

## ENSC 405W Grading Rubric for Design Specification

Criteria	Details	Marks
<b>Introduction/Background</b>	Introduces basic purpose of the project.	<b>/05%</b>
<b>Content</b>	Document explains the design specifications with appropriate justification for the design approach chosen. Includes descriptions of the physics (or chemistry, biology, geology, meteorology, etc.) underlying the choices.	<b>/20%</b>
<b>Technical Correctness</b>	Ideas presented represent design specifications that are expected to be met. Specifications are presented using tables, graphs, and figures where possible (rather than over-reliance upon text). Equations and graphs are used to back up/illustrate the science/engineering underlying the design.	<b>/25%</b>
<b>Process Details</b>	Specification distinguishes between design details for present project version and later stages of project (i.e., proof-of-concept, prototype, and production versions). Numbering of design specs matches up with numbering for requirements specs (as necessary and possible).	<b>/15%</b>
<b>Test Plan Appendix</b>	Provides a test plan outlining the requirements for the final project version. Project success for ENSC 405W will be measured against this test plan.	<b>/10%</b>
<b>User Interface Appendix</b>	Summarizes requirements for the User Interface (based upon the lectures and the concepts outlined in the Donald Norman textbook).	<b>Graded Separately</b>
<b>440 Plan Appendix</b>	Analyses progress in 405W and outlines development plans for 440. Includes an updated timeline, budget, market analysis, and changes in scope. Analyses ongoing problems and proposes solutions.	<b>Graded Separately</b>
<b>Conclusion/References</b>	Summarizes functionality. Includes references for information sources.	<b>/05%</b>
<b>Presentation/Organization</b>	Document looks like a professional specification. Ideas follow logically.	<b>/05%</b>
<b>Format/Correctness/Style</b>	Includes letter of transmittal, title page, abstract, table of contents, list of figures and tables, glossary, and references. Pages are numbered, figures and tables are introduced, headings are numbered, etc. References and citations are properly formatted. Correct spelling, grammar, and punctuation. Style is clear, concise, and coherent. Uses passive voice judiciously.	<b>/15%</b>
<b>Comments</b>		



## EAGLE EYE SYSTEMS

Dr. Andrew Rawicz  
School of Engineering Science  
Simon Fraser University  
Burnaby, British Columbia  
V5A 1S6

RE: ENSC 405/440 Design Specifications for the Eagle Eye Tracker

Dear Dr. Rawicz,

The following document contains an overview of the design specifications of our project for ENSC 405/440: the Eagle Eye Tracker. The goal of the project is to create an automated targeting and tracking system. The system will be ideal in locations where aerial security is of utmost importance – such as airports, prisons, and international borders. The Eagle Eye Tracker’s design is highly flexible due to its deep learning-based algorithm, and thus has many more potential applications – including pest control, cinematography, and surveillance.

The purpose of the design specifications is to provide both specific and high-level design details regarding how the project will be implemented, with distinctions being made between the various stages of the project. These design specifications are referenced to the requirements defined in our previous requirements specifications document.

This document outlines progress made thus far, as well as anticipated work for the next stage of the project as it transitions into ENSC 440. Appendices are attached which describe test plans for the project, as well as user interface design and planning for the prototype development.

If you have any questions or comments, please feel free to direct them to [aathwal@sfu.ca](mailto:aathwal@sfu.ca).

Sincerely,

Arman Athwal  
Chief Communications Officer



# EAGLE EYE SYSTEMS

## DESIGN SPECIFICATIONS

---

# Eagle Eye Tracker

---

### *Authors*

ARMAN ATHWAL (aathwal@sfu.ca 301226860)

MATEEN ULHAQ (mulhaq@sfu.ca 301250828)

MARTIN LEUNG (mml28@sfu.ca 301233998)

BUD YARROW (byarrow@sfu.ca 301230000)

NAIM TEJANI (naimt@sfu.ca 301235623)

VICTOR YUN (vyun@sfu.ca 301229725)

April 1, 2018

## **Abstract**

The Eagle Eye Tracker represents a novel approach to tackle one of the pre-eminent technological challenges of the decade: object tracking. The device is a portable, automated system that can be placed in any region where a user wants to monitor. While other devices and solutions have limited applications due to environment-specific hardware and software systems, the Eagle Eye Tracker was designed to be a unified solution for any tracking application. This universality is achieved through sophisticated deep learning algorithms and modular hardware design — the combination of which allows for user customization and continual performance improvements.

To deliver on this ambitious promise, the Tracker is more than just an automated device that one places atop a wall. The final product will sport interfaces through which users can customize the functionality of the Tracker through simple parameter selection (e.g. choosing to track Unmanned Aerial Vehicles (UAVs) but ignoring birds). Further, an API will be provided that will allow extensive customization through a user's own object identification models. This is a crucial factor in the Tracker's capability to continuously improve — as users find more applications for the device and generate their own ideas, the Tracker's performance and scope will naturally improve over time without the need for purchasing a new version. To supplement this, an online community will be established to allow users to express their creativity and share their ideas.

Due to the versatility of the product's applications, the hardware and software systems must be flexible enough in their base design to allow for application-specific customization in the future. In this document, we outline the design choices that allow the Tracker to achieve this. The information presented herein will be referenced to the requirements which were defined previously and are attached to this document. As these requirements corresponded to separate stages of the project (Proof of Concept, Prototype, Final Product), as do these design specifications. Further, an appendix is attached which describes our product validation testing procedures and user interface design, as well as an appendix which outlines our plan for carrying this project forward into ENSC 440.



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>System Overview</b>	<b>3</b>
<b>3</b>	<b>Optical Design</b>	<b>4</b>
3.1	Imaging . . . . .	4
3.1.1	Proof of concept . . . . .	6
3.1.2	Prototype . . . . .	6
3.2	Laser . . . . .	7
3.2.1	Prototype . . . . .	9
3.2.2	Product . . . . .	9
<b>4</b>	<b>Mechanical Design</b>	<b>9</b>
4.1	Proof of concept . . . . .	9
4.2	Prototype . . . . .	11
4.2.1	Upper Rotating Stage . . . . .	12
4.2.2	Lower Rotating Stage . . . . .	13
<b>5</b>	<b>Electrical Design</b>	<b>14</b>
5.1	Proof of Concept . . . . .	14
5.2	Prototype . . . . .	15
5.2.1	Power Supply System . . . . .	16
5.2.2	Laser Control System . . . . .	17
5.2.3	Motor Control System . . . . .	18
5.2.4	Overall Control System . . . . .	22
5.2.5	Alternate Motor Control System . . . . .	23
<b>6</b>	<b>Software Design</b>	<b>24</b>
6.1	Proof of Concept . . . . .	24
6.2	Prototype . . . . .	25
<b>7</b>	<b>Conclusion</b>	<b>26</b>
<b>8</b>	<b>Test Plan Appendix</b>	<b>28</b>



