

CS 148 Final Project Fall 2018

Petar Penkov ppenkov@stanford.edu

My scene is a cozy guitar room that has three electric guitars mounted on the wall with hooks, an acoustic guitar propped against large floor speakers, and an armchair with a side table that can be seen only through the mirror. The scene is illuminated with an area light, positioned on the ceiling, and with three spotlights meant to shift the focus towards the mounted guitars. With the exception of the guitars that are meant to draw the viewer's attention (hence the contrast), I tried to make use of warmer colors to achieve the "cozy room" effect.

Assets

Online Assets

I found all assets listed in the section and their respective textures using Google, TurboSquid, and Free3D. Here is a list of the models found online:

- Electric Guitars - unmodified
- Armchair - unmodified
- Outlet - unmodified
- Speakers - I had to fixup the texture of the speakers as there was an artifact, resembling a chip
- Cups - unmodified
- Table - unmodified
- Plug of the cable - I removed the actual cable so I can model it as described below
- Wall hooks - I extracted the model for the wall hooks from the tips of a nice coat hanger and reduced the proportions of the sphere relative to the cylinder

Created Assets

- Walls / Ceiling / Floor - all of these are simple planes
- Mirror - both the frame and the mirror were created in Maya
- Cable - to get a realistic custom look, I modeled the curvature as an EP curve in Maya, and then extruded the volumetric model using a cylinder of the right size.

Technical Contributions

Parallel Ray tracing

I did a naive parallelization of the main ray tracing loop, using the thread library in C++. I managed to utilize all 4 cores on my computer for nearly all of the rendering process by running the raytracer on 8 threads.

Spotlights

As an implementation of spotlights was missing from the provided ray tracer, I had to implement them using the formulas from assignment 4.

Photon Gathering

I built on top of assignment 8 to complete the photon mapping implementation. I followed Jensen's guide to also account for the photon color during the Russian Roulette phase and to determine the indirect lighting component of the final color from the neighbors. Unlike Jensen's guide, I used a constant radius search rather than nearest neighbors. I also found that using the Manhattan distance gave me a reasonable approximation and did not change the kd-tree implementation. Finally, I added a cone filter which made the final color more accurate.