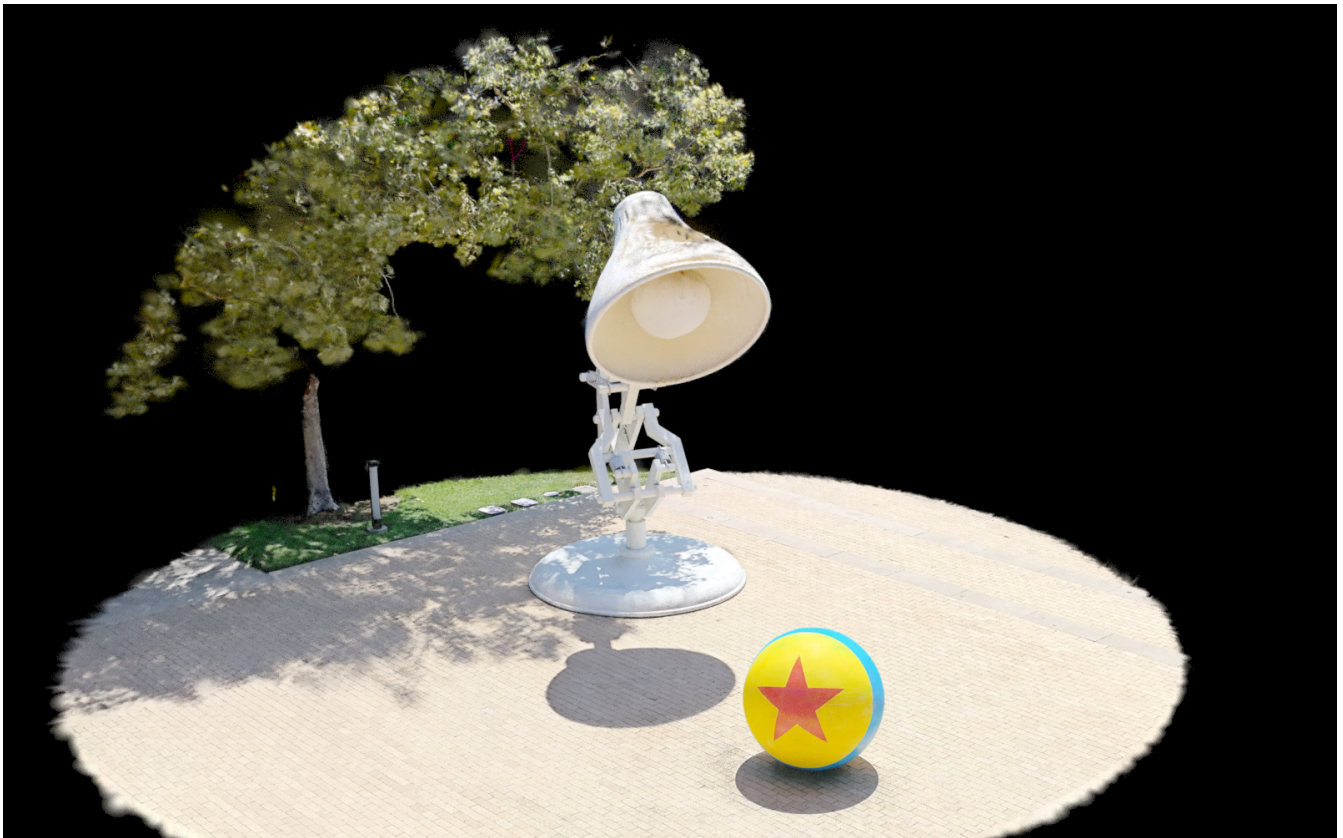


Gaussian Splats

Example Setup with Houdini and RenderMan 28



RenderMan 28 supports Gaussian splats through their implementation in USD. A scan becomes a native USD particle field, carrying its positions, size, orientation, opacity, and captured color into the scene. If your scan starts as a PLY file, convert it to USD first. A basic conversion script is included.