## EAGLE EYE SYSTEMS

8.1	Mechanical tests . . . . .	28
8.2	Hardware tests . . . . .	28
8.3	Software tests . . . . .	29
8.4	Integration tests . . . . .	30
8.5	User tests . . . . .	31
<b>9</b>	<b>UI Appendix</b>	<b>32</b>
9.1	Introduction . . . . .	32
9.2	User Analysis . . . . .	32
9.3	Technical Analysis . . . . .	34
9.3.1	Discoverability . . . . .	34
9.3.2	Feedback . . . . .	34
9.3.3	Conceptual models . . . . .	35
9.3.4	Affordances . . . . .	35
9.3.5	Signifiers/Mappings . . . . .	35
9.3.6	Constraints . . . . .	35
9.4	Engineering Standards . . . . .	36
9.5	Analytical Usability Testing . . . . .	36
9.6	Empirical Usability Testing . . . . .	38
9.7	Conclusion . . . . .	40
<b>10</b>	<b>440 Planning Appendix</b>	<b>41</b>
10.1	Introduction . . . . .	41
10.2	Scope/Risks/Benefits . . . . .	41
10.2.1	Scope . . . . .	41
10.2.2	Benefits . . . . .	42
10.2.3	Risks . . . . .	43
10.3	Market Competition/Research Rationale . . . . .	44
10.3.1	Drone Detection . . . . .	44
10.3.2	Bird/Animal Deterrent . . . . .	45
10.4	Personnel Management . . . . .	46
10.4.1	Technical/Design Roles . . . . .	46
10.4.2	External Contacts . . . . .	48
10.5	Time Management . . . . .	49
10.5.1	High-Level System Design . . . . .	50



EAGLE EYE SYSTEMS

10.5.2	Hardware Design . . . . .	50
10.5.3	Software Development . . . . .	50
10.5.4	Contingency Plan . . . . .	51
10.6	Budget Management . . . . .	52
10.7	Funding . . . . .	52
10.8	Conclusion . . . . .	53
<b>11</b>	<b>Glossary</b>	<b>54</b>
<b>12</b>	<b>Appendix: Requirement Specifications</b>	<b>57</b>
12.0.1	Mechanical Requirements . . . . .	57
12.0.2	Electronics Requirements . . . . .	57
12.0.3	Optical Requirements . . . . .	58
12.0.4	Software General Requirements . . . . .	58
12.0.5	Identification Requirements . . . . .	58
12.0.6	Tracking Requirements . . . . .	59
12.0.7	Environmental and Safety Requirements . . . . .	59

## List of Figures

1	Concept diagram . . . . .	4
2	Depth of Field . . . . .	5
3	Hyperfocal Depth of Field . . . . .	6
4	Canon Rebel T3 DSLR . . . . .	7
5	Gaussian Beam . . . . .	8
6	Galilean Beam Expander . . . . .	8
7	Proof of concept . . . . .	10
8	Mechanical System . . . . .	12
9	Upper Mechanical Stage . . . . .	13
10	Lower Mechanical Stage . . . . .	14
11	Power Electronics Schematic . . . . .	16
12	Power hardware flowchart . . . . .	17
13	Laser hardware flowchart . . . . .	18
14	Laser Controller Schematic . . . . .	18



EAGLE EYE SYSTEMS

15	Motor Controller Schematic . . . . .	19
16	Two Motor Controller Schematic . . . . .	20
17	Feedback Motor Controller Schematic . . . . .	21
18	System Controller Schematic . . . . .	22
19	Motor hardware flowchart . . . . .	23
20	Software flowchart . . . . .	24
21	GUI mockup. . . . .	33
22	Gantt Chart . . . . .	49
23	Neural network . . . . .	51

## List of Tables

1	Budget . . . . .	52
---	------------------	----





# 1 Introduction

Imagine you are a corrections officer of a major federal prison. The day is late, the shift is almost over, and the inmates are about to retire to their cells from the outdoor common area within the prison walls. Standing in your guard tower, you monitor the situation below you absentmindedly. While scanning the perimeter, you notice something in the corner of your eye. You motion to investigate, but your judgment gets the better of you and you decide it was most likely a bird. A few seconds later, a low humming sound. You look over your shoulder, and hovering over the center of the courtyard is what appears to be a toy helicopter, carrying something under its frame. Its form takes shape, and your heart stops. You reach for your firearm with honed reflexes, but the AK-47 assault rifle is already falling towards the ground. An inmate stands in the center of the courtyard, arms outstretched at the ready.

Now, imagine yourself a farmer. The year has been hard. The rains have been sparse and the air too dry, but worst of all — the pests. You can work around the unfavorable weather, but the pests persist and multiply. As you stand on your porch and look over the berry crops, you painfully reminisce of all the measures attempted and all the money spent to combat them, to no avail. This year, a quarter of the crop will be lost. Next year, even more. The bird snacking on the eye of your scarecrow is just salt in the wound.

Finally, after failed careers as a prison guard and farmer, you've found your calling as a stunt driver in Hollywood. Business is booming, as mindless action films have taken over the industry. Today's job is a high-speed car chase for the next blockbuster. You've done the stunt perfectly — four times. The camera crew has been unable to get the shot of your moving vehicle due to its high speed. This same issue has been plaguing every one of your films, and you nervously think of the risk involved in performing these dangerous stunts repeatedly. As you enter the car for the fifth time, you can't shake the feeling that the debilitating issues that have followed you across your impressively diverse career have something in common.

What if you were right? What if these seemingly unrelated issues were connected by the same, subtle problem? What if the prison guard knew about the aerial



## EAGLE EYE SYSTEMS

threat in advance, the farmer could protect their crops from the comfort of their rocking chair, and the stunt camera crew barely had to move a finger — all thanks to the same device?

This is no longer a question. Eagle Eye Systems has created an elegant solution to all these problems and more — the Eagle Eye Tracker. The Tracker can identify any object of choice and track it through time and space. This sounds like a simple enough concept — but the Tracker’s complexity lies in its ability to perform more than one specialized task without sacrificing accuracy or efficiency. The tracker’s flexibility is made viable through two main design choices — deep learning and modularity.

Take our inattentive prison guard as a not-so-fictional example [9]. To prevent against such threats, the prison institution could purchase and install a sophisticated aerial defense system, which may even include projectile-based counter measures. While this would certainly be effective, it would be obscenely costly and would become less effective over time as technology progresses and offenders become more clever. Alternatively, the prison could purchase a set of Trackers, set them to look for incoming threats, and place them in strategic locations around the perimeter. When a threat is detected, security personnel are alerted well before the threat becomes critical and have enough time to take appropriate action.

Our frustrated farmer could use the same device — with a twist. The Tracker’s modular design allows our farmer to attach a low power laser to its rotating stage, place one or more of these modified Trackers around their berry field, and set the system’s parameters to bird deterrence. The same Trackers that were placed along a prison perimeter are now operating on a new sub-routine that is looking specifically for birds, and will shine the harmless laser at the pests to scare them away — a proven and humane method of agricultural pest control. Similarly, the stunt camera crew could equip a high-speed camera to the device, set the appropriate object identification parameters, and voila — a hands-free method to get that perfect high-speed shot.

These examples serve to illustrate the Tracker’s abilities at a high level and how these various functionalities are supported through its design. In this document, we discuss this design in specific detail in terms of its sub-systems — electrical, mechanical, and so on. Further, we distinguish between the functionalities and



designs of our different product versions. These design specifications — along with the attached appendices which outline user interface design and plans for moving into the next stage of the project — aim to describe at both low and high levels how the ambitious vision of the Eagle Eye Tracker will be turned from concept into realization.

