ROB501 - Performance of Image-Based Visual Servoing (IBVS)

Eric Lefort

November 23 2024

1 Initializations

The following hard initial camera poses were used for testing the system's performance.

Hard Initialization 1: The camera is on the opposite side of the plane containing the points and rotated by π about the y-axis, making it quite far from the final position.

Hard Initialization 2: Image-based visual servoing is unable to solve for this initial configuration given the initial gain value as the points cross over each other during IBVS and the Jacobian matrix becomes singular. Very few gain values will allow this to converge. The camera is farther in the $-z$ direction from the desired position and rotated about the z-axis by π.

Hard Initialization 3: Slightly derotating the initial configuration from hard 2 allows the simple Jacobian-based IBVS algorithm to converge for more gain values.

It is difficult to converge from many of these configurations, so we restrict the search space on the gains to a more suitable and illustrative range when presenting the results. The ranges are shown in the table below along with the exact initialization parameters.

Table 1: Initialization Parameters and Gain Search Range

Initialization	**Translation**	**Roll**	**Pitch**	**Yaw**	**Gain Range**
Base	$\begin{bmatrix}-0.2 & 0.3 & -5.0\end{bmatrix}^T$	$\pi/10$	$-\pi$	$-\pi/12$	0.05 to 2.0
Hard 1	$\begin{bmatrix}-0.1 & 0.3 & 5.0\end{bmatrix}^T$	$\pi/10$	$-\pi$	$-\pi/12$	0.05 to 0.8
Hard 2	$\begin{bmatrix}-0.1 & 0.3 & -15.0\end{bmatrix}^T$	$\pi/10$	$-\pi/9$	$\pi \cdot 11/12$	0.05 to 0.2
Hard 3	$\begin{bmatrix}-0.1 & 0.3 & -15.0\end{bmatrix}^T$	$\pi/10$	$-\pi/9$	$\pi \cdot 11/12$	0.05 to 0.2

2 Optimization Method

To find the optimal gain values, Gaussian regression was employed. This method leverages machine learning techniques to model the relationship between the gain values and the performance of the IBVS system. The following kernels were used to build the regression model:

This kernel combination was selected to balance the smooth general trends (RBF), the sharp sensitivity spikes (RationalQuadratic), and the unpredictable noise (WhiteKernel).

We used the expected improvement acquisition function to sample new points with an exploration-exploitation parameter $\xi = 0.1$. The number of samples we perform depends on the initialization, as we decided to perform more samples for initializations with less smooth gain vs iterations relationships.

For the purposes of this optimization, we set a low threshold of 150 for the number of iterations to convergence. For this reason, any data points shown at 150 can be considered to not have converged.

3 Results

Known depth

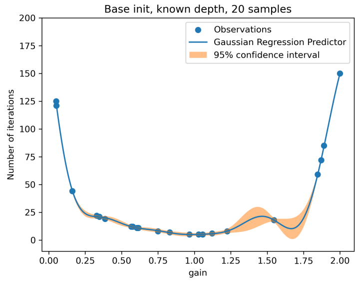

Figure 1: known depth for Base Init

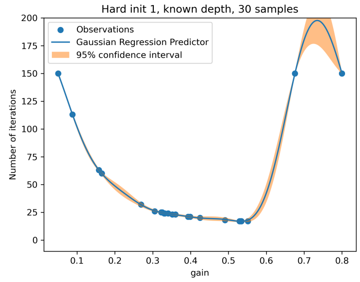

Figure 2: known depth for Hard Init 1

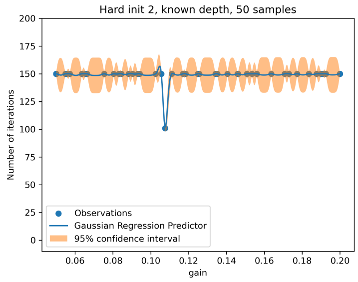

Figure 3: known depth for Hard Init 2

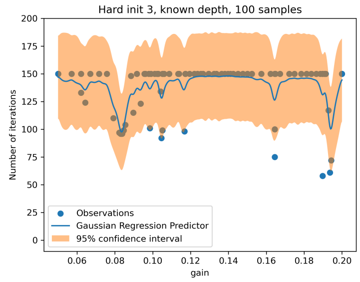

Figure 4: known depth for Hard Init 3

Table 2: Optimal (fastest) Convergence for Known Depth IBVS

Initialization	**Gain**	**Iterations**
Base Init	1.0563	5
Hard Init 1	0.5516	17
Hard Init 2	0.1076	101
Hard Init 3	0.1897	58

