Laser Pointer Turret Based Mosquito Air Defence System

Final Report

A. Hartman
20475323

Submitted as partial fulfilment of the requirements of Project EPR402
in the Department of Electrical, Electronic and Computer Engineering
University of Pretoria

November 2023

Study leader: Prof. P. de Villiers

Part 1. Preamble

This report describes work that I did *<to be completed>*.

Project proposal and technical documentation
This main report contains an unaltered copy of the approved Project Proposal (as Part 2 of the report).

Technical documentation appears in Part 4 (Appendix).

All the code that I developed appears as a separate submission on the AMS.

Project history
This project makes extensive use of existing algorithms on ... Some of the algorithms I used were adapted from ... Where other authors' work has been used, it has been cited appropriately, and the rest of the work reported on here, is entirely my own.

Language editing
This document has been language edited by a knowledgeable person. By submitting this document in its present form, I declare that this is the written material that I wish to be examined on.

My language editor was ____________________________________.

____________________________________	____________________
Language editor signature	*Date*

Declaration

I, A. Hartman understand what plagiarism is and have carefully studied the plagiarism policy of the University. I hereby declare that all the work described in this report is my own, except where explicitly indicated otherwise. Although I may have discussed the design and investigation with my study leader, fellow students or consulted various books, articles or the internet, the design/investigative work is my own. I have mastered the design and I have made all the required calculations in my lab book (and/or they are reflected in this report) to authenticate this. I am not presenting a complete solution of someone else.

Wherever I have used information from other sources, I have given credit by proper and complete referencing of the source material so that it can be clearly discerned what is my own work and what was quoted from other sources. I acknowledge that failure to comply with the instructions regarding referencing will be regarded as plagiarism. If there is any doubt about the authenticity of my work, I am willing to attend an oral ancillary examination/evaluation about the work.

I certify that the Project Proposal appearing as the Introduction section of the report is a verbatim copy of the approved Project Proposal.

A. Hartman	Date

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

GPIO	general purpose input/output
GPU	graphics processing unit
MADS	mosquito air defence system
PID	proportional-integral-derivative
RPM	revolutions per minute

Part 2. Project definition: approved Project Proposal

This section contains the problem identification in the form of the complete approved Project Proposal, unaltered from the final approved version that appears on the AMS.

Part 3. Main Report

1. Literature study

Malaria is still one of the leading causes of death in low-income countries according to the World Health Organisation [1]. Mosquitoes that do not carry diseases are also a general nuisance in the everyday life of people living in mosquito-prone areas. Therefore, it is necessary to pursue improvement in our defence against mosquitoes.

To be able to design a laser pointer turret-based mosquito air defence system it is necessary to understand the principles of computer vision object detection and real-time tracking.

One approach towards tracking is to perform pattern matching. In general pattern matching is searching and checking images for the presence of other given images (patterns) to find and mark the patterns' locations (if any) within the given images. However, the study conducted by Hurtik et al. [2] presents results that indicate the best frame rate they achieved was 0.43 frames per second. This is far too slow to be used in a real-time tracking application.

Another approach is to perform particle filter-based tracking. This considers the proximity and behaviour of other targets. In the case of social insect tracking, it is known that two targets cannot occupy the same space, and targets will actively avoid collisions. Unfortunately, the joint particle tracker proposed in [3] suffers from exponential complexity.

A popular approach is to separate the detection and tracking functions. While numerous deep learning algorithms can detect objects based on appearance, it is worth noting that mosquitoes, particularly when not filmed up close, prove too minute to be reliably detected using appearance-based methods. A viable alternative is to detect objects by isolating the background and foreground of the image [4]. The foreground of the image contains the objects of interest. In [5] objects that are too close to one another are split into two and abnormally small objects are merged.

A proposed tracking method is the Simple Online Real-time Tracking (SORT) algorithm [6]. The algorithm is composed of an estimation model which makes use of a Kalman filter and a data association system that is solved optimally using the Hungarian algorithm.