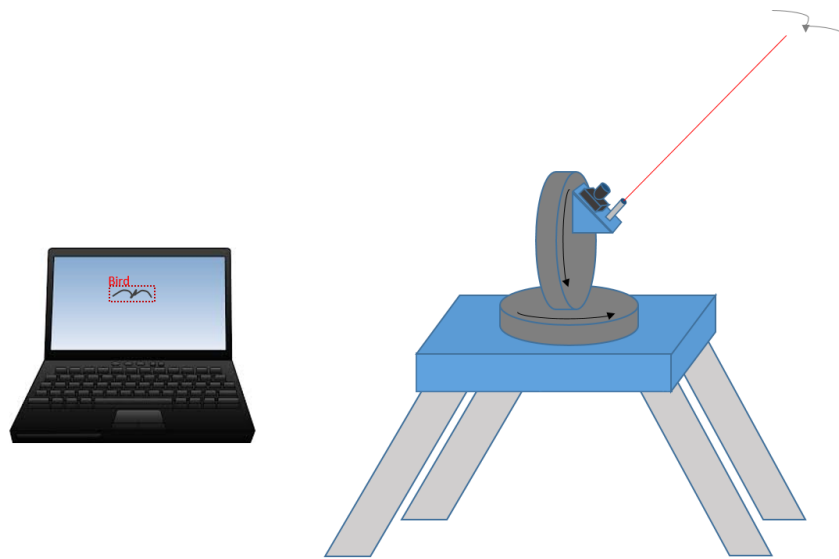
## 2 System Overview

The Eagle Eye Tracker will provide users with the capability to autonomously identify and track a target object of their choosing. A neural network will be configured to recognize several objects, which will grow over time. This will give the system the ability to distinguish between target and non-target objects, so that the Tracker can make the decision to ignore said objects. A major advantage of the neural network is that its database of recognizable objects can be updated over time. Thus, the Tracker can improve without the need for new versions of the entire product.

Once the system is configured to identify desired objects, the user will install the system in a region where it has a clear field of view for optimal surveillance. The Tracker will then run in idle mode, where it pans the imaging system back and forth over the desired field of view. Once a targeted object enters the field of view, the recognition algorithm will identify the object and provide a bounding box outlining the object in the video feed. The tracking algorithm will then activate and direct the mechanical system to move to center the targeted object in the camera's field of view, and keep it there through adjustments done in feedback to the object's motion. [Figure 1](#) provides a general illustration of this scenario.



EAGLE EYE SYSTEMS



**Figure 1:** Conceptual diagram visualizing the Eagle Eye Tracker, which looks not entirely unlike our intended finished product.

If set up to do so, the system will enable a laser pointer to mark the target being tracked as shown in [Figure 1](#) above. Simultaneously, the Eagle Eye Tracker will be programmed to alert the user of a detection, so that they may respond as necessary as soon as possible. The Tracker will also log detections and relevant metrics such as time and tracking duration of objects to provide maximum surveillance information to the user, upon their request. Please refer to the very end of this document for our requirement specifications.

## 3 Optical Design

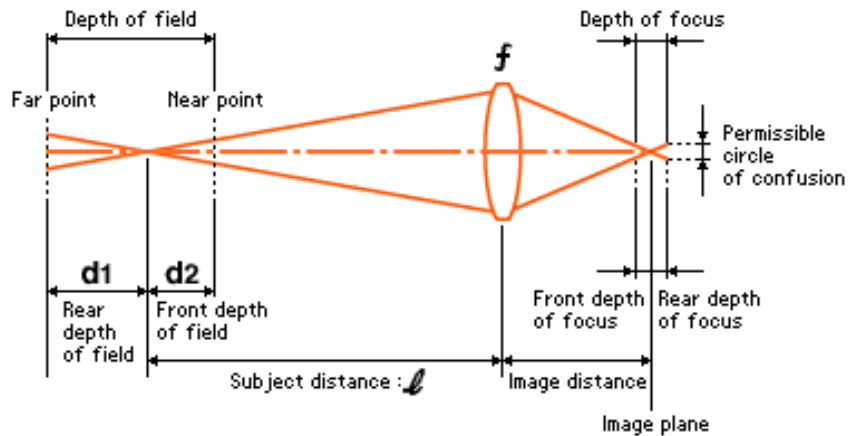
### 3.1 Imaging

An important aspect of imaging is keeping the target object in focus as described in REQ 1.3.1-BC. This can be done in two ways. The first is through dynamic focusing, by which we sweep focus over different ranges in real time. However, this may not be required and could add unnecessary complexity. The alternative is



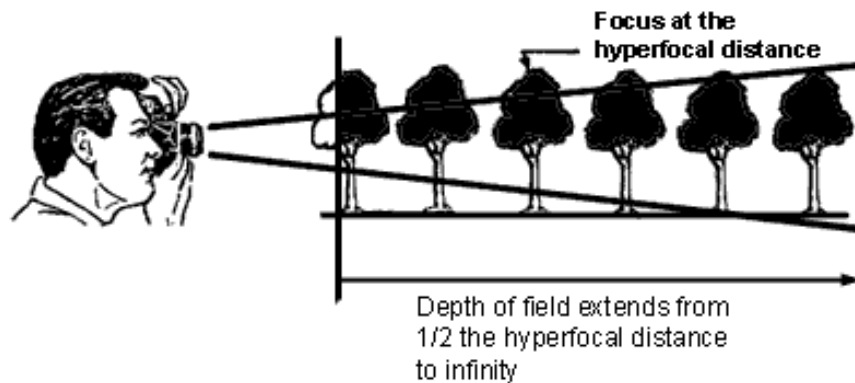
## EAGLE EYE SYSTEMS

to used a fixed focal point. This will create a range where objects will be in focus to a certain degree. This range is known as the depth of field and extends from the near point to the far point. The depth of field arises from the depth of focus, given by the permissible circle of confusion. The permissible circle of confusion is defined based on the minimum acceptable resolution, for max resolution is based on the sensor pixel spacing. This is illustrated in [Figure 2](#).



**Figure 2:** Conceptual diagram visualizing depth of focus and depth of field.

“When the lens is focused on the hyperfocal distance, the depth of field extends from half the hyperfocal distance to infinity.” - Photography, Phil Davis, 1972. By focusing at the hyperfocal distance the depth of field is maximized, which is ideal for the Tracker. This is depicted in [Figure 3](#) below.



**Figure 3:** Conceptual diagram visualizing the focusing camera at the hyperfocal distance to maximize the depth of field.

### 3.1.1 Proof of concept

The proof of concept was designed as a platform on which the software algorithms could be implemented and tested. Thus, its mechanical stage cannot support a significant weight, and a simple webcam is used as the imaging modality. Hence, the webcam's built-in auto focus is used, which is needed since in this case the webcam operates primarily on objects in close proximity to the system. With this setup, the system should be able to function as intended for objects within 1 to 10 meters from the camera.

### 3.1.2 Prototype

To improve resolution and object identification capability, a DSLR camera will replace the webcam as the prototype imaging modality. The model used is the Canon Rebel T3 which can be seen in [Figure 4](#). This camera includes a 12.2 megapixel CMOS (APS-C) sensor, digital gain ranging from ISO 100 to 6400 for shooting from bright to dim light. The specifications of this camera should meet or resolving requirements stated in REQ 1.3.2-ABC.

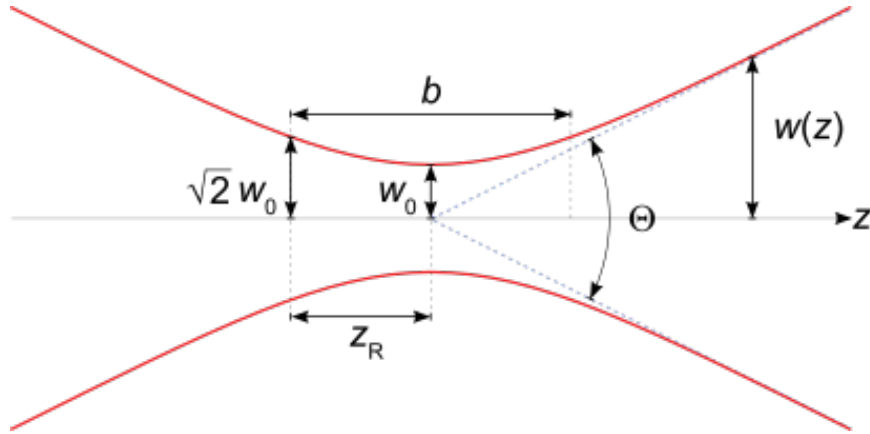


**Figure 4:** Image of Canon Rebel T3 with 50mm to 250mm telephoto zoom lens.

A telephoto lens will be used to ensure that the objects of interest are large enough in the field of view. Using a 250mm focal length lens with the Rebel T3 as seen above in [Figure 4](#), we can determine the depth of field. Using an f number of 5.6 and setting the focus at 350m, we create a depth of field between 200m and 1000m, which is ideal for long range object surveillance. For objects closer in proximity, auto focusing may be required. With this in mind we should be able to meet our long range resolution requirements stated in REQ 1.3.1-BC.

