PxrUnified

Image from *Incredibles 2*

PxrUnified is initially included in RenderMan as an experimental integrator. Eventually this integrator will become full-featured and ready for reliable production rendering.

Note that PxrUnified does not *currently* support

- Holdout workflow
- LPE output in Bidirectional mode
- Mattes (you may have errors)
- Shadow Collectors
- Accumulated Opacity (compTrans)
- Exclude Shadow Subsets

PxrUnified is the integrator Pixar relies on for its own film and animated shorts. It implements both the forward path tracing algorithm and bidirectional choices along with faster caustics using the Manifold Next Event Estimation technique here called **Manifold Walk**. We also include two options for **Indirect Guiding** that improves indirect lighting by sampling from the better lit or more important areas of the scene. One indirect guiding method is based on Practical Path Guiding For Efficient Light-Transport Simulation by Muller et al. This approach has a relatively high overhead, but can be beneficial for some scenes, particularly those with highly specular indirect interactions. The second approach, which we call risPathGuiding, has a much lower overhead and is more effective for diffuse and glossy scenes. See Path Guiding.

Note that the Max Path Length is the single control used to control global trace depth, PxrUnified ignores the settings for separate specular and diffuse trace depth. The implication is that a simpler control is intuitive but some scenes may take longer to render and appear differently than in other integrators where users have optimized the bounces based on their scene. Setting the control to 0 means only direct lighting and camera visible effects. Increasing the setting allows light to travel further into and off of objects in the scene.

For direct illumination, the numLightSamples and numBxdfSamples parameters determine the number of light vs. Bxdf (material) direct illumination samples, and the resulting direct illumination contributions are combined using multiple importance sampling.

For indirect illumination, the numIndirectSamples parameter controls whether rays are sent to sample indirect paths. Only values of 0 and 1 are currently supported.

SubSurface Over-sampling is used to help reduce the noise when rendering complex lighting and objects with subsurface shading enabled.

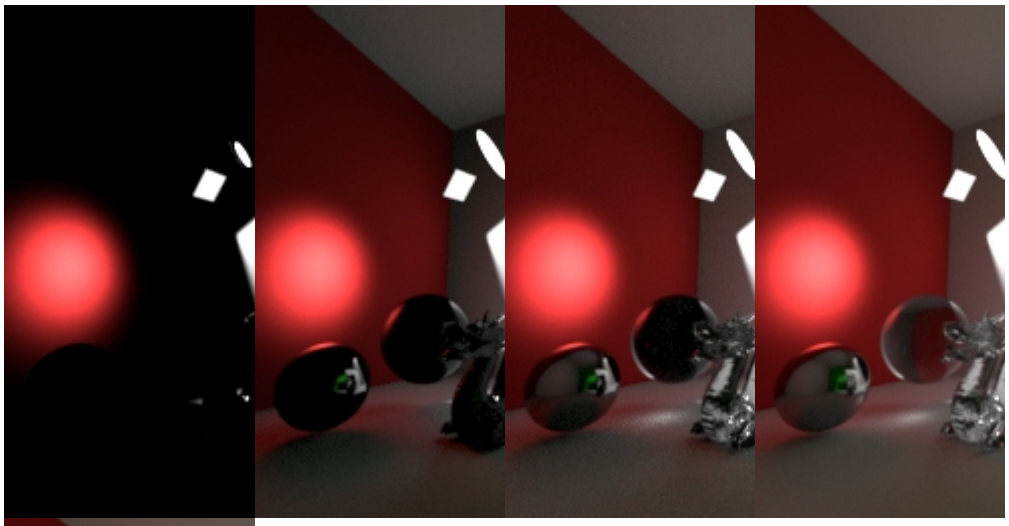
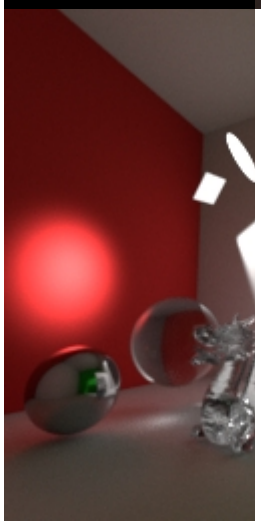
For more technical information on the advanced features, you can take a look at these research papers:

