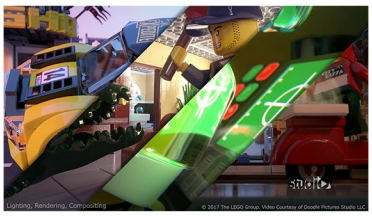
DavidDuVoisin

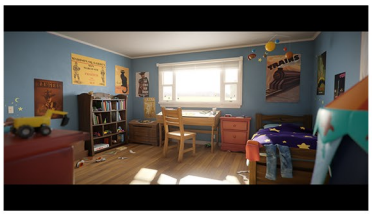
Lighting Technical Director

davidduvoisin.com | daduvo11@gmail.com | 1(808)430-9308



Lego City Studio
Lighting, Rendering, Comp

This is a small selection of my favorite projects from our Lego City Studio series. The brick-built environments, while heavy on memory, are a ton of fun to work with. For the night sequence light rig I wanted to have lights in each of the 30+ lamps. To make this setup easy, I wrote a basic script to find each lamp head and setup the lights and geometry blockers for it. This script also made making changes easy when I had to add/remove lights/blockers. Sequence and shot lighting and rendering were handled in Redshift for Maya. Compositing was done in Blackmagic Fusion.

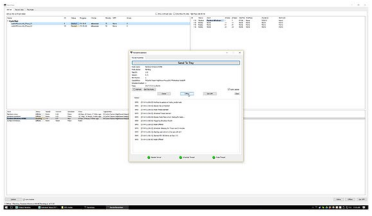


Sebastian's Room
Shading, Lighting, Rendering, Comp

This project started as a RenderMan 18 project for a short film I worked on in college. The textures were set up for a traditional blinn/phong shading setup so I converted them to a PBR workflow using Substance B2M, Substance Painter, and Photoshop. I shaded, lit, and rendered it using Redshift for Maya and composited it in Blackmagic Fusion. This was a great project to learn more about shading in Redshift, which I was much less familiar with at the offset.

Lego Star Wars
Lighting, Rendering, Comp

The Lego Star Wars projects are interesting because they're usually using environments and lighting inspired by the movies. Sampling becomes tricky on some of the interiors due to the amount of rough reflective materials. The explosions were particularly tough on the final sequence because sampling the reflections from the volumes was incredibly slow and noisy. To solve this, I converted the heat channel from each of the volumes to a mesh and used the meshes as mesh lights. For these projects I oversaw shot lighting, rendering, and compositing using the same software



Hydra Renderfarm
Programming

Project Hydra started as a basic render farm project from Cogswell College. I forked it and built off of it significantly to it to make it into a full-featured render farm software. It is written mostly in Python using PyQt for the frontend and MySQL for the backend. It features most modern render farm features such as queue prioritization, crash/error handling, logging, node scheduling, multiple job types/softwares, etc. Doing this project I learned a ton about Python and it has made me very confident in my Python development skills. It still has a bit to go but with any luck we'll be using this in production some day.