ALBiR Unit 5: Tuning a Visual Tracking System

Group 3 Jenny Shang, On Yin Chow, Julie Lin, Dominic Wilkinson, Aaruni Arora, John Bonnici

1 Introduction

Visual tracking systems are integral to both biological and robotic sensorimotor control, allowing dynamic interaction with moving objects. This experiment explores visuomotor feedback using a robotic platform equipped with an onboard camera to track moving colour-coded targets. By analysing the system's response to different input frequencies, the experiment applies system identification principles to assess feedback control dynamics.

A key focus is the characterisation of the tracking system using Bode plots, which provide insights into gain and phase lag across varying frequencies. The experiment investigates the trade-off between tracking smoothness and reaction speed by modifying the PID control parameters. This adjustment impacts the system's stability, precision, and response time.

2 Methods

2.1 Experimental Setup

This experiment uses a PixyBot with an onboard camera to track specified coloured targets. The robot was positioned 30 cm away from the laptop screen to track the MATLAB generated target videos. Detection thresholds were set to recognise red and blue as targets.

The camera was placed so that the edges of the screen corresponded to pan angles of approximately $\pm 30°$. The system's input frequency response was tested with tracking videos with increasing oscillation rates from 0.03 Hz to 1.0 Hz, and each frequency trial saved in a separate CSV file (Lin, 2024) (dragonflyneuro, 2024).

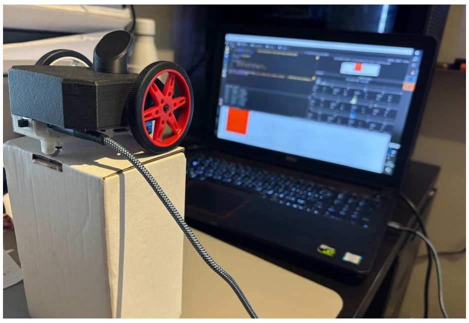


Figure 1: Experiment set-up

2.2 Data Collection and Processing

Each CSV file contained three columns: timestamps, camera/servo heading, and control signal (result of applying PID function to the error signal). Both the camera and error signals were mean-normalised and then only the steady-state segment of the data was selected for further analysis; specifically the section from the second to the last peak (found via a peak-finding algorithm). In this study, we generated a sine function to simulate the "ideal" target motion:

$$\text{Target Angle} = A\sin(2\pi f(t - t_{shift}) - \frac{3\pi}{2}). \tag{1}$$

where A is the amplitude, t_{shift} is the time point of the second peak with a $\frac{3\pi}{2}$ phase correction.

2.3 Bode Plot Construction

Root mean square (RMS) values were computed for camera and target angles, and system gain was expressed as:

$$\text{Gain (dB)} = 20\log_{10}\left(\frac{\text{RMS(Camera Angle)}}{\text{RMS(Target Angle)}}\right). \tag{2}$$

Phase lag was determined by smoothening and fitting the camera and target angles to a sine wave using the (`lsqcurvefit`) function in MATLAB, obtaining phases ϕ_{camera} and ϕ_{target} respectively. The difference $\phi_{\text{lag}} = \phi_{\text{target}} - \phi_{\text{camera}}$ defines the system's phase response at each frequency (Tanaka, 2025).

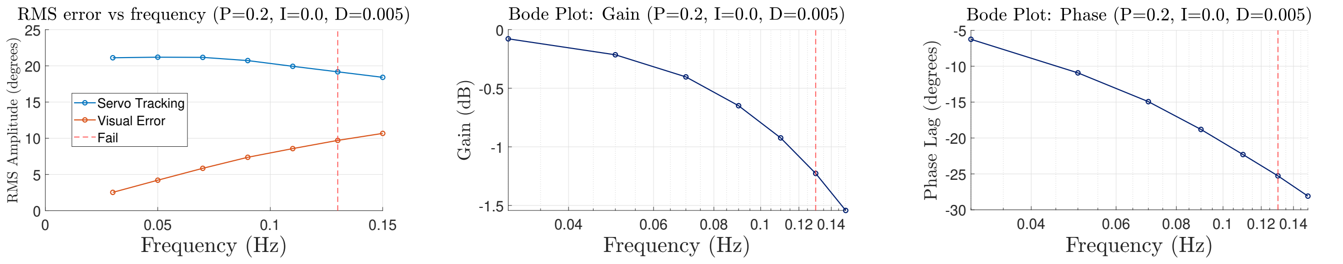
2.4 Control System and Stability Margins

In order to evaluate the impact of different PID settings on the tracking performance, two distinct configurations were implemented. Default PID values of $P = 0.2, I = 0.0, D = 0.005$, and tuned values of $P = 1.0, I = 0.05, D = 0.05$ were tested. Bode plots of gain and phase versus frequency were generated for both configurations to identify the gain crossover (where Gain = 0 dB) and the phase crossover (where $\phi = -180°$).