In the proposed mosquito air defence system mosquito detection will be based on background and foreground isolation. This method is suitable because the system will operate in a known test environment where the background will change minimally. The online real-time nature of this system makes the case for pattern matching and particle filtering unfavourable because of the computationally intensive nature of these techniques. The methods in [4], [5], and [6] will be further investigated for the proposed system.

2. Approach

The aim of this project was to develop a system that would illuminate flying mosquitoes with a laser turret. Throughout the remainder of this document this system will be referred to as the mosquito air defence system (MADS). The project can be viewed in terms of a set of integrated subsystems. The subsystems are as follows:

Laser Turret Control System
This subsystem is responsible for controlling the laser turret. The laser control system is discussed in

Laser Detection System
This subsystem is responsible for detecting the actual position of the laser with the camera. The laser detection is required to provide feedback to the laser turret control system. The laser detection system is discussed in

Mosquito Detection System
This subsystem is responsible for detecting the position of the mosquitoes. The mosquito detection system is discussed in

Mosquito Tracking System
This subsystem is responsible for tracking the mosquitoes and predicting their future positions. The mosquito tracking system is discussed in

The subsystems were integrated on a real-time embedded system and will be discussed in detail in the relevant sections. The system is designed to operate in a controlled environment constructed specifically for this project. The functional block diagram of the MADS is shown in Figure 1.

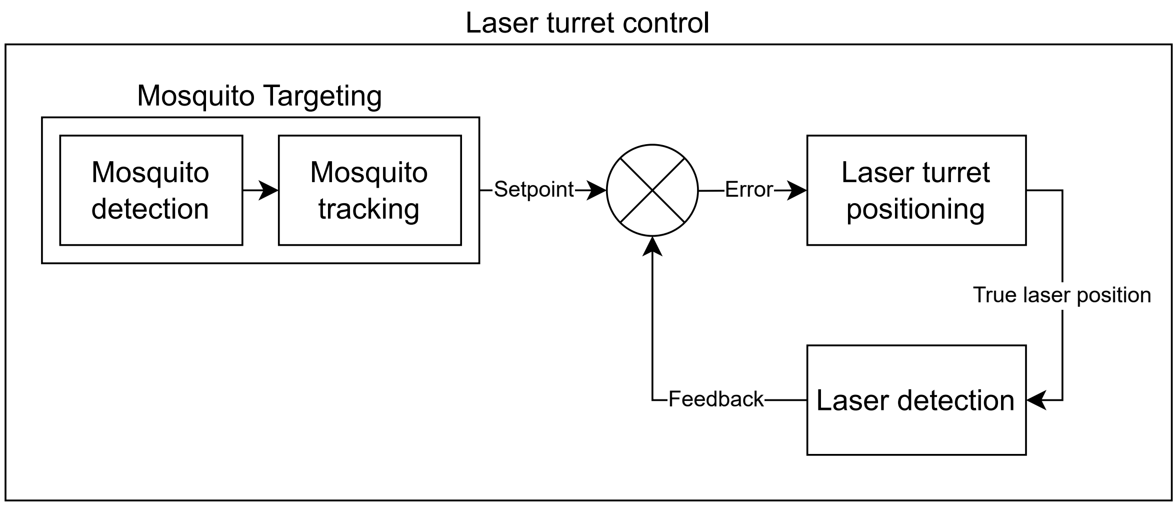

Figure 1.
Functional block diagram of the MADS.

The design choices for the MADS were made with careful consideration of the implicit real-time operation requirement of the system. All aspects of the MADS were designed to be as lightweight as possible in terms of computational complexity.

