RenderMan 23.4

Welcome to RenderMan 23.4!

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 23.4

- Checkpointing can now take a sequence of intervals that it steps through with each checkpoint, this allows output at different specified timings. Please see the documentation link
- A new section, 'channelrenamings' has been added to the denoiser configuration file. This allows the Denoiser to internally rename channels before processing, providing greater flexibility on the input images it can handle
- Added new option to disable per-instance geometric light attributes for compatibility reasons: Option "future" "int enableGeometricLightInstanceAttributes" [1] (default)
- OSL shaders in RenderMan can utilize substitution of <udim>, <u>, <v>, <U>, <V> and <primstr:varname> token substitutions in filenames for texture() and gettextureinfo() shadeops
- Add a new backwards-compatible adaptivemetric hider option for adaptive sampling in RenderMan 23: "variance-v22". This switches back to the old relative variance calculation from RenderMan 22

Fixes

- Fixed various bugs relating to the checkpointing and restoration of Cryptomattes
- Fix baking crash when dicing bilinear subdivision mesh
- The baking of subsurface scattering results has been improved
    - When baking illumination and subsurface scattering in world space, explicitly set: Attribute “dice” “string strategy” “worlddistance”
- Avoid a crash with invalid scene display and display channel configurations
- A bug that caused ray depth values retrieved from a pattern node to produce inconsistent results has been fixed
- Fix deprecated python SGManager.ScopedScene utility class destructor
- Fixed a problem where dielectrics with intersection priority and interior volumes may have produced darker shadows than expected
- RLF/Alembic: the renderer would refuse to render scenes containing RLF data created with an older protocol version
- Fixed a crash with cached opacity
- Fixed a bug that could lead to sporadic crashes when using OSL trace()
- Fixed a bug that affected cross-frame denoising of AOVs
- We now correctly combine the two shadow exclusion subsets defined by Attribute "trace" "string shadowexcludesubset" and the "string shadowExcludeSubset" parameter on Light shaders that support this functionality
- A bug has been fixed that prevented packed alembic archives of points with velocity and changing topology from being rendered correctly
- Fixed a bug where curves were producing correct dPdtime and dPcameradtime AOVs *only* if curves had deformation motion

Miscellaneous Changes

- The dPdtime AOV no longer contains camera motion. Users should use the dPcameradtime AOV to output a motion AOV with camera motion
- Checkpoint commands can now take a new %E token which will expand into the elapsed time in a human-readable h:mm:ss form. This is convenient for logging with echo
- Checkpointing with a sequence of intervals will now resume the sequence from where it left off after recovery. Recoveries from checkpoints by prior RenderMan versions will still
  restart the sequence from the beginning
- Support live render edits of screen window as an option or camera property
- Avoid emitting error R09065 when adaptall is enabled

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.