RenderMan 23.2

Welcome to RenderMan 23.2!

Welcome to RenderMan 23.2. This release introduces improvements to the previous RenderMan in very significant ways.

Please dive right into the release notes below for more detailed information on the latest version of your favorite renderer!

New Features in 23.2

• BxDFs have a new parameter to improve shadowing when using a bump or normal map, the new behavior is off by default to preserve the old look. The improvement results in smoother, more natural shadowing





Left is the improved method, Right is the old behavior (default in 23)

Fixes

- · Fixed a bug where displacement and volume shaders initialization would happen too late (i.e. after displacement and/or dicing)
- Added PxrSurface "glassRefractionRoughness" parameter also to PxrLayerMixer OSL shader
- Prman would crash with the PxrProjector node if a coordinate system was not provided
- Avoid deadlock when user plug-ins throw exceptions. Please note that all user plug-ins are expected to catch their own exceptions.
- · Fix intermittent baking crash with subdivision meshes
- · Fixed a bug where uniform primvars of Subdiv and Poly meshes could return uninitialzed derivatives
- Fix memory leak of framebuffer data when baking
- Fixed a bug that could cause unrelated lights across a scene to randomly flicker when using a GOBO on any light
- · Fixed artefacts in PxrTangentField. The is a new "int backwardCompatible" parameter should you prefer the old behavior
- Fix color array, matrix, and matrix array parameters with the python bindings
- · Fixed a bug where scenes with very high light counts could sometimes get stuck for a long period of time during initial scene processing
- Fix incorrect varying quadric privmar length in scene graph API
- PxrNormalMap, PxrBump: Fixed a rare crash when a misconfigured connected pattern fails to deliver the surface normal for bump layering
- · We now invalidate curves clusters (if needed) after group flattening, before building the reference instances to avoid a crash in curve procedurals

Miscellaneous Changes

- · Robustified replacing Pixar formatted textures on disk during live renders
- The precision of OSL transforms from current to arbitrary spaces has been improved
- Threading performance for the tessellation of dense displaced poly and subdiv meshes has been improved

Known Limitations

Interactive/Live Rendering Limitations

- Bucket order or size cannot be changed during live rendering
- Changes to Presence do not update when using the opacity cache option
- Motion Blur will disappear during interactive rendering with scene changes
- Objects are not re-diced during interactive camera edits
- · Mesh lights cannot be interchanged as geometry without restart.

RenderMan Pro Server

- PxrUnified integrator is currently experimental as it does not yet support all the standard rendering features.
- Meshlights cannot be instanced

- · Load on demand procedurals are not supported anymore, all procedurals are now loaded immediately
- We do not read point data from OpenVDB files
- PxrSurface back diffuse color is not output to the albedo color AOV
- Analytical lights placed inside volumes may yield artifacts when made visible to the camera. As a work around, the light camera visibility should be turned off, and a geometry with a similar shape should be used (visible to camera, invisible to transmission and indirect rays), with the proper emissive bxdf.
- Using the '.' character in the handle for an OSL shader could cause unpredictable results during re-rendering.
- Per-Instance baking is not supported, only the reference instance.
- 3d baking: no direct bake-to-ptex support.
- PxrBakePointCloud cannot directly render ptex.
- · Sample/Display filter plug-ins do not have access to lighting services for light dependent effects, e.g. lens flare.
- Adding new mesh light on existing geometry during IPR results in double geometry.
- Motion blurred polygons do not motion blur normals when deformed. Use Subdivision meshes instead.
- When attempting to access an array primvar, you must first check the size of the array primvar and allocate the appropriate space. Not doing so may lead to a crash.
- Points and curves cannot be used as geometric lights.
- Deformation motion blurred volumes don't currently work with densityFloatPrimVar or densityColorPrimVar. You will need to use a PxrPrimVar node connected to densityFloat and densityColor instead.