BOOK COVER TEMPLATE

Table of Contents

Introduction 1.1

Publishing on Github Pages 1.2

Chapters 1.3

- Chapter 2: A placeholder page 1.3.1

Blender resources 1.4

- Blender Plugins 1.4.1
- Rendering Resources 1.4.2

3D Digital Studio: Edition Two

In this book, I cover the theory, practice, and purpose related to the production of 3D animation and beyond. Included are projects, exercises, lectures, and tutorials meant to help students of the subject grow and be put on a pathway for mastery.

Hosting on Github Pages

1. Fork this repository
2. Create a branch called `gh-pages`
3. Enable Pages to deploy from `gh-pages` branch.
4. Make sure Actions have permission to run on this repo. `gitbook_action.yml` workflow will automatically publish a Gitbook on the `gh-pages` branch.

Updating a new book

Modify the following files:

- Configuration settings: `book.json`
- Table of contents: `SUMMMARY.md`
- `cover.jpg, cover_small.jpg` (cover.jpg is published as the cover image in the PDF export generated by `.github/workflows/gitbook_action.yml`)
- `LICENSE.md`

Installing gitbook cli

Requirements: NodeJS v4 and above

`npm install -g gitbook-cli`

Local development

Install local development dependencies: `bundle install`

Install gitbook plugins: `gitbook install`

Build the static website using: `gitbook build`

Build and serve on localhost: `gitbook serve`

Note: PDF and eBook files generate via github action, and not the Gitbook generator, therefore, PDF files will be unavailable in local development.

Chapter 2

Internal link: Let's go to Chapter 1

Recommended Plugins

Hard surface modeling

HardOps/Boxcutter + Fluent

Texturing

- PBR Bridge
- UV Packmaster
- UV Squares
- Node Wrangler(included),

Simulation

- HumanGen
- Simply Cloth
- Flip Fluids
- RBD Lab

Rendering resources for Blender

Rendering can take a very long time depending on and number of variables including your scene's complexity, lighting, software version, the number of shots you have to render, whether you've optimized your scene or not, etc.

If you have more than one computer available to use, it's best to split the rendering task into separate frame ranges. If all computers are equally performant, you can equally divide the number of frames you have by the number of computers. Open the scene up on each computer, set the frame range in the render settings, and run the renders. If you have access to a computer lab, at night they are typically empty, so you can spit the frames up by as many computers as there are available. 720 frames divided by 20 computers equals 36 frames per computer. If it takes 3 minutes per frame to render, that would be 108 minutes, or 1.8 hours. If you were to render this on one computer, it would take 36 hours!

Optimize your scene

Do everything you can to lower render times. Always use the latest version of Blender, render times typically improve from version to version. I recommend using **EEVEE** to shorten render times. If using EEVEE, your scene will render much faster, though it may lose some visual quality compared to Cycles. If using Cycles, try to take advantage of the image **denoising** feature in the render settings. Balance the quality of the denoising with the **samples quality**. Typically, you can get a fairly low-samples render (128 or less) to look pretty good with denoising.

Further reading

- Rendering in Cycles: [https://docs.blender.org/manual/en/latest/render/cycles/optimizations/index.html](https://docs.blender.org/manual/en/latest/render/cycles/optimizations/index.html "https://docs.blender.org/manual/en/latest/render/cycles/optimizations/index.html"

Use more than one computer (Render farms)

Split renders up and **use multiple computers** you have physical access to. CPU render will give more consistent render results across computers, but GPU will render more quickly. You may notice visual differences between images rendered with the GPU from the CPU on the same computer. You will have to manually retrieve the frames from those computers, so it might be best for them to render directly to a shared cloud folder. Be very organized if you do this, it's easy to get confused with so many files.

To take the burden off of managing files, you can use a paid **add-on** called Crowd Render. This plugin will ask you to enter the IP address of your computers to use them as a render node. The images will be automatically sent to your computer when completed. This only works with computers that you own.

Another solution is to use an **online render farm**. Render farms have been expensive in the past, but have since come down in price and at the same time render engines have become more performant. What used to cost $150 to render a 30 second animation should now cost ~$20. I recommend looking at a range of rendering companies, starting with Concierge Render.

If you have the budget and plan to do CGI in the future, it makes sense to **invest in hardware** like an nVidia RTX 30xx or 40xx series graphics card, which perform the best/$ at the time of this writing.

Cloud rendering companies

- https://www.conciergerender.com/
  - Cycles and EEVEE
  - Cost-effective cloud rendering solution
  - $5 free credits
- Blendgrid
- https://render.st/
  - Cycles only

  - Unlimited $50/month

Network rendering plugin

- https://www.crowd-render.com/