

CS 148 Final Project

Ken Hong | kenhong@stanford.edu



1500 x 2160

Variants

Variant A	Variant B
	
500 x 720 Different camera angle, no rendering effects.	500 x 720 50%-grey diffuse texture, no rendering effects.

Note: Variant B reveals the presence of shadow acne in the background plane — I decided to ignore this as my implementation of the background plane is not affected by light rays.

Description

Concept

I wanted to create a scene that is able to evoke the image of a Japanese alleyway, and incorporate elements that are both modern and slightly retro-looking. I tried to make sure that the models that I found looked somewhat realistic, as I was going for a realistic aesthetic with this scene.

I set the angle of view of the camera to be wider than default so that I could introduce slight distortion into the image, accentuating the depth of the alley. The camera was also rotated along the roll axis, which was done to introduce asymmetry into the image for visual interest.

Implementation

The composition of the scene was done in Autodesk Maya, and preliminary lighting was added to help me visualize the final lighting setup. The scene was then exported and imported into the ray-tracer, where further tuning of the lighting and textures was done.

The final image was rendered at 64 samples per pixel and took 52 hours to complete on a Google Cloud Compute Engine instance with 80 cores at 2.8 GHz. The rendering time was longer than expected, and could be due to the large number of lights in the scene coupled with the added rendering effects.

Technical Contributions

Summary of Modifications by File

`PerspectiveCamera.cpp` Depth of field
`PhotonMappingRenderer.cpp` Global Illumination
`BackwardRenderer.cpp` Emission
`Material.cpp`, `BlinnPhongMaterial.cpp` Background plane
`ImageWriter.cpp` Post-Processing Effects
`RayTracer.cpp` Performance

Depth of Field

To achieve a subtle depth of field effect, I used a simulated aperture of $r = 0.075$ units to send out new rays which allowed me to achieve the look that I wanted. A byproduct of the depth of field effect was that there was much more undesirable noise in the not-in-focus parts of the image, and required the image be rendered at a much higher sample count to be less noticeable.

Global Illumination

I made an attempt to incorporate global illumination using photon mapping, although I am not sure how much the effect contributed to the final image. I did not observe much change except very close to the lights (e.g. the glow on the surface of the lights).

Emission

In order to have emissive materials in the scene, I added a very simple modification in which emissive materials have their final colors (after BRDF calculation) boosted by some constant value. I tweaked this constant for each emissive light (the four signs and the lamp) until I had the desired effect. I used point lights to account for the actual light emissions from these objects, which did not work as well as I had hoped.

Post-Processing Effects

To add aesthetic effect, I implemented two effects after the main loop of the ray-tracer. The first of these was a vignette effect as I wanted to dim the edges of the images as most real camera lenses would do. The second was a noise effect, which I wanted to add to prevent some materials from being too 'perfect' (no variation in color), as well as to simulate the appearance of film grain. The vignette effect applied to a pixel was calculated to be proportionate to the distance of the pixel to the center of the image. The noise was randomly-generated and clamped to a certain threshold.

Performance

I implemented multi-threading in the main loop with OpenMP. I chose to parallelize only the outer loop, which led to performance bottlenecks towards the end of renders as specific threads would remain idle before others had finished if there are significant differences in the render times of different rows of the image. Also, as the render time was so long and I did not want to lose progress, I added an output function to the loop that records the value of each pixel in an intermediate file once it is finished, so in the case my renderer or computer crashes I can start from where I had left off. It also allowed me to inspect the image before the render had finished if necessary.

Assets and Sources

Legend	Table of Assets																																										
	<table><tr><th>Code</th><th>Description</th><th>Creator and Source</th></tr><tr><td>1</td><td>Signs (1)</td><td>Self, modeled in Autodesk Maya</td></tr><tr><td>2</td><td>Sign (2)</td><td>Self, modeled in Autodesk Maya</td></tr><tr><td>3</td><td>Billboard</td><td>Self, modeled in Autodesk Maya</td></tr><tr><td>4</td><td>Road</td><td>Mellowmesher, Turbosquid</td></tr><tr><td>5</td><td>Road Text</td><td>Self, modeled in Autodesk Maya</td></tr><tr><td>6</td><td>Wires</td><td>Self, modeled in Autodesk Maya</td></tr><tr><td>7</td><td>Air Conditioner</td><td>Roozbeh Edjbari, Turbosquid</td></tr><tr><td>8</td><td>Lights</td><td>theernestobecerra, Free3D *slightly modified model</td></tr><tr><td>9</td><td>Houses</td><td>sumutf, Turbosquid *slightly modified model</td></tr><tr><td>10</td><td>Lamp</td><td>android99, Turbosquid *slightly modified texture</td></tr><tr><td>11</td><td>Person</td><td>NumikDigital, CGTrader</td></tr><tr><td>12</td><td>Building</td><td>Dimensional Dreams, Turbosquid *slightly modified texture</td></tr><tr><td>13</td><td>Sky</td><td>Shutterstock</td></tr></table>	Code	Description	Creator and Source	1	Signs (1)	Self, modeled in Autodesk Maya	2	Sign (2)	Self, modeled in Autodesk Maya	3	Billboard	Self, modeled in Autodesk Maya	4	Road	Mellowmesher, Turbosquid	5	Road Text	Self, modeled in Autodesk Maya	6	Wires	Self, modeled in Autodesk Maya	7	Air Conditioner	Roozbeh Edjbari, Turbosquid	8	Lights	theernestobecerra, Free3D *slightly modified model	9	Houses	sumutf, Turbosquid *slightly modified model	10	Lamp	android99, Turbosquid *slightly modified texture	11	Person	NumikDigital, CGTrader	12	Building	Dimensional Dreams, Turbosquid *slightly modified texture	13	Sky	Shutterstock
Code	Description	Creator and Source																																									
1	Signs (1)	Self, modeled in Autodesk Maya																																									
2	Sign (2)	Self, modeled in Autodesk Maya																																									
3	Billboard	Self, modeled in Autodesk Maya																																									
4	Road	Mellowmesher, Turbosquid																																									
5	Road Text	Self, modeled in Autodesk Maya																																									
6	Wires	Self, modeled in Autodesk Maya																																									
7	Air Conditioner	Roozbeh Edjbari, Turbosquid																																									
8	Lights	theernestobecerra, Free3D *slightly modified model																																									
9	Houses	sumutf, Turbosquid *slightly modified model																																									
10	Lamp	android99, Turbosquid *slightly modified texture																																									
11	Person	NumikDigital, CGTrader																																									
12	Building	Dimensional Dreams, Turbosquid *slightly modified texture																																									
13	Sky	Shutterstock																																									