## 3.2 Laser

Laser modules may be added to the system for different applications. When choosing the right laser for a particular use, it is important to not only consider the application but also keep in mind the properties of the light emitted from the laser. High quality lasers output a beam modeled as a Gaussian function. Gaussian beams behave differently than regular light beams.

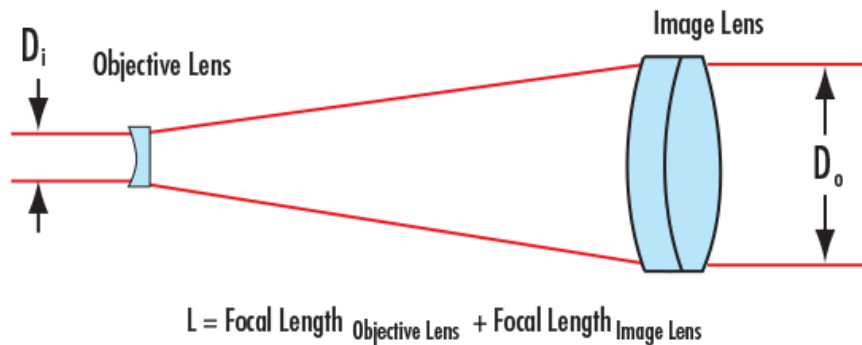


**Figure 5:** Conceptual diagram visualizing the spreading of light from a laser as a Gaussian beam.

An important property of the Gaussian beam is the range in which it stays fairly confined and does not spread out significantly, this is called the Rayleigh range,  $z_R$ , as seen in [Figure 5](#). The functional form is seen below with  $\lambda$  being the wavelength of light.

$$z_R = \frac{\pi w_0^2}{\lambda} \quad (1)$$

The square of the spot size,  $w_0^2$ , is directly proportional to the Rayleigh range, so a larger initial spot size is required to keep the light confined over a larger distance. To allow for high power transmission and compactness a Galilean beam expander ([Figure 6](#)) may be used to enlarge the initial spot size to extend the Rayleigh range.



**Figure 6:** Schematic of Galilean beam expander





### **3.2.1 Prototype**

One potential reason to add a laser module is to simply mark the target object when tracking it. This is what will be done for the prototype as required by REQ 1.3.4-BC. For our intended purpose, a simple laser pointer will be used. This will most likely be a diode laser emitting light in the visible spectrum in common colors of red or green, optimizing color for visibility in daylight. The power will be kept low enough to avoid risking accidental eye injury by using Class 1 or 2 lasers.

### **3.2.2 Product**

In the case of pest deterrent applications, a laser will be used to scare the birds away. The laser will therefore be in the visible spectrum, optimizing color to match the sensitivity of bird eyes to the color. Furthermore, the laser power will be kept below the eye damage threshold since this method of deterrence is meant to only scare birds and not cause any physical harm.

For drone interception and defense applications, a high-power laser weapon may be used if deemed absolutely safe and necessary by an appropriate institution as stated in REQ 1.3.5-C. In this case, the laser would require immense power over 10 kW. However, this could be delivered in a short pulse so continuous energy consumption need not be so large. One high powered laser type is the CO<sub>2</sub> laser, which produces infrared radiation. Many modular additions would be required to support such a laser system, and would only be considered in strict collaboration with recognized authorities.

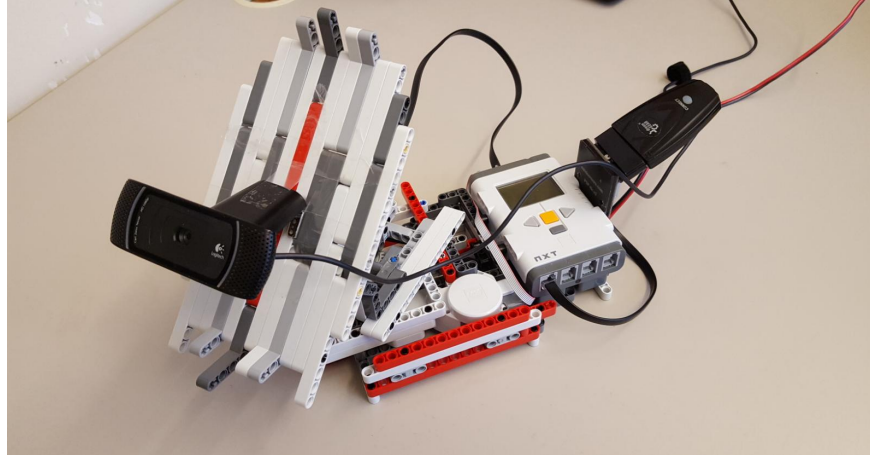
## **4 Mechanical Design**

### **4.1 Proof of concept**

For our proof of concept, which we will present on April 9, we will be using a LEGO Mindstorms model with a camera attached. Our design can be seen below



in [Figure 7](#).



**Figure 7:** Our proof-of-concept Eagle Eye Tracker.

Two geared servo motors (REQ 1.1.1-ABC) were used in the design because they provided the best accuracy and power out of other options in the LEGO Mindstorms toolkit, while also having encoders built in to them to monitor position.

The lower stage was designed to have the motor solidly fixed in place with the axle transferred through a gear ratio to the upper stage base to increase accuracy and torque. The lower stage was also designed such that the upper stage base could smoothly slide over the lower stage while providing a solid foundation and taking pressure off the motor axle.