Approximate depth

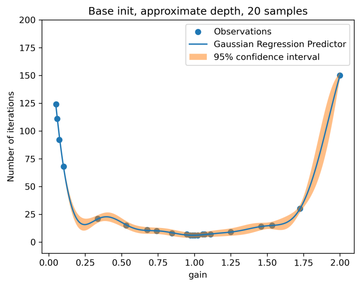

Figure 5: approximated depth for Base Init

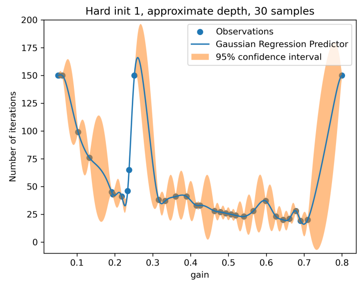

Figure 6: approximated depth for Hard Init 1

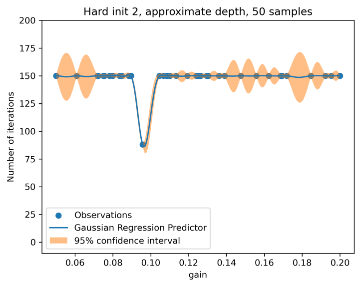

Figure 7: approximated depth for Hard Init 2

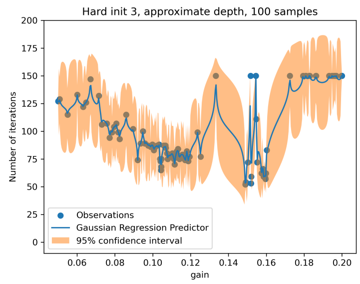

Figure 8: approximated depth for Hard Init 3

Table 3: Optimal (fastest) Convergence for Approximated Depth IBVS

Initialization	**Gain**	**Iterations**
Base Init	1.0027	6
Hard Init 1	0.6908	19
Hard Init 2	0.0957	88
Hard Init 3	0.1490	52

4 Discussion

Looking at the tables for the fastest convergence for the known depth vs approximated depth IBVS optimizations shows a surprising result, that known depth improves convergence rate for only half of the initializations and makes it worse for the other half. It is however important to examine the plots in detail. For base init and hard init 1, we see that known depth consistently reduces the required number of iterations to converge, as expected. For hard init 2, we see a single data point converging in both cases and, given the "noisiness" or sensitivity of this particular plot, we can safely attribute this discrepancy to noise, or, at the very least, we would require many more converging data points to draw useful conclusions about this scenario. For the 3rd hard initialization, we actually see that known depth increases the number of iterations till convergence. This is surprising.

In general we see that the "optimization landscape" for initializations hard 2 and hard 3 are very noisy, in the sense of being highly sensitive to small changes in the IBVS gain. For example, observe Figure 4. This has to do with the way IBVS handles these initial poses. When the camera is flipped, especially by 180 degrees, the image features may appear in a mirrored configuration, causing the Jacobian to behave unpredictably or become ill-conditioned. This makes it harder for the controller to properly converge to the desired pose and can lead to singularities appearing during servoing. Because of this inherent flaw in our controller, we expect large variability in results with these particular initializations and should not place too much weight on these results. Rather we should devise a more reliable scheme that will improve convergence outcomes or simply avoid such initializations when using IBVS.

For selecting a good, gain, I would suggest repeating this experiment with easier intial poses, that may be far from the objective pose, but avoid rotations $> \frac{3}{4}\pi$ and avoid revolving around the image subject points by more than $\frac{\pi}{2}$. This will ensure we get a good gain for the regime in which this system is stable.

5 Next Steps

It would be interesting to model the system's sensitivity to initial camera pose. A small delta could be added to the initial camera pose to see how the gain behaves for the new system. I expect it will similarly show sensitivity but with spikes in different places.