3. Design and implementation

3.1 Things I made

- Laser turret housing.
- Worked out the speed, torque, and step size required for the stepper motors.
- Interfaced with stepper motors using GPIO pins to drive the stepper motor drivers.
- Mapping between camera pixels and real-world co-ordinates. Had to do camera calibration and account for different perspectives of camera and turret.
- Calculate steps from angle required based on distance using Pythagoras.
- Distinguish between laser reflections using the geometry of the laser turret.
- Laser detection from first principles optimised with GPU kernels. 1. Gaussian smoothing, 2. Binarise with threshold, 3. Morphological operations (closing and opening), 4. Connected components labelling, 5. Find centroids of connected components, 6. Distinguish laser detections with turret geometry.
- Mosquito detection. Same image processing steps expect step 2 is either a less than threshold or background subtraction.
- Mosquito tracking using SORT algorithm. (Kalman filter and Hungarian algorithm).
- Laser turret PID controller. Must still be tuned because the error calculated is inaccurate. Should tuning be done without developing a model? Or should a model be developed? How do you develop a model?
- Feedback for turret. The current steps are saved at the instance that the frame is captured. The frame is then processed using the laser detection system. The pixel co-ordinates are converted to steps. The step error is calculated and added to the current steps.
- System integration on embedded system running in real-time on multiple threads.

3.2 Design summary

This section summarises the project design tasks and how they were implemented (see Table 1).

Deliverable or task	Implementation	Completion of deliverable or task, and section in the report
The mosquito detection subsystem had to be designed and implemented by the student.	The mosquito detection subsystem was designed and implemented from first principles.	Completed.
The laser detection subsystem had to be designed and implemented by the student.	The laser detection subsystem was designed and implemented from first principles.	Completed.
The laser turret control subsystem had to be designed and implemented by the student.	The laser turret control subsystem was designed and implemented from first principles.	Completed.
The mosquito tracking subsystem had to be designed and implemented by the student.	The mosquito tracking subsystem was designed and implemented from first principles.	Completed.
The various subsystems had to be integrated on a real-time embedded system.	The various subsystems were integrated on a real-time embedded system.	Completed.
Appropriate motors needed to be selected for the laser turret.	The stepper motors were selected based on the requirements of the laser turret.	Completed.

Table 1.
Design summary.

3.3 Theoretical analysis and modelling

3.3.1 Mapping camera pixels to real-world co-ordinates

To control the laser turret the setpoint and laser position detected with the camera must be converted from the pixels in the camera's co-ordinate frame to the turret's co-ordinate frame. This is done using the camera's intrinsic and extrinsic parameters. The camera's intrinsic parameters are used to convert the pixels to a ray in the camera's co-ordinate frame. The camera's extrinsic parameters are used to convert the ray in the camera's co-ordinate frame to a ray in the turret's co-ordinate frame. The ray in the turret's co-ordinate frame is then converted to the turret's setpoint and laser position. The intrinsic and extrinsic parameters are found using camera calibration.

3.3.2 Camera calibration

a. Intrinsic parameters

These parameters are inherent to the camera and remain constant unless the camera's internal settings (like focus) are changed. They are typically found by calibrating the camera using multiple views of a known pattern (like a checkerboard).

- **Camera Matrix (K)**: Defines the camera's internal characteristics. The principal components are:

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{1}$$

  where f_x and f_y are the focal lengths in pixels and c_x, c_y are the co-ordinates of the principal point. The principal point is the pixel co-ordinate where the camera's principal axis intersects the image plane.

- **Distortion Coefficients (D)**: Captures lens distortion. This is a vector with up to 5 elements in the common "plumb-bob" model of OpenCV

$$D = [k_1, k_2, p_1, p_2, k_3]$$

  where k_1, k_2, k_3 are radial distortion coefficients and p_1, p_2 are tangential distortion coefficients.

b. Extrinsic Parameters

These parameters capture the camera's orientation and position concerning the world or an external reference frame.

- **Rotation Vector (rvec)**: Represents the orientation of the camera. It's a 3x1 vector used to derive the rotation matrix R.
- **Translation Vector (tvec)**: Represents the position of the camera. It's a 3x1 vector capturing the translation in X, Y, and Z directions.

They represent the position and orientation of the camera relative to the world (or in your case, the turret's frame). Every time you move the camera or change the scene, these parameters would change. These are found using functions like `solvePnP` in OpenCV, which computes the pose of an object given some known 3D points on the object and their corresponding 2D projections in the image.

3.4 Simulation and Prototyping

Detection and tracking done in python with videos.