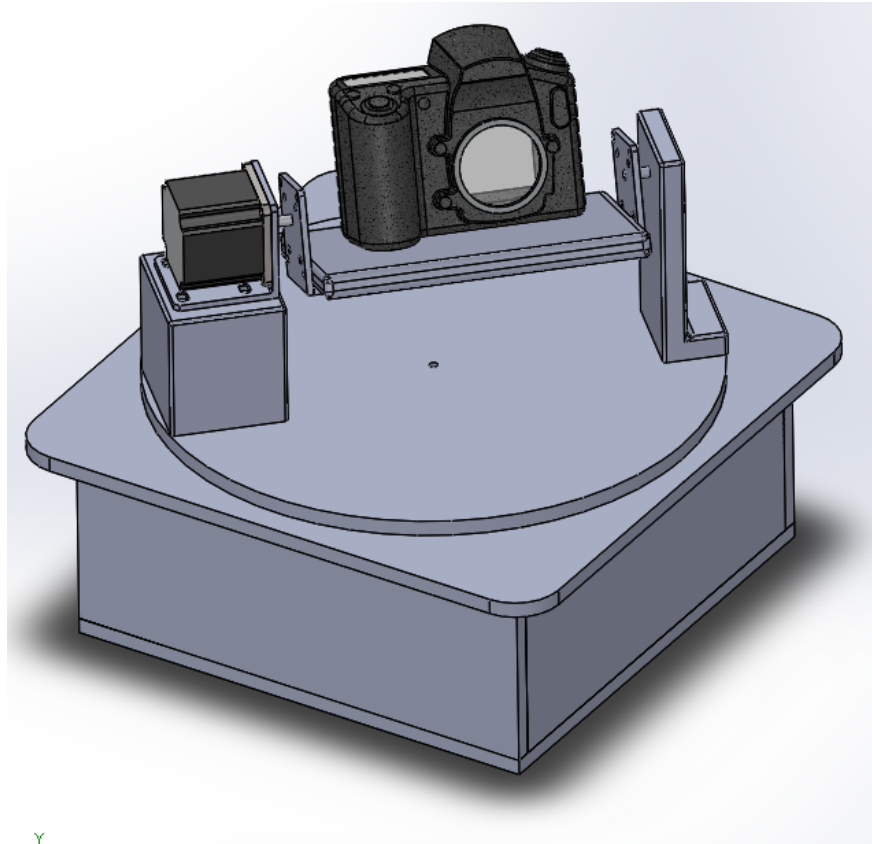
The upper stage was designed to have a large mounting surface with at least 90 degrees of rotational freedom. The motor axle was also transferred through a gear ratio to add clearance and increased power for the mounted load. However due to the low strength of the motor and clearance distance the holding torque the motor can provide is only enough to support the load of a web cam. This is one aspect that the prototype will be designed to improve dramatically in order to have a higher resolution camera with a decent zoom lens mounted. Another drawback of the Mindstorms design is that between the motors and gears there is a significant amount of wiggle leading to reduced accuracy.



## 4.2 Prototype

Extensive time has been spent on the mechanical design in order to ensure our prototype meets our standards. A comprehensive SolidWorks assembly model has been constructed. It includes two NEMA-23 motors along with their respective rotating stages. Much thought and consideration went into the design of each component to ensure precise functionality. For actuation of the two rotating stages, stepper motors were chosen because of their precise resolution and possibility of microstepping. In full step operation, the chosen stepper motor has a step angle of  $1.8^\circ$ , giving 200 steps for an entire azimuthal rotation, and 100 steps to sweep the entire angular plane. Furthermore, we plan on microstepping the motor, reducing the step size by a factor of at least 16. A model for our tracker is shown below in [Figure 8](#). The size of the base is 37 by 37 centimeters and its height is about 35 cm.

This design is broken up into two major components, each with its own motor. These components are the upper plate, which rotates in the angular plane, and the lower plate, which rotates in the azimuthal plane. These two degrees of freedom are necessary for the camera's field of view, within the required distance, to span all space as required by REQ 1.1.1-ABC. We took careful considerations in determining the size of the hole in which the motor's D-shaft is placed. Making it the same size as the shaft will not allow for smooth assembly. The diameter of the motor shaft is 6.35mm and therefore, we have given the holes in the mounting bracket and the rotating plate a diameter of 6.37mm. We plan to use aluminum for most of our manufacturing because it does not rust as required in REQ 3.0.4-BC. It is also easy to machine and cost-efficient.



**Figure 8:** SolidWorks model of Eagle Eye Tracker's complete mechanical system.

#### 4.2.1 Upper Rotating Stage

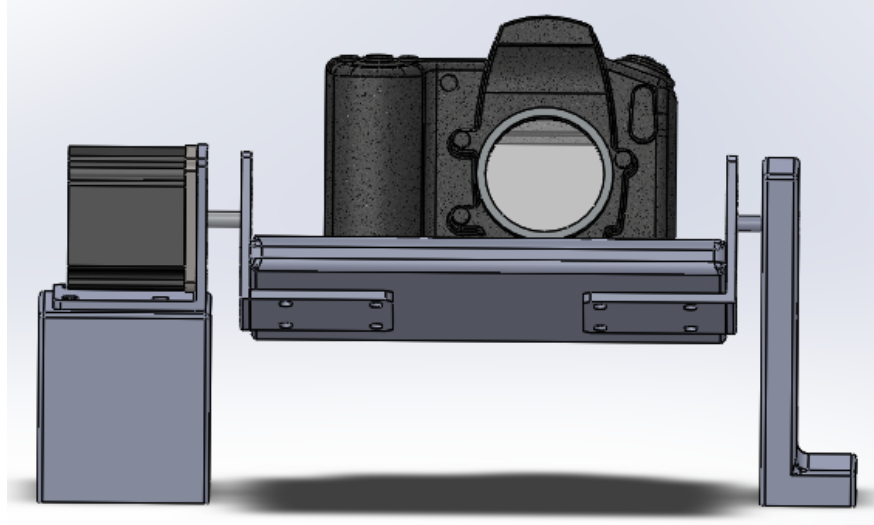
This part of our product will rotate the camera in the angular plane. The local coordinate system will locate the object of interest and provide coordinates in the camera's frame of reference. This stage's motor will be utilized if the local y-coordinate is nonzero. In other words, it will adjust the height of the target. The following formula gives the arc length spanned by each step for the upper stage in terms of our microstep factor.

$$a = \frac{0.01}{f} \cdot r\pi \quad (2)$$

where  $a$  is the arc length spanned,  $f$  is the microstep factor, and  $r$  is the object distance. This means an object 10 m away can be detected with only a 2 cm angular



resolution. Our upper rotating stage design is seen below in [Figure 9](#).



**Figure 9:** SolidWorks model of Eagle Eye Tracker's upper stage with mounted DSLR camera.

In order to minimize torque, the rotation axis has been placed approximately at the center of mass. This may be modified depending on the weight distribution of the camera used. The design for modularity of the camera stems from REQ 1.1.5-BC.

#### 4.2.2 Lower Rotating Stage

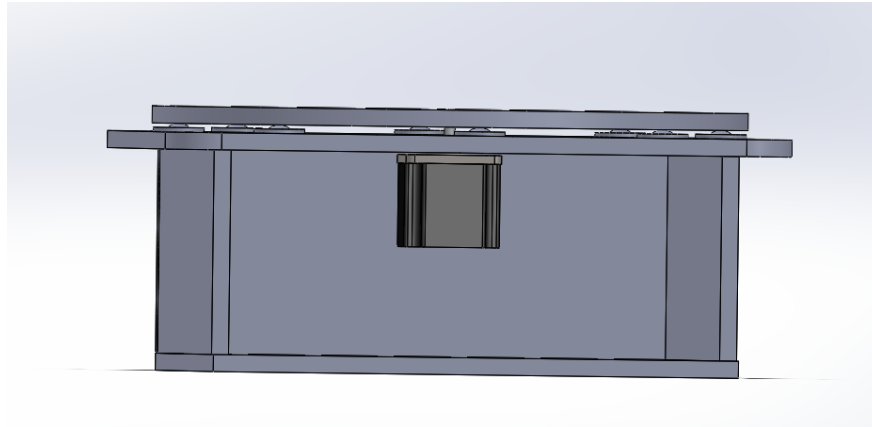
The lower rotating stage rotates in the azimuthal plane. Its corresponding motor will activate if the local x-coordinate is nonzero. Both the rotating plate and the enclosure cover need to be thick enough to support our DSLR camera as well as a laser without bending. We have decided that a thickness of 1 cm is sufficient. The following formula gives the arc length spanned by each step.

$$a = \frac{0.01}{f} \cdot r\pi \sin \theta \quad (3)$$

As in the analogous equation for the lower rotating stage,  $a$  is the arc length spanned,  $f$  is the microstep factor and  $r$  is the object distance.  $\theta$  is the angle of the upper



rotating stage relative to the vertical (in radians). Our lower rotating plate design is shown in [Figure 10](#).



