ROB501 Assignment 4 Image-Based Visual Servoing Richard Wu | 1008078296

Introduction

This report will state the experimentally optimal control gains I found for the IBVS controller system of this assignment and the procedure I used to find them.

Procedure

Using the test harness of `part_03_learner_example.py` as a template, I tested the convergence of the IBVS controller for a range of gain values. To summarize, my testing involved the following steps:

1) The initial camera pose was changed to be more challenging:
    a) Orientation (in Euler angles): $\mathrm{roll} = \pi/3$, $\mathrm{pitch} = -\pi/16$, $\mathrm{yaw} = -\pi/6$
    b) Position: $d = \begin{bmatrix} -0.05 & 8 & -2 \end{bmatrix}^T$
2) For a gain of 0.01, run `ibvs_simulation` with `do_depth` set to `True` (to use estimated depths) and record the time it takes to convergence.
3) Repeat step 2 for gains up to 2.00 at intervals of 0.01. So a total of 200 simulations will be run.
4) Repeat steps 2 and 3 but with the `do_depth` parameter set to `False` (to use true known depths instead).
5) The gain that results in the fastest convergence time is selected as the optimal gain.

A couple notes:

a) Since simulations sometimes took several minutes, this whole experiment ran for a few hours.
a) Results from the procedure are hardware dependent. I ran this experiment on a Windows computer with a 12th Gen Intel(R) Core(TM) i5-12400F CPU.

Results

Figures 1 and 2 show convergence time (in seconds) vs. gain for the cases of using estimated and true depths, respectively. For some gains, the incremental simulation resulted in numerical issues (Jacobian pseudo-inverses could not be computed as the Jacobian was singular, due to the rotation matrix being no longer orthonormal); these gains were excluded from the results. In particular, gains from around 1.60-2.00 all resulted in computational errors, so these are not included. Further, several gains that produced very long simulation times are also removed, as they are treated as outliers (and are not important in determining optimal gain anyways).

The best gain for using estimated depths was 1.1 with a convergence time of 2.18s. The best gain for using true known depths was 1.0 with a convergence time of 1.30s.

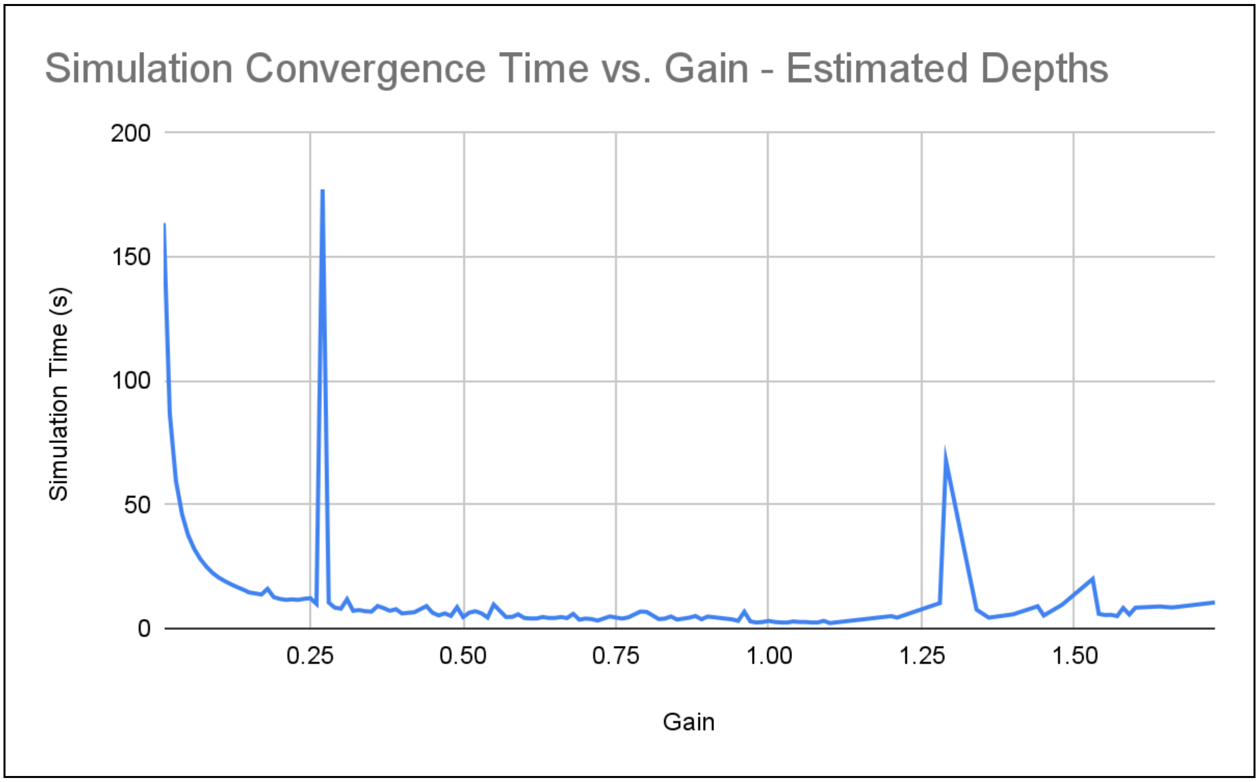

Figure 1. Simulation convergence times for range of gains, using estimated depths.