3.5 Hardware design

Turret inspired by 2d laser scanners. Motor step resolution, speed, and torque calculations. Motor driver selection based on microstepping capabilities. Turret CAD design. Math translation between angle and distance of laser.

3.5.1 Mosquito enclosure and system positioning

The mosquitoes were placed inside an enclosure with a white lining and a glass front panel for the laser beam to shine through. The enclosure is a rectangular prism with dimensions $90 \times 38 \times 32$ cm as seen in Figure 2. The laser turret and the camera were placed outside the enclosure in known positions relative to the enclosure. The bracket, shown in Figure 3, was designed to hold the laser turret and the camera in place relative to the mosquito enclosure.

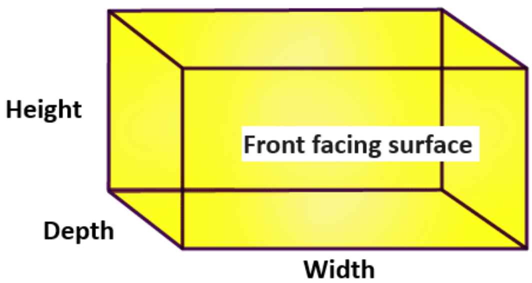

Figure 2.
Mosquito enclosure dimensions.

3.5.2 Laser turret design

The laser turret design was inspired by commercial two-axis laser scanners. A schematic for a typical two-axis laser scanner can be seen in Figure 4. The mirrors are connected directly to the output shaft of the motor. The origin of the laser turret is where the laser beam shines orthogonal to the mosquito plane, which occurs when the two mirrors are at 45° relative to the angle of the laser beam when the origin of the laser beam is parallel to the mosquito plane. When a single axis of the laser turret is considered it can be seen that a right triangle is formed between the laser turret and the mosquito plane as shown in Figure 5. Using the properties of a right triangle the mirror angle θ required to shine the laser a distance $\pm x$ from the origin of the laser turret can be calculated using

$$\theta = \arctan\left(\frac{\pm x}{z}\right) \tag{2}$$

where z is the distance between the turret and the mosquito plane.

The specific geometry of the laser turret was designed with the goal to practically obtain

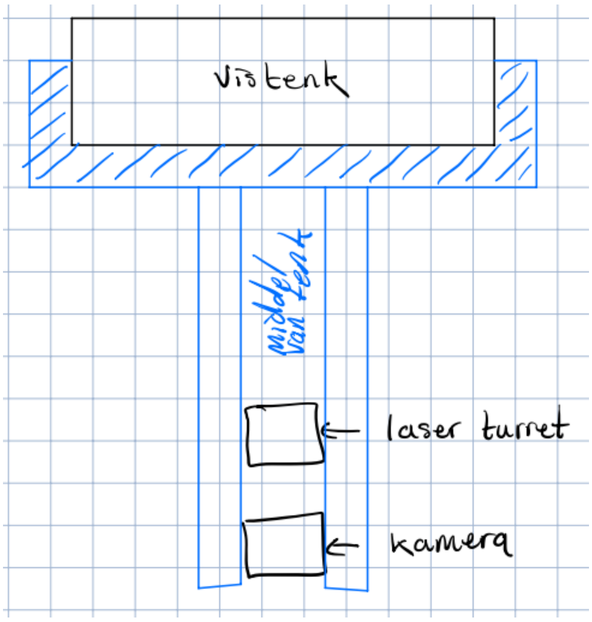

Figure 3.
System positioning bracket.

a sufficiently small lateral step size of the laser on the mosquito plane, while maintaining sufficient speed.

a. Lateral step size and speed

The lateral step size of the laser must be on the order of 2 mm (size of mosquito plus size of laser).