**Figure 10:** Solidworks model showing inside Eagle Eye Tracker's lower stage.

There will also be ball bearings below the circular plate to reduce the weight of the motor shaft. These bearings are placed farther out at the edge of the rotating plate to not place too much unidirectional force from the weight of the upper stage at the edges of the plate.

## 5 Electrical Design

### 5.1 Proof of Concept

The hardware of the Eagle Eye Tracker proof of concept was constructed using LEGO Mindstorms NXT, and it primarily serves as a way to test the tracking software. The software controls the movement of two geared LEGO Mindstorms NXT servo motors, which differs from the stepper motor-driven system in the prototype. The servo motors used in the proof of concept have a built-in rotation sensor which allows the software to keep track of the position of the motor. Compared with stepper motors in the prototype, the servo motors in the proof of concept have significantly less torque. Driver limitations in the servo motors mean that torque



should not exceed 15 N cm for too long. As a result, the load for the prototype is a Logitech webcam, which was chosen for its light weight and functional video quality.

Originally, the LEGO Mindstorms NXT was powered by six AA alkaline batteries. However to eliminate the cost of replacing all the batteries when depleted, the batteries were placed with an AC adapter that matched the voltage and current requirements of the LEGO Mindstorms kit. The peak current of the kit is 300 mA, while the average is around 200 mA. Given that the fitted AC adapter has a current limit of 400 mA, it will be sufficient and appropriate.

## 5.2 Prototype

The driver chosen must be able to drive a bipolar stepper motor and perform microstepping operations. For prototyping, the Allegro A5984 chip provides a cost-effective and relatively simple way to drive the motors. If it is discovered that 2 Amps per phase is not sufficient for driving the load, an alternative is planned which involves replacing the A5984 driver with the Trinamic TMC249A driver. This driver is not limited to 2 Amps per phase, and can readily supply the 2.8 Amps of current that the motor itself is rated to handle with design of an external H-bridge. If further power is necessary, the current motor may be upgraded to better suit the requirements of the Tracker's load.

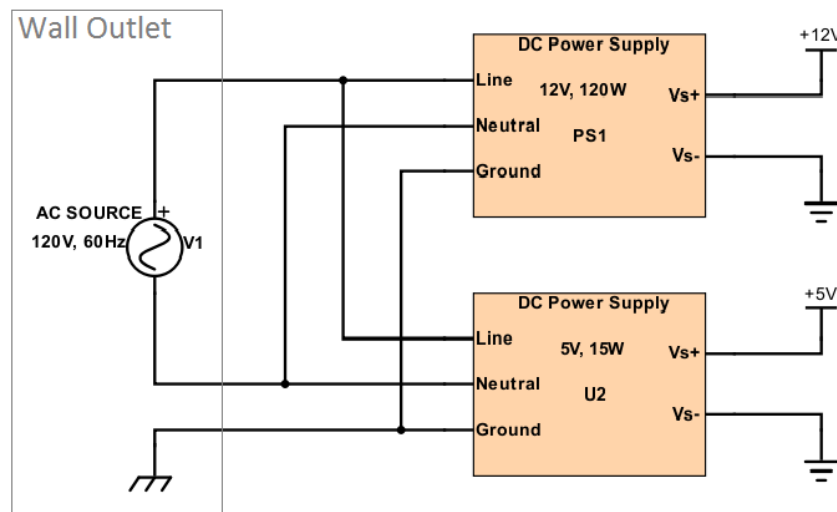
Onto comparing the Serial Peripheral Interface (SPI) with General-Purpose Input/Output (GPIO), standards used for the Trinamic and Allegro drivers, respectively. For chips with few pins and simple data, GPIO is a breeze to wire. Conversely, for chips with many pins, SPI is the dominant choice due to it usually requiring much fewer wires since all the data is travelling serially through the wires.

It is vital that motors for the top and bottom stages are powerful enough to rotate the two stages with ease.



## 5.2.1 Power Supply System

The power of the system is important to ensure proper operation of all electronic components. For our prototype we will take power from the grid as seen in [Figure 11](#). This will allow us to use as much power as required for proper operation.



**Figure 11:** Schematic illustrating power source, and supply voltages.

Firstly, power is taken from a standard US/Canada wall outlet running at 120V AC with a frequency of 60Hz. The source power is then converted to the DC levels required by our system using two DC power supplies.

A 12V power supply will power the stepper motors. Since each motor has two coils which may consume up to 2.8A of current at 2.5V, a total of 11.2A is required at maximum power. To avoid overloading the power supply, it will be specified to have at least 120W. The driver itself will take in the 12V power supply voltage and create the waveforms necessary to move the motor as required, and consuming the current needed.

Meanwhile, a 5V power supply will run the microcontroller, laser, and other electronics. A Raspberry Pi may be used in place of the microcontroller. In this case, it is recommended to have a 5V/2A power supply. To power the other elec-

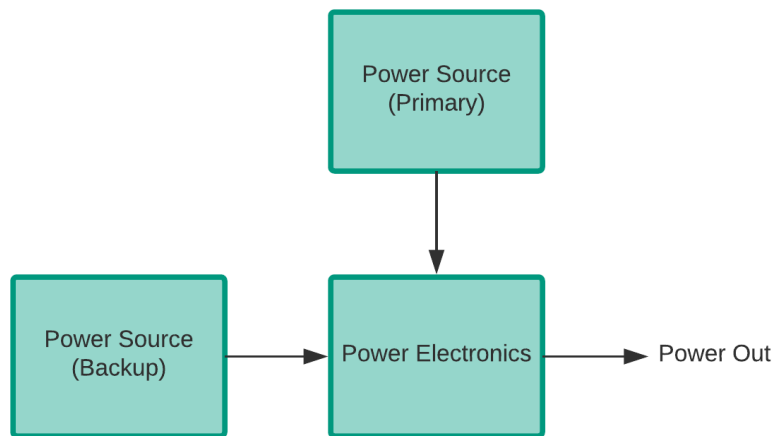




## EAGLE EYE SYSTEMS

tronics and to avoid overloading the power supply, an additional amp is provided, specifying the power supply at 15W.

Extending the system to a marketable product, it is imperative to ensure our system has an uninterruptible power supply, as seen in [Figure 12](#) and as required by REQ 1.2.3-C. This will ensure powering off-grid for use in remote locations, such as on a farmer's field. Also if there is a disruption in the grid power supply, such as a power outage, the system will continue to run.



**Figure 12:** Flowchart illustrating power source. We will use two sources for reliability and portability.

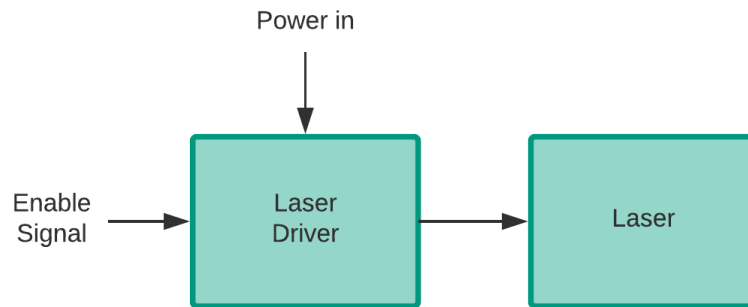
A primary power supply such as the wall outlet or solar panels would be used as required by REQ 3.0.1-C. A secondary backup supply, such as a rechargeable battery, would be used to avoid the system going offline due to failure of the primary source as needed in REQ 1.2.4-C.