- Thomas Müller, Markus Gross, and Jan Novák. Practical Path Guiding for Efficient Light-Transport Simulation . Computer Graphics Forum (Proceedings of EGSR 2017), vol. 36, no. 4
- Johannes Hanika, Marc Droske and Luca Fascione. Manifold Next Event Estimation. Eurographics Symposium on Rendering 2015, vol. 34, no. 4

Basic Parameters

Parameter	Description

traceLightPaths	This enables bidirectional path tracing modes. PxrUnified has 4 integration modes, based on how light paths are sampled. More options are available and some are invalid based on selections. Similar to PxrVCM selections as noted below. More details on settings to come as the integrator is finalized.
	• **Off**
	  ◦ Unidirectional path tracing, like PxrPathTracer
	• **On**
	  ◦ Bidirectional path tracing, like PxrVCM with connect on and merge off. Indirect Guiding is disabled.
	• **Photons Only**
	  ◦ Progressive photon mapping, like PxrVCM with connect off and merge on. Indirect Guiding is disabled. Photo Controls are then exposed.
	• **UPS/VCM**
	  ◦ Bidirectional path tracing with progressive photon mapping, like PxrVCM with connect on and merge on.

maxPathLength	Controls the absolute upper bound on the maximum ray depth. For example, a value of 0 for this parameter will allow direct illumination only, while a value of 4 will permit up to 3 bounces of global illumination. Default value is 8. Ray *Depth per diffuse or specular interaction is ignored unless useTraceDepth is enabled.* •  • 
useTraceDepth	When active, the integrator will follow the trace depth assigned to objects in the scene, this is often used as an optimization in render to limited the number of types of bounces to what's necessary visually. See Trace Depth.

maxRayDistance	This controls the distance (in scene units) an *indirect* ray will travel before being terminated. If your scene scale is such that objects far from one another would not visually interact indirectly you can use this to optimize the scene. Note that rays traveling through windows and other objects are indirect rays and may be terminated too early and disappear from glass refraction. (Imagine a render where you view a mountain range through your living room window, be sure these rays reach far enough.)
catchAllLights	On the last hit, evaluate the lights without shadowing, the default is off (0.0). If you find refraction goes dark, setting this slider to 1.0 adds the full contribution of the light samples. Turning this on may brighten your scene.
emissionMultiplier	Multiplies all emissive and "glow" properties set on materials and lights.
accumOpacity	Controls whether or not the path tracer will keep track of accumulated opacity along the path. The default is off. If you plan to render refractive objects and care about alpha values for compositing, you might want to turn this on, car windows for example.
specularCurvatureFilter	Filter the roughness for specular interactions, setting it to 0.0 is no filtering or "ground truth". The default is designed to reduce noise.
numLightSamples	Controls the number of light samples for direct illumination per camera hit point. The default is 1. Small numbers may improve performance of interactive and progressive updates. Larger numbers may appear to converge better while being slower to update.
numBxdfSamples	Controls the number of Bxdf samples for direct illumination per camera hit point. The default is 1. Small numbers may improve performance of interactive and progressive updates. Larger numbers may appear to converge better while being slower to update.
numIndirectSamples	Operates as a toggle of off (0, or no indirect and on (1)
sssOversampling	Controls the number of subsurface rays to spawn per camera hit point. The default is 8.
allowMultilobeIndirect	Allows the re-use of samples during hair rendering to speed up convergence. This may alter the resulting LPEs.
rouletteDepth	Controls the ray depth to begin performing Russian Roulette. Russian Roulette may terminate rays after this depth especially if they contribute little to the result. The default is 1.
rouletteThreshold	Controls the path throughput threshold below which to perform Russian Roulette. This controls at which point the contribution of the ray is unimportant enough to terminate the ray. The default is 0.2.
directClamp	Sets the amount where values will be clamped, for direct lighting the default is high to preserve the dynamic range of a scene's lighting. Lowering it will reduce noise like fireflies at the cost of dynamic range. The default is 1000000.0
indirectClamp	Sets the amount where values will be clamped, for indirect lighting the default is low, this avoids noise in secondary effects that have already absorbed some energy anyway. Lowering it will reduce noise but also darken indirect effects. The default is 8.0
causticClamp	Sets the amount where values will be clamped, for caustics the default is high to preserve the light patterns that result from light focusing off of or through objects. Lowering it will reduce noise like fireflies at the cost of dynamic range. The default is 1000000.0