This example uses a lamp, a ball, and a small piece of the plaza. Load the scan, flip it upright, and render. Then use a few ordinary Houdini nodes to push the points around. The lighting and shadows are already part of the capture; this example does not relight them.

Open the scene

Unzip the folder and open **Gaussian_Splats_v2.hip**. In Solaris, go to **/stage**, display **VIEW_FINAL_SCENE**, choose **/cameras/main**, and select **RenderMan XPU - CPU**. Select **STYLIZE_YOUR_SPLAT** once to use its controls while the complete scene stays visible.

Thanks to Kosmo for the hardware that created this scan. www.kosmoworld.com

Load. Flip. Render.

Load Splat is a Sublayer LOP. Its path is **\$HIP/pixar-lamp-and-ball.usdc**, so the HIP and USD can move together. The supplied USD is ready to render; conversion is only needed when bringing in another PLY scan.

Flip is a Transform LOP targeting **/scan**. Rotate X is **180 degrees**. Translation is **(10.4, 1.8, -3.2)**, placing this scan near the origin. These settings fit this capture. Another scan may need a different orientation, placement, or scale.

The scene at a glance

Stay at /stage to adjust the look and see the final scene.

LOAD_SPLAT Sublayer	\$HIP/pixar-lamp-and-ball.usdc
FLIP Transform	Rotate X: 180 degrees
STYLIZE_YOUR_SPLAT SOP Modify	Choose a look and adjust its controls here.
FINAL_LOOK Camera	The final view: /cameras/main
RENDERMAN RenderMan Render Settings	1600 x 1000 pixels / 64 maximum samples
VIEW_FINAL_SCENE Null	Keep the display flag here while you stylize.

RENDER_IMAGE > Render to Disk saves an EXR in the render folder.

Final Look creates the camera. RenderMan sets the image size and points to that camera. Select **RENDER_IMAGE** and press **Render to Disk** to save an EXR in **\$HIP/render**.

Keep the final scene in view

The controls are at the top level. Double-clicking Stylize Your Splat enters its internal geometry network, where the preview looks different from a RenderMan splat render. Go back to /stage to use the finished view. The internal nodes are there if you want to explore how it works.

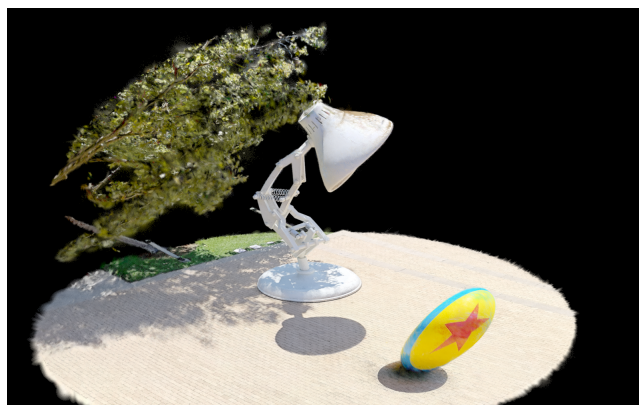
Stylize Your Splat

Choose a Look on **Stylize Your Splat**. Only the controls for that choice are shown. Each choice starts from the same scan, and **As Captured** returns to the original appearance. Keep the display flag on **VIEW_FINAL_SCENE** and stay in Solaris while adjusting.



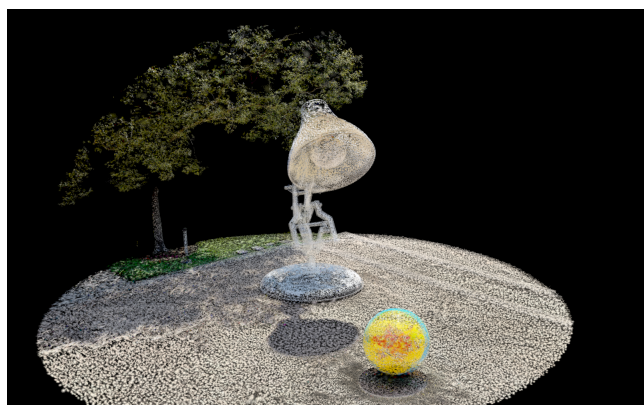
Wobble

Attribute Noise adds movement to the points. Wobble Amount starts at 0.24; Wobble Size starts at 0.65. Turn the amount down for a quieter result.