### 5.2.2 Laser Control System

The addition of a laser in the Prototype will also us to demonstrate the tracking with a laser continuously marking the tracked object. The laser will only be turned on

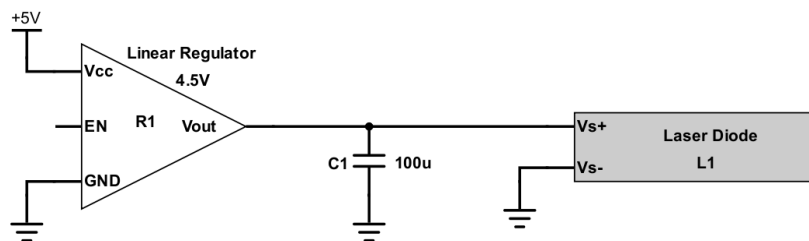


when tracking the targeted object to save energy or to prevent accidentally shining the laser at people or other unintended objects.



**Figure 13:** Flowchart illustrating laser control system.

This control system is illustrated above in [Figure 13](#). In our prototype we will use a linear voltage regulator with an enable pin as seen in [Figure 14](#). The voltage regulator will set the required voltage to the laser pointers supply voltage, 4.5V in this case. The microcontroller will control the enable pin which will either supply power to the laser when tracking or stop power when not tracking as specified by REQ 1.2.2-BC.



**Figure 14:** Schematic of laser control system.

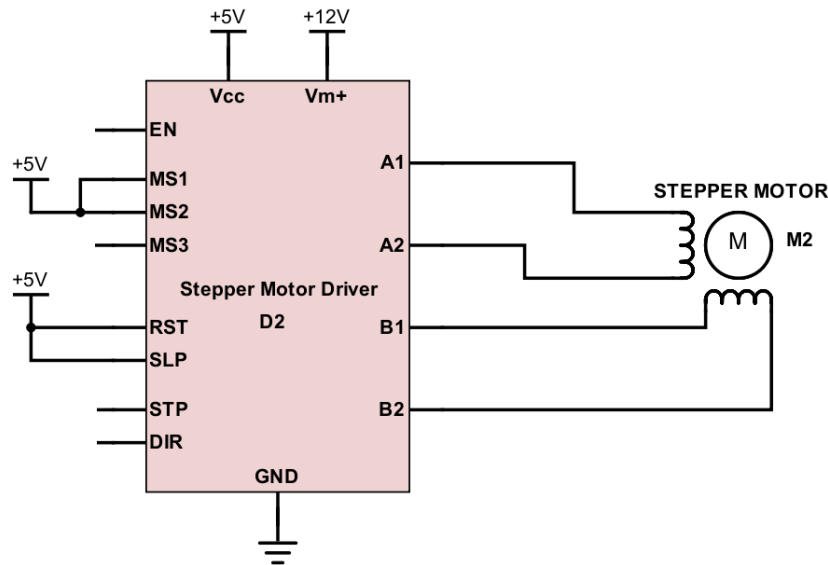
### 5.2.3 Motor Control System

The motor was tested with the Allegro A5984 driver as seen in [Figure 15](#). This driver is capable of driving stepper motors to a maximum of 2 Amps per phase.



## EAGLE EYE SYSTEMS

This is to accommodate loads (cameras) of varying masses, and to ensure that the product is not unreasonably restricted to lighter loads. The power to drive these loads comes from the 12V power supply and is connected to  $V_{m+}$ , while the 5V supply powers the electronics in the chip.



**Figure 15:** Schematic of motor control system.

The Allegro A5984 is capable of 32 step microstepping, which reduces the step resolution of the motor to  $0.05625^\circ$ , meeting our angular resolution requirement in REQ 1.1.2-BC. The level of microstepping is set by pins MS1, MS2, and MS3. If MS1, MS2, and MS3 are all set high the driver is set to one eighth microstepping. If MS3 is low, the driver is in 32 step microstepping. Thus in the configuration in [Figure 15](#) the microstepping can be dynamically changed to balance speed and precision as required by REQ 1.1.3-BC and REQ 1.1.4-BC.

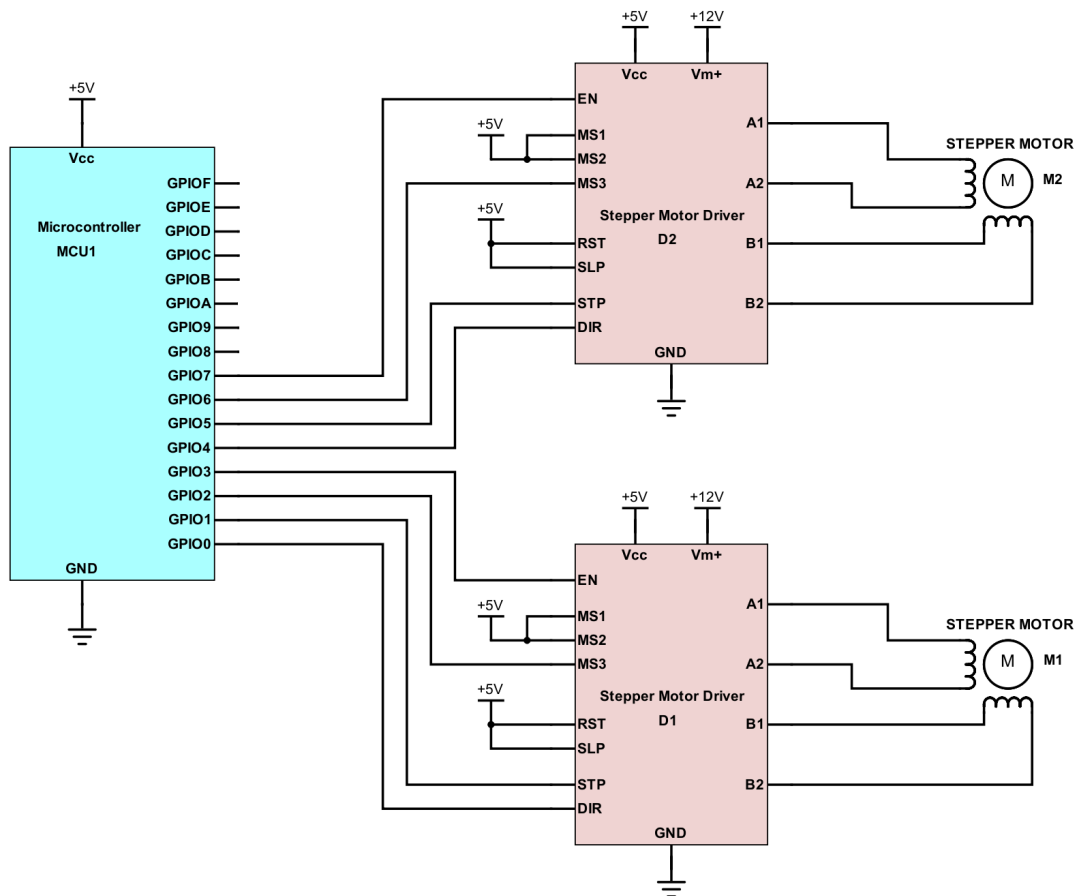
Alternatively the speed of rotation can be controlled by the STP pin, whose rising edge triggers the motor to move one step, whether it be a full step or a microstep. This will have to be controlled dynamically in ramp-ups and ramp-downs for smooth running. The DIR input controls the direction in which the motor is to be driven. The enable pin, EN, sets whether or not power is delivered to the H-bridge driving stage.



## EAGLE EYE SYSTEMS

The A5984 is a chopper driver, effectively resulting in higher torque at higher speeds compared with direct drivers. When driving large loads, the driver needs to supply more current, making the A5984 heat up. To mitigate this, the chip will be soldered onto a PCB such that the metal plate on its underside contacts the copper layer on the PCB.

