your exposure and light intensity values, but the default value is reasonable for the default exposure/light intensity values. This will typically help more in scenes with lots of lights with a finite area of influence,

like a cityscape for example. It will help much less with a single strong source of light.

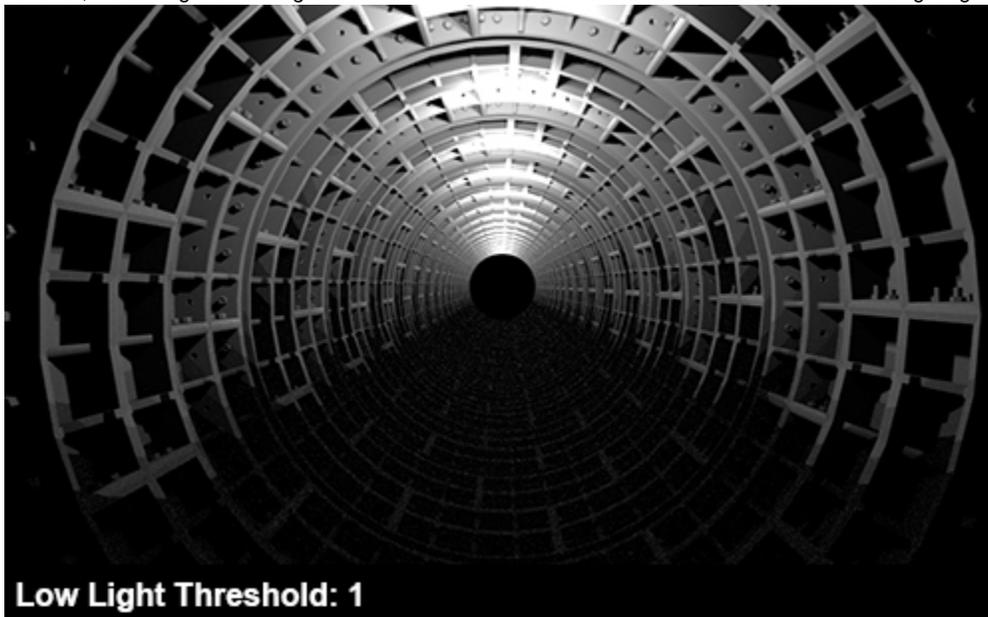
You should never be able to see the boundary when this optimization kicks in. If you can see it, you need to make the threshold lower. The aim is to prune shadow rays that don't contribute to the final image

In the example below, increasing the *Low Light Threshold* to 0.1 shows a good speedup in render time (in this case, more than twice as fast) with little noticeable difference in the image.

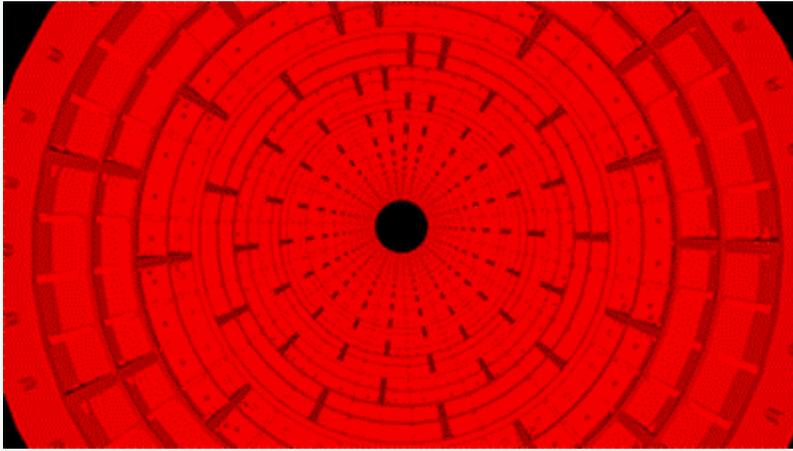


A slightly higher threshold can offer significant rendering speed increases without noticeably darkening the image (rollover image).

However, increasing the *Low Light Threshold* value too much can have an adverse effect on the lighting:



The Utility shader's *nlights* color mode can be used to see how many lights are influencing a shading point when adjusting the *Low Light Threshold*.



Low Light Threshold: 0 to 1. Red = all lights, blue = low values, black = no lights.