Rendering

Houdini exposes all the useful features used for render frames and sequences. Below are the topics covered as well as some descriptions of features you'll see discussed.

Image from *Coco*

There are many parts to rendering your completed scene. From handling quality to choosing the appropriate outputs for later compositing, this section handles all of those topics.

Quality

To understand how noise removal and anti-aliasing works, we describe Sampling and Filtering for users. This is where the bulk of your tuning may happen and it's important to understand the balance between quality and performance.

Trace Depth

Controlling trace depth can alter the look and performance of your scene. Different integrators also handle these settings differently. RenderMan makes tuning of these parameters easy to achieve your required look within your resource constraints.

Interactive Rendering and Checkpointing

Progressive/Interactive Rendering makes easy work of tuning your scene materials and lights. Checkpointing and Recovery takes this progressive feature and extends it to final renders where you can create save points in your renders as they refine. You can use these for faster approval of incomplete frames and continue where you left off without wasting precious CPU cycles!

Diagnostics

RenderMan diagnostics helps you visualize your scene performance in an easy to read display. Sometimes it may not be clear where your scene is spending the most time or resources, but our diagnostic output will help you pinpoint even the most obscure performance information.

Holdouts and Outputs

Holdout workflow is essential for integration of CG objects into live action plates. In some applications this is as simple as a button click! To improve compositing integration and even fully rendered shots, RenderMan makes use of powerful outputs to make adjustments in compositing easier and powerful with ultimate flexibility.

Baking

Reusing certain data in rendering can simplify asset management as well as improve rendering efficiency. Baking provides ways to bake to a 2D or point cloud output.

Nested Dielectrics

When rendering surfaces that interact with one another, especially transparent/refractive ones, it's important to understand the Intersect Priority settings. Intersect priority allows users to simply add attributes to get the correct refractions without tedious modeling to solve the problem.
Link