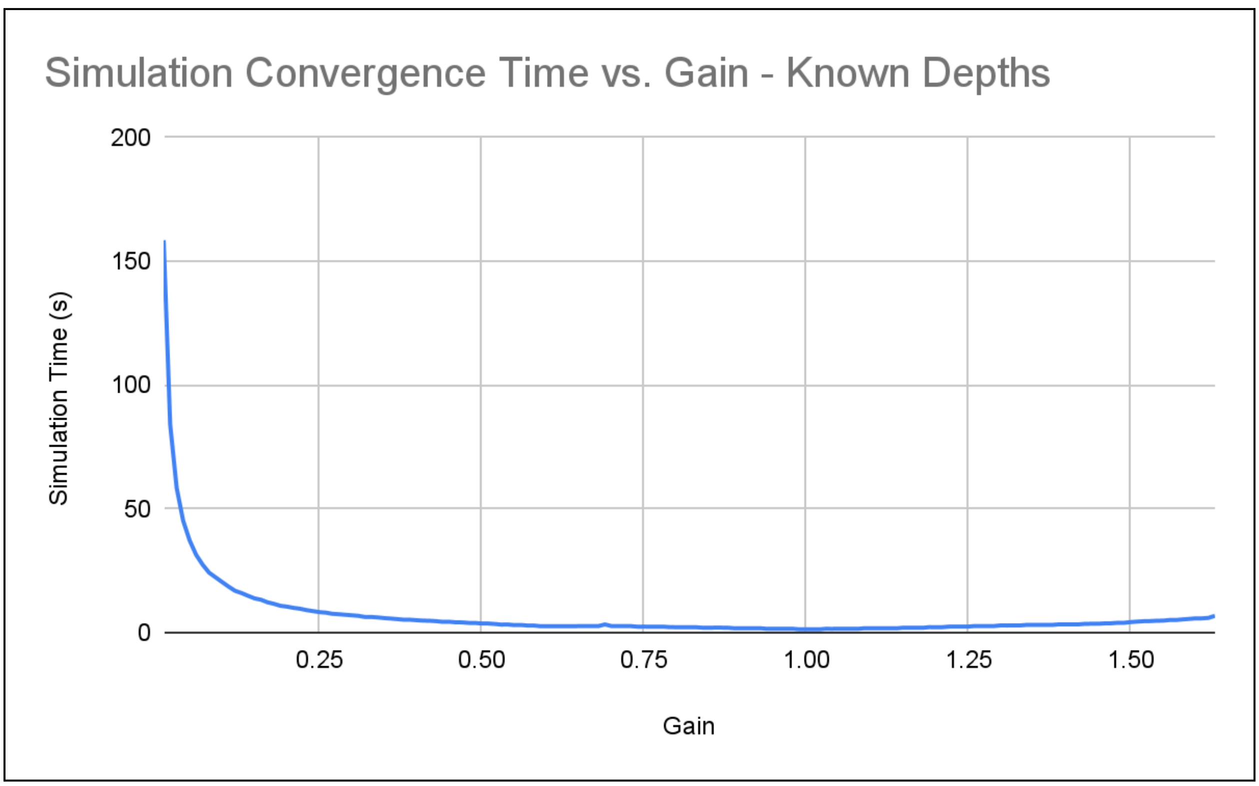

Figure 2. Simulation convergence times for range of gains, using true, known depths.

Figure 3 shows a portion of the data (from gain 0.3 to 1.6) with some outliers removed. We see a pretty clear curved trend with the minimum around a gain of 1. We can see simulation times for estimated depths is greater, it is performing worse.