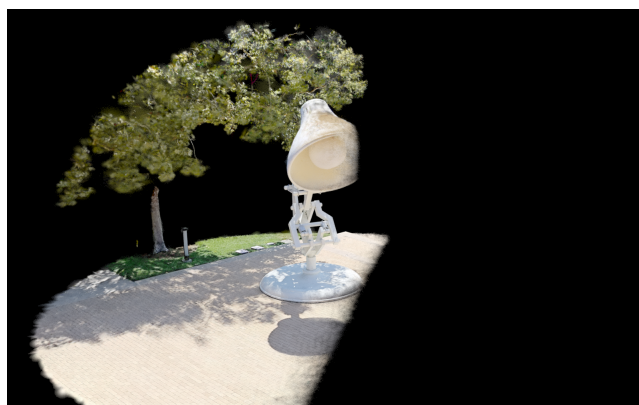
Twist

A Bend SOP turns the points through 90 degrees along Y. Adjust Twist to wind it up or unwind it. The original colors and captured shadows move with the points.



Tiny Splats

Attribute Adjust Vector multiplies splat scale by 0.3. The same number of points makes smaller marks, revealing more space between them. Bring Splat Size toward 1 to fill them back in.



Fade From Side

Attribute Adjust Float reduces opacity across the scene, along Z. Fade Start and Fade End bracket the transition; the Side Fade ramp shapes it. The default range is -2 to 2, giving a clear side-to-side fade from the final camera.

Details

The close-ups make the changes easier to see. Wobble moves the splat centers. Tiny Splats changes their coverage. Both can expose gaps in the capture, and both preserve its color. Try one small change at a time.



As Captured

The captured color and surface detail.



Wobble

The same view with point noise.



Tiny Splats

Smaller marks reveal the individual splats.

Five looks. One scan. No added lighting.

Bring in another scan

The included script reads standard, uncompressed 3D Gaussian splat PLY files and writes a native **ParticleField3DGaussianSplat** at **/scan**. It preserves spherical-harmonic color data, converts stored log scales and opacity values, and leaves the input file unchanged. It refuses to overwrite an existing output.

PLY to USD

Use ordinary **64-bit Python 3.9-3.14**. No Houdini installation is needed to convert. From the folder containing the script, run setup once, then convert:

```
py Convert_PLY_to_USD.py --setup  
py Convert_PLY_to_USD.py scan.ply scan.usdc
```

On Mac or Linux, use **python3** instead of **py**. Setup needs internet access and installs the libraries in a private folder beside the script. Later conversions use it automatically.

Load the resulting scan.usdc with a Sublayer LOP. Recheck Flip and Final Look for the new scan. The converter accepts standard, uncompressed 3DGS PLY with SH degrees 0-3. See Conversion.txt for supported systems and formats.

A few details

Tested with **Houdini 22.0.429** and **RenderMan 28.0b1, XPU CPU** on Windows. Use matching RenderMan for Houdini with Gaussian-splat support and a working render license.

The supplied scan contains **1,256,332 splats**. All five looks retain that count. The HIP uses standard nodes and channel references, with no custom code nodes or session scripts. The standalone converter is only used before loading a new PLY scan.

The HIP saves a built-in ACES 1.3 OCIO configuration, so no external color file is needed. Article images use a linear-to-sRGB display conversion. Camera: 45 mm, 1600 x 1000 pixels. Maximum samples: 64.

Thanks to Kosmo for the hardware that created this scan. www.kosmoworld.com