RenderMan 23.3

Welcome to RenderMan 23.3!

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 23.3

- macOS 10.15 Catalina is now supported
- We now support Illumination/Integrator baking as described here in the documentation: Baking Illumination

Fixes

> ⚠ Note PxrProjector has changed default behavior. The invertT parameter is now OFF by default, in previous releases it was ON by default.

- PxrProjectionLayer: The missingAlpha value was ignored when a texture didn't have an alpha channel. The missingAlpha is a single float value and can not be textured
- The filter and blur parameters for the PxrMultiTexture pattern now work as expected
- Fixed a rare crash in PxrRamp, when a node is connected to the splineMap parameter but fails to deliver valid data
- A bug causing vector based motion blur not to work with alembic archives has been addressed
- The PxrTee inputRGB port now accepts incoming connections of type color, vector, point, or normal
- Fixed a potential memory overflow in RixRNG
- Fixed a problem with "it" not loading custom commands when the system had a font without a family name
- Improved robustness of the divide operation in OSL shaders
- The "int reentrant" [0] attribute was not correctly ensuring non-thread safe procedurals ran on a single thread. Now the attribute works as expected
- Fixed a multi-threading bug that would lead to a crash when rendering Alembic procedurals
- Setting the post-checkpoint command to an empty string no longer causes the renderer to abort after emitting the first checkpoint
- Fixed a bug that caused curves to darken incorrectly when varying width was applied along the length of the curve
- Fixed a camera transform and projection bug with geometry procedural plugins; they now correctly transform to screen, raster, and NDC space
- Fixed a bug that prevented the dPdtime and dPcameradtime AOVs from outputting the correct values
- PxrSurface manifold walk NaNs/Infs/near-infinite values when refractionGain is 0.0 have been fixed
- Fixed a bug causing a crash if primvar "u" or "v" were added to a subdivision surface primitive
- An issue that prevented OSL from executing correctly on 10th generation Intel chipsets has been resolved
- Fixed a bug that caused the adaptive sampler to oversample when variance channels were provided for AOVs like "N"
- PxrProjector: Improved near/far clipping: parallel to the film plane rather than radial. Fixed filmFit in Fill and Overscan mode
- Fixed bugs relating to the checkpointing and restoration of Cryptomatte data

Miscellaneous Changes

- "it" will no longer automatically try to apply the nVidia Denoiser to all images when enabled. Images now are categorized into color and non-color when they are created, and only color ones are denoised. The RGB indicator on the Pixel Readout now changes to XYZ if "it" does not consider the AOV to be color. The user can use ImageToggle Is Color (Shift+A) to override this setting on an image
- The darkfalloff parameter to control adaptive sampling was brought back in 23.3. This can now be used in combination with the exposurebracket parameter to finely tune the amount of adaptive sampling for multiple levels of exposure while also improving performance in areas where the samples will not make a significant perceptual difference.
- Added a new geometric control: int curve:widthaffectscurvature [0/1]. The default is 1, meaning that the calculation of curvature in round curves geometry will behave exactly as before. Setting the attribute to 0 will make curvature only account for the analytic curvature along the curve itself and ignore any high curvature due to thin curve widths. Setting the attribute to 0 is probably the most artist friendly way to render curves with BxDFs such as PxrSurface and also in the case where curvature is used to model shading effects in patterns (for example, to vary colors based on region of hair that may be smooth or frizzy).
- When attempting to install our software on a Windows platform that did not meet minimum hardware requirements, it would fail with no valid indication of why. Now attempting to run the installer on Windows 10 results in an error dialog if the hardware specs are too low.
- The denoiser now displays an error message and exits if the input images' data windows do not match, instead of crashing.
- Optimized the deletion of many instances in the scene graph API.
- Bucket order can now be changed during live rendering.
- Added the capability for PxrLayerSurface to support the new shadow bump terminator parameter present in PxrSurface.
- PxrProjector added overscan and orthographic camera support.

Known Limitations

Interactive/Live Rendering Limitations

- Bucket 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 a restart

RenderMan Pro Server

- When baking illumination sample filters and display filters are not currently supported and may lead to a crash.
- 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.