3 Results & Discussion

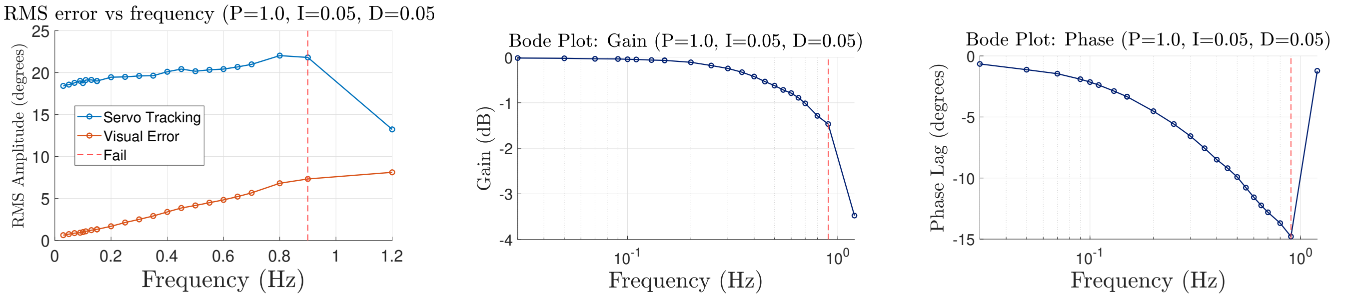
3.1 Default PID Configuration

The default PID configuration successfully tracked the target, with the error increasing with frequency until it failed to track at 0.15 Hz. There is a decrease in servo-tracking amplitude against frequency (Fig 2(a)) as the control system struggles to keep up with the moving target and oscillates with a lower amplitude than the target for frequencies greater than 0.09 Hz. This is reflected in the bode plot (Fig 2); at low frequencies, the system maintained a gain close to 0 dB which represents near-perfect tracking. But, as the frequency increased, the gain decreased, suggesting that the servo was no longer fully replicating the input signal's amplitude, and the tracking accuracy decreased. The phase lag decreases rapidly, reaching approximately $-25°$ at failure, indicating greater instability at higher frequencies.



((a)) ((b)) ((c))

Figure 2: Default PID plots: (a) Servo Tracking vs Visual Error, (b) Bode plot of gain, and (c) Bode plot of phase lag.



((a)) ((b)) ((c))

Figure 3: Tuned PID plots: (a) Servo Tracking vs Visual Error, (b) Bode plot of gain, and (c) Bode plot of phase lag.

3.2 Tuned PID Configuration

Following initial tests with the default PID parameters, the controller was tuned to enhance tracking performance (Instruments, 2023). With a primary focus on improving the camera's ability to track higher frequency videos, over smoother tracking at the frequencies already achieved. The P gain was increased from 0.2 to 1.0 to reduce the rise time, thereby improving the system's response speed. An I term of 0.05 was introduced to minimize steady-state error; however, its value was kept low to prevent the excessive lag often associated with integral action, as system speed was prioritized. Increasing P and introducing I reduced system stability, necessitating an increase for D from 0.005 to 0.05 to provide sufficient damping. This adjustment reduced overshoot and settling time, restoring system stability.

These adjustments extended the frequency range over which the camera could effectively track the target, with the maximum trackable frequency before failure increased from 0.13 Hz (default PID) to 0.9 Hz (tuned PID). Additionally, the tuned system maintained a gain close to 0 dB (Fig 3(b)) and exhibited reduced phase lag (Fig 3(c)) over a broader frequency range.

3.3 Comparison

The tuned PID configuration improved tracking performance. The visual error remained lower than the corresponding default PID visual error, and failed at a higher frequency of 0.9 Hz. The failure is suspected to be due to the camera's low frame rate to track high-frequency oscillations. The RMS plot (Fig 3(a)) underlines the ideal, undershoot, and overshoot trend followed by the servo over the frequency range. The relatively stable and close to 0 dB gain in Fig 3(b) shows better tracking accuracies over a broader frequency range. The phase lag in Fig 3(c) only drops to $-4°$, as compared to $-25°$ in the default configuration at failure (0.13 Hz), highlighting an improvement in stability.

4 Conclusion

This study highlights the importance of PID tuning for achieving accurate visual tracking. The default PID performed well at lower frequencies but exhibited increased error and reduced gain beyond 0.1 Hz. The tuned PID improved stability and reduced error across the tested range; however, it also introduced overshoot at higher frequencies. Different approaches could be taken to achieve smoother results, but that would come at the cost of a limited frequency range. Future work should focus on improving data collection methods, using higher frame-rate cameras, applying reinforcement learning algorithms to optimize control, and automating robot positioning to minimize variability and enhance reliability.

5 References and Acknowledgements

We acknowledge the use of ChatGPT (OpenAI, chat.openai.com) to understand the code and correct grammar. We confirm that no content generated by AI has been presented as our work.

References

dragonflyneuro (2024). Github - dragonflyneuro/icl_bioeng_albir_openmv: Animal locomotion and bioinspired robotics course code repository. GitHub. [Online]. Available: https://github.com/dragonflyneuro/ICL_BioEng_ALBIR_OpenMV.

Instruments, N. (2023). The pid controller & theory explained. Available at: https://www.ni.com/en/shop/labview/pid-theory-explained.html.

Lin, H. T. (2024). Unit 5: Target tracking worksheet. Animal Locomotion and Bioinspired Robotics Course, Department of Bioengineering, Imperial College London.

Tanaka, R. (2025). Signals and control 2 lecture slides and notes. Department of Bioengineering, Imperial College London.