The required lateral speed is determined in accordance with the field specifications the longest length of the mosquito enclosure is 1 m. To ensure that the laser could illuminate the set point within 2 seconds of receiving a step input it was assumed that the laser must be able to move with a velocity v of least 1 m/s opposed to 0.5 m/s required to move the 1 m length of the mosquito enclosure within 2 seconds. This was done to accommodate for the settling time of the control system. This is also faster than the top speed of a mosquito which is 0.67 m/s according to SOURCE.

b. Lateral to angular step size

In the design of the laser turret there were multiple dimensions that were dependent on each other, it is up to the designer to choose certain practical dimensions from which the rest of the dimensions can be calculated. In this design the dimensions of the mirrors were chosen for practicality as 30×15 mm. These dimensions will be used throughout the rest of the design.

The turret axis on which the laser beam is first incident on will be referred to as the first axis and the other axis will be referred to as the second axis for the remainder of this discussion. The rotation range of the first axis is bounded by the geometry of the turret since the laser beam reflected from the first axis must be incident on the second axis. With the chosen mirror dimensions, the maximum angle through which the first axis can rotate from the origin (45° point) is $\theta_{max} = \pm 20°$ (add conservatively estimated) resulting in a 40° range of motion. This

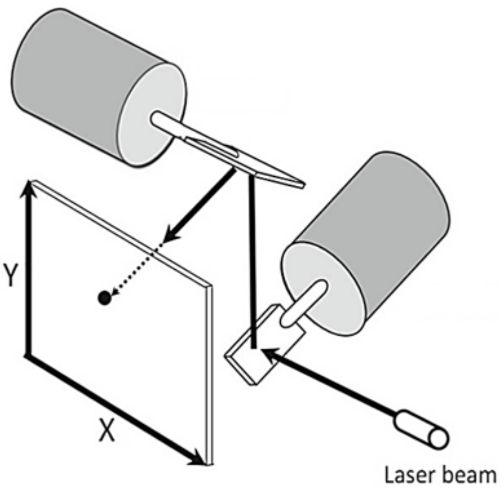

Figure 4.
Two-axis laser scanner schematic. This figure was modified from [7].

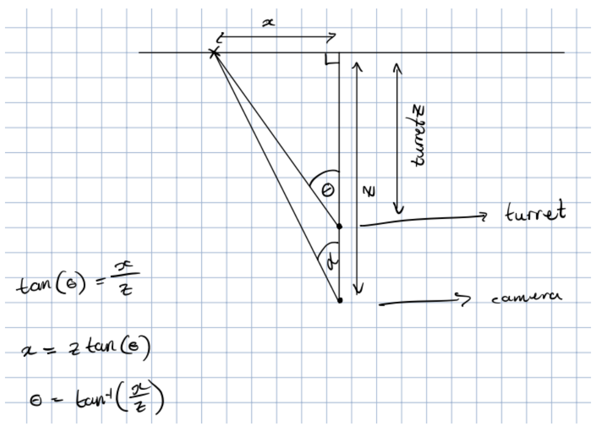

Figure 5.
Mirror angle calculation.

was determined geometrically in Figure 6. The lines representing the mirrors in Figure 6 are 30 mm long and the whole figure is drawn to scale. The rotation range of the second axis is not bounded by the geometry of the turret since the laser beam reflected from the second axis is incident on the mosquito plane. Therefore, the turret was oriented such that the first axis moves the laser beam parallel to the shorter width of the mosquito enclosure and the second axis moves the laser beam parallel to the longer length of the mosquito enclosure.

To maximise the angle through which the turret must rotate to move the laser beam across the height of the mosquito enclosure the minimum distance z_{min} between the turret and the mosquito plane was determined by rearranging Equation 2 to give

$$z = \frac{\pm x}{\tan\theta}. \tag{3}$$

where x = 38 cm is the height of the mosquito enclosure. The minimum distance z_{min} was calculated by substituting $\theta_m ax$ = 20° into Equation 3 to give

$$z_{min} = \frac{38\text{cm}}{\tan 20°} = 104.4.\text{cm}. \tag{4}$$

The required motor step resolution was calculated by substituting x_{min} and z_{min} into Equation 2 to give

$$\arctan\left(\frac{x_{min}}{z_{min}}\right) = \arctan\left(\frac{2\text{mm}}{104.4\text{cm}}\right) = 0.11° \tag{5}$$