The Eagle Eye Tracker requires two motors to control each angular degree of freedom. Connecting two A5984 drivers and two stepper motors to a microcontroller realizes this as seen in [Figure 16](#)



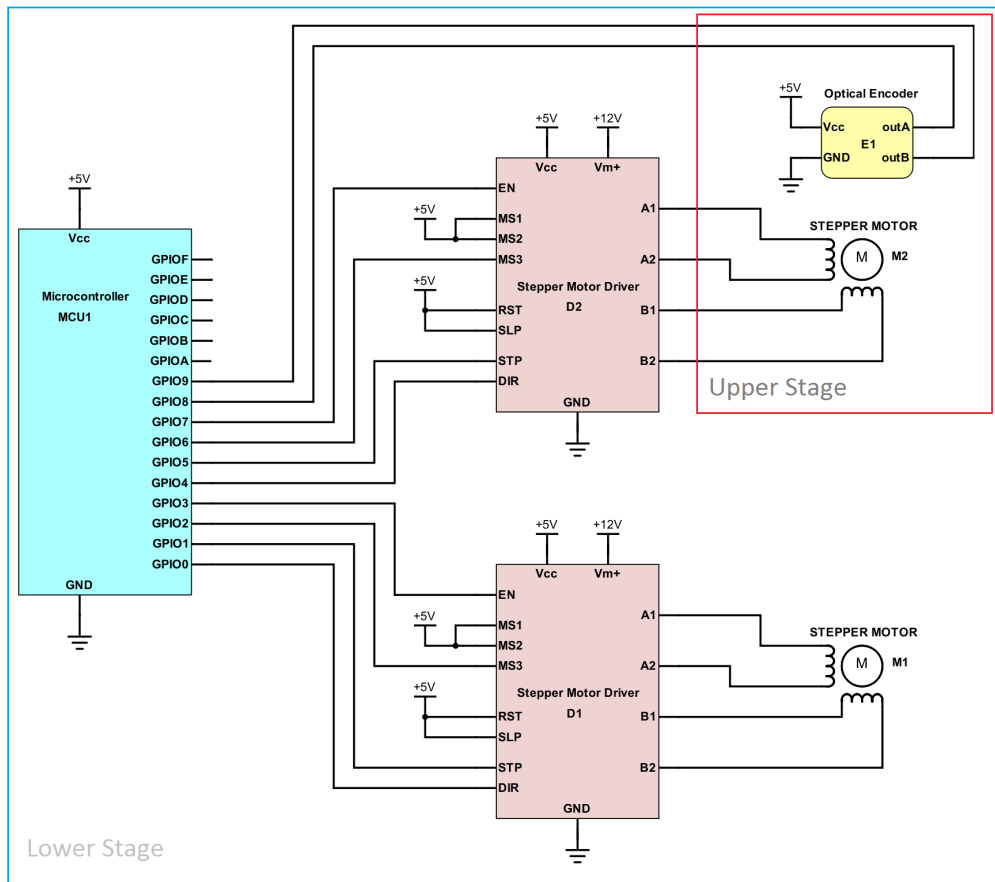
**Figure 16:** Schematic of two motor control system controlled through microcontroller.

Knowledge of the absolute polar angle is part of the software requirements to ensure accurate tracking. To satisfy this requirement, an optical encoder is



EAGLE EYE SYSTEMS

used to measure the angular change in the position of the upper stage's motor. This information is transferred to the microcontroller through two digital outputs. The outputs are square waves which produce a wave period for every 1/300 of a revolution, producing 300 wave periods per full rotation. The outputs waveforms are offset by a phase of either  $+90^\circ$  or  $-90^\circ$  depending on the direction of rotation. This information will need to be collected and stored in the microcontroller to keep track of the absolute polar angle. Incorporation of the encoder into the system can be seen in Figure 17.

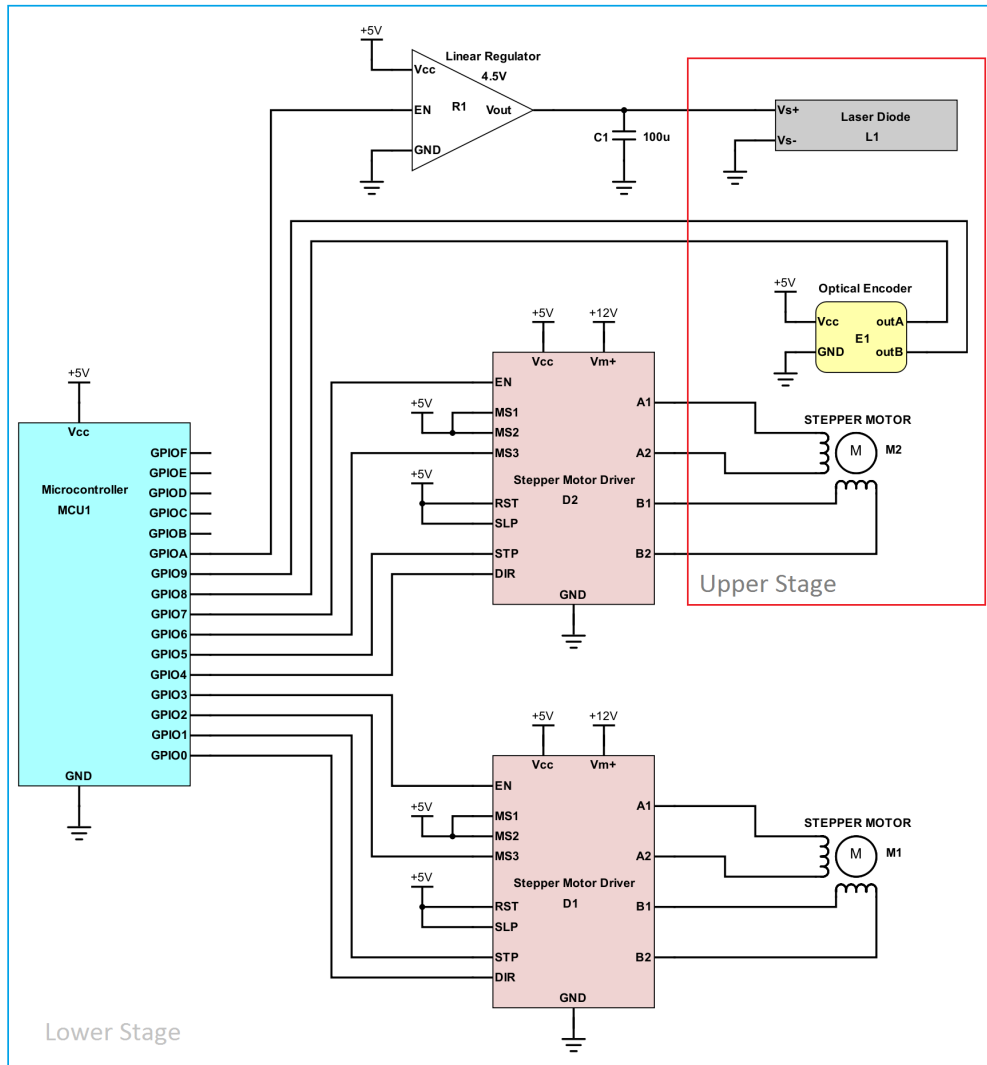


**Figure 17:** Schematic of full motor control system with addition of an optical encoder to track absolute angle of the motor on the upper stage.



### 5.2.4 Overall Control System

Incorporation the laser control system with the motor control system gives us the full electronic control system, which may be seen in [Figure 18](#)