Manifold Walk (Not Currently Available With BDPT modes)

Parameter	Description
manifoldWalk	This enables or disables the calculation for the manifold walk. Note that there is more needed to setup the scene and you can find these settings in, Lights, PxrSurface, and the integrator that work together.
maxIterations	This is the maximum number of iterations the walk will perform before giving up and terminating
maxInterfaces	This is the maximum expected number of transmissive interfaces the walk is expected to encounter on its way to the light. Values too low may prevent it from exiting and finding the light source.
walkThreshold	This is the threashold used to control the convergence of the manifold walk. Higher values may generate noise.
enableVolumeCaustics	This allows for volumetric caustics in the medium (if one is present). The object enclosing the medium must be part of a specific trace set: mw_volume_include_subset This is an expensive technique.

RIS Indirect Guiding

Parameter	Description
risPathGuiding	When set to 1, this enables indirect path guiding. While rendering is underway, this technique learns the distribution of indirect radiance in the scene and combines this information, using multiple importance sampling, with Bxdf sampling to improve convergence time. It's particularly effective for glossy and diffuse surfaces, and in areas where indirect light dominates.Path Guiding

PPG Indirect Guiding

Parameter	Description
indirectTrainingSamples	This is the number of samples used to "learn" the scene's light distribution. This number should never be more than the minSample set by the hider and no more than 10 to 20% of the maxSamples. 64 should be the minimum (meaning your minSamples should be at least 64 or if set to 0 it's the square root of your maxSamples and possibly too low)
indirectSpatialBlurRadius	This blurs boundaries of the spatial tree, this only affects the sampling, it's not a blurring parameter for the resulting samples. Use this if you see discontinuities caused by the grid structure.
indirectDirectionalBlurRadius	This blurs the directions given in the spatial tree, this only affects the sampling, it's not a blurring parameter for the resulting samples. Use this if you see discontinuities caused by the grid structure.
indirectOversampling	Over sample the lookup tree for improved results. Usually a very subtle effect.

Photons (Enabled with Trace Light Paths set to Photons or UPS/VCM modes)

Parameter	Description
photonEstimationRadius	A search radius in scene units for photon lookup optimization, typically the default is fine unless you want a different preview as it converges
photonEstimationNumber	The number of photons to use for estimation inside the above search radius
photonVisibilityRod	A targeting shape for photons, on or off
photonVisibilityRodDirectProb	How tightly to target photons to the Rod box specified below. Values approaching 1.0 will begin to bias the image
photonVisibilityRodDirectMin	The minimum bounds of the photon targeting area
photonVisibilityRodDirectMax	The maxiumum bounds of the photon targeting area
photonAdaptive	Based on the paper: *"Adaptive Progressive Photon Mapping"*

Diagnostic

Parameter	Description
suppressNaNs	This invalidates individual samples that return "not a number" before they get splatted to the image. There is overhead to perform this check and is off by default.
enableShadingTimers	(developer option) Enables timers that will collect shading cost of shaders group by the profileGroup user attribute
enableSampleTimers	(developer option) Enables the timers that track pixel sample iteration cost

Standard AOVs

On top of regular LPE-based AOVs, this integrator outputs a number of standard AOVs typically used by compositors.

Declaration	Contents	Channels

color __Pworld	P in world-space	__Pworld.r : x component __Pworld.g : y component __Pworld.b : z component
color __Nworld	Nn in world-space	__Nworld.r : x component __Nworld.g : y component __Nworld.b : z component
color __depth	Multi-purpose AOV	__depth.r : depth from camera in world-space __depth.g : height in world-space __depth.b : geometric facing ratio : abs(Nn.V)
color __st	Texture coords	__st.x : s __st.y : t __st.z : 0.0
color __Pref	Reference Position primvar (if available)	__Pref.r : x component __Pref.g : y component __Pref.b : z component
color __Nref	Reference Normal primvar (if available)	__Nref.r : x component __Nref.g : y component __Nref.b : z component
color __WPref	Reference World Position primvar (if available)	__WPref.r : x component __WPref.g : y component __WPref.b : z component
color __WNref	Reference World Normal primvar (if available)	__WNref.r : x component __WNref.g : y component __WNref.b : z component