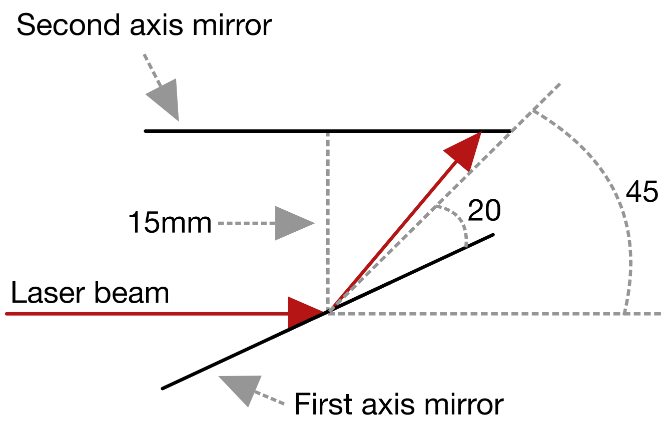

Figure 6.
Rotation range of first axis with chosen mirror dimensions.

c. Lateral speed to angular speed

From paragraph b it is known that 0.37999

$$\begin{aligned} 0.37999\,\text{m}\,\text{s}^{-1} &= 40°\,\text{s}^{-1} \\ \Longrightarrow 0.37999 \times \frac{1}{0.37999} &= 40 \times \frac{1}{0.37999} \\ \Longrightarrow 1\,\text{m}\,\text{s}^{-1} &= 105.26978\,\text{s}^{-1} \end{aligned} \tag{6}$$

The angular speed required to move the laser beam at a lateral speed $v = 1\,\text{m}\,\text{s}^{-1}$ was calculated with θ_{max} determined in paragraph b using

$$\omega = v \times \theta_{max}. \tag{7}$$

3.5.3 Motor requirements

~~The motor requirements were determined based on system requirement 4 (The laser must be able to illuminate the set point within 2 seconds accurate to within 1 mm.).~~

The need for precise position control of the laser meant that only stepper and servo motors were considered. The required torque and required revolutions per minute (RPM) was calculated to enable appropriate motor selection. ~~and the resulting laser step size on the mosquito plane were calculated.~~ In the calculations only a single axis of the turret was considered since the motors and mirrors for both axes will be identical and operate independently.

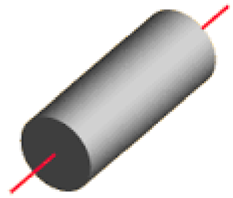

Figure 7.
Central axis of a cylinder.

The required torque τ was calculated using

$$\tau = I\alpha \tag{8}$$

where I is the moment of inertia and α is the angular acceleration.

The moment of inertia for the mirror was calculated by assuming the mirror was solid cylinder with diameter d equal to the width of the mirror. This will produce an inflated moment of inertia, however, this is not a concern since having extra torque is not a problem. The moment of inertia I of a cylinder with rotation about its central axis as seen in Figure 7 is given by

$$I = \frac{1}{2}Mr^2 \tag{9}$$

where M is the mass of the cylinder and $r = {}^{d}\!/_{2}$ is the radius of the cylinder. The mass of the cylinder M was calculated using the density of glass $\rho = 2500\text{kg/m}^3$ and the volume of the cylinder V given by

$$V = \pi r^2 h. \tag{10}$$

The height h and radius r of the cylinder was determined by assuming practical dimensions for the mirror. The dimensions of the mirror was assumed to be 30 mm by 15 mm. Thus, the mass of the mirror M is given by

$$M = \rho V = 2500\text{kg/m}^3 \times \pi\left(\frac{0.015}{2}\text{m}\right)^2 0.03\text{m} = 0.01325\text{kg} = 13.25\text{g}. \tag{11}$$

Given the above assumptions, the moment of inertia I was calculated as

$$I = \frac{1}{2}Mr^2 = \frac{1}{2} \times 0.01325\text{kg} \times \left(\frac{0.015}{2}\text{m}\right)^2 = 1.172 \times 10^{-7}\text{kg}\cdot\text{m}^2. \tag{12}$$

The angular acceleration of the motor α is calculated by

$$\alpha = \frac{\Delta\omega}{\Delta t} \tag{13}$$

where $\Delta\omega = \omega_f - \omega_i$ is the change in angular velocity and Δt is the change in time. This velocity was translated into the angular velocity ω_f using

$$\omega = v \times \theta_{max} \tag{14}$$

where θ_{max} is the angle through which the mirror must rotate to produce a 1 m lateral displacement of the laser on the mosquito plane.