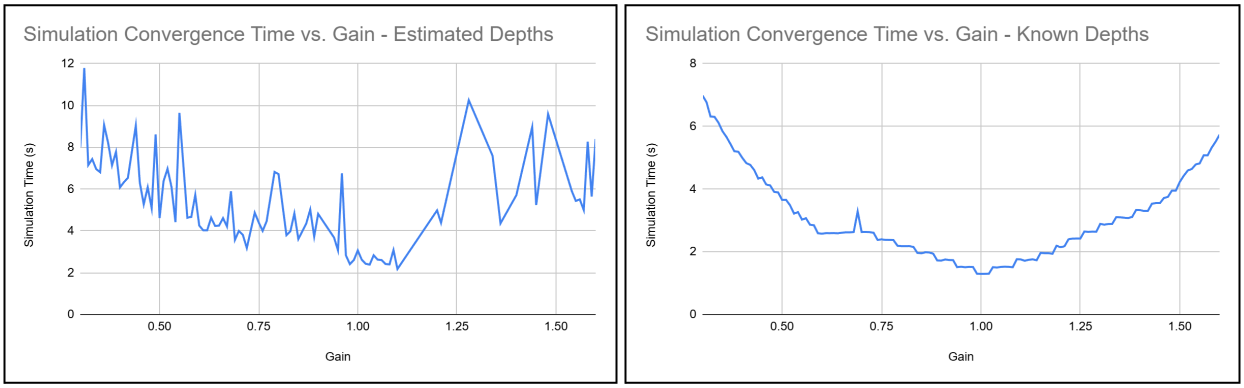

Figure 3. Side by side comparison of simulation times from gains of 0.3 - 1.6.

Using estimated depths results in more “random” convergence times. However, the ratio of simulation times for estimated to known depths is not very large (see Figure 4). The average ratio is about 1.8, so the difference in performance is not very significant. This means that estimating depths is a valid approach.

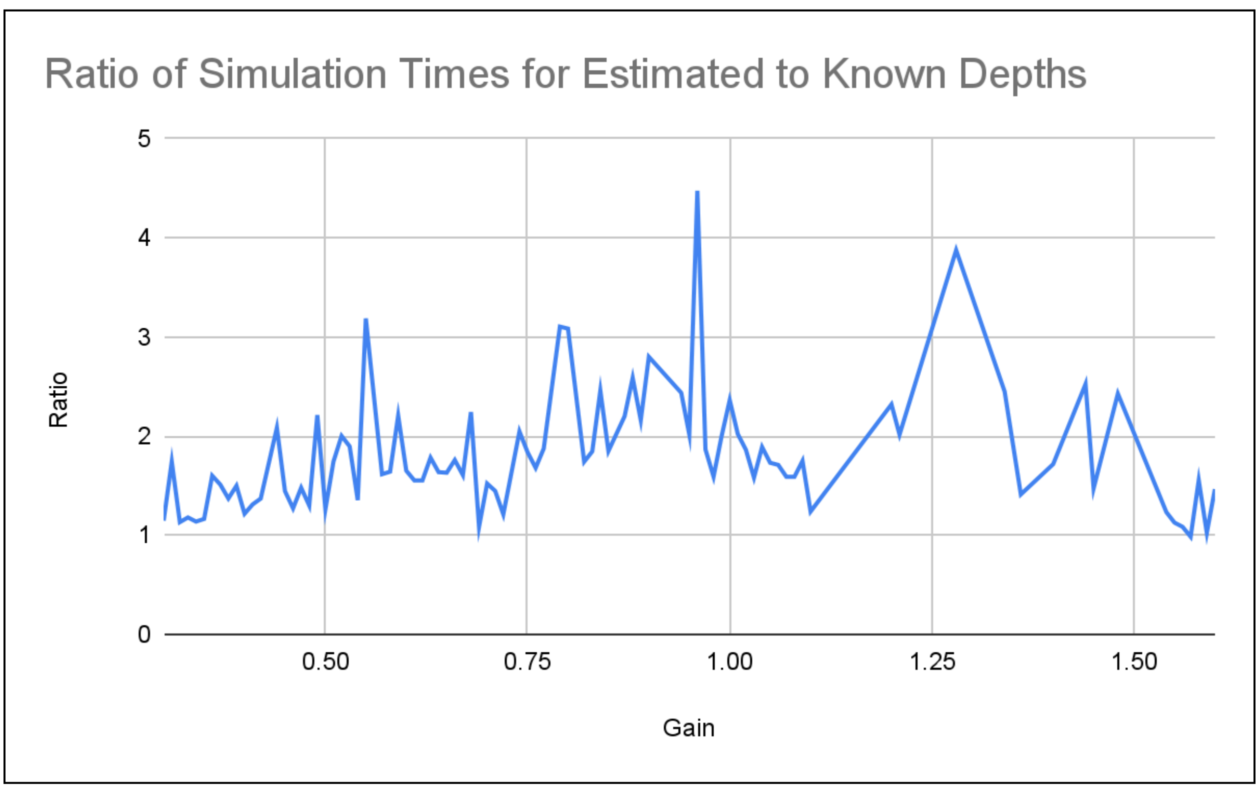

Figure 4. Ratio of simulation times for estimated to known depths for gains from 0.3-1.6.