The change in time Δt was determined by assuming the laser must be able to accelerate from a stand still to 1 m/s extremely rapidly to ensure that the motors could respond to the irregular flight pattern associated with a mosquito. It was assumed that this acceleration must occur within 10 ms.

Thus, the required angular acceleration α is given by

$$\alpha = \frac{\Delta\omega}{\Delta t} = \frac{0.698\text{rad/s}}{0.01\text{s}} = 69.8\text{rad/s}^2. \tag{15}$$

and thus the required torque τ is given by

$$\tau = I\alpha = 1.172 \times 10^{-7}\text{kg}\cdot\text{m}^2 \times 69.8\text{rad/s}^2 = 8.17 \times 10^{-6}\text{N}\cdot\text{m}. \tag{16}$$

3.6 Hardware implementation

general purpose input/output (GPIO) interfacing with stepper motor driver.

3.7 Software design

Mapping between pixels and real distance. Mosquito detection and tracking. Laser detection and tracking. proportional-integral-derivative (PID) controller. Feedback for turret. Distinguishing between laser reflections.

3.8 Software implementation and optimisation

Only using red channel, graphics processing unit (GPU), low resolution, etc.

3.9 Final system integration and testing

Real-time multi threading.

4. Results

4.1 Summary of results achieved

4.2 Qualification tests

5. Discussion

5.1 Critical evaluation of the design

5.1.1 Interpretation of results

5.1.2 Critical evaluation

5.1.3 Unsolved problems

5.1.4 Strong points of the design

5.1.5 Expected failure conditions

5.2 Considerations in the design

5.2.1 Ergonomics

5.2.2 Health and safety

5.2.3 Environmental impact

5.2.4 Social and legal impact

5.2.5 Ethics clearance

6. Conclusion

6.1 Summary of the work completed

6.2 Summary of the observations and findings

6.3 Contribution

6.4 Future work

7. References

[1] W. H. Organisation. (2020, 12) The top 10 causes of death. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death

[2] P. Hurtik, D. Číž, O. Kaláb, D. Musiolek, P. Kočáek, and M. Tomis, "Software for visual insect tracking based on f-transform pattern matching," in IEEE Second International Conference on Data Stream Mining & Processing, Lviv, Ukraine, 2018.

[3] Z. Khan, T. Balch, and F. Dellaert, "Efficient particle filter-based tracking of multiple interacting targets using an mrf-based motion model," in Proceedings of the 2003 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems, Las Vegas, Nevada, USA, 2003.

[4] W. Liang, H. Wang, and H. Krim, "A behaviour-based evalutation of product quality," in International Conference on Acoustics, Speech, and Signal Processing, 2016.

[5] Y. Bao and H. Krim, "Video tracking of insect flight path: Towards behavioral assessment," in IEEE, 2018.

[6] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online realtime tracking," Queensland University of Technology, University of Sydney, Tech. Rep., 2017.

[7] "Polarization effects of galvos in SLA 3D printing," https://henryquach.org/galvopolarization.html, accessed: 2023-10-14.

Part 4. Appendix: technical documentation

HARDWARE part of the project

Record 1. System block diagram

Record 2. Systems level description of the design

Record 3. Complete circuit diagrams and description

Record 4. Hardware acceptance test procedure

Record 5. User guide

SOFTWARE part of the project

Record 6. Software process flow diagrams

Record 7. Explanation of software modules

Record 8. Complete source code

Complete code has been submitted separately on the AMS.

Record 9. Software acceptance test procedure

Record 10. Software user guide

EXPERIMENTAL DATA

Record 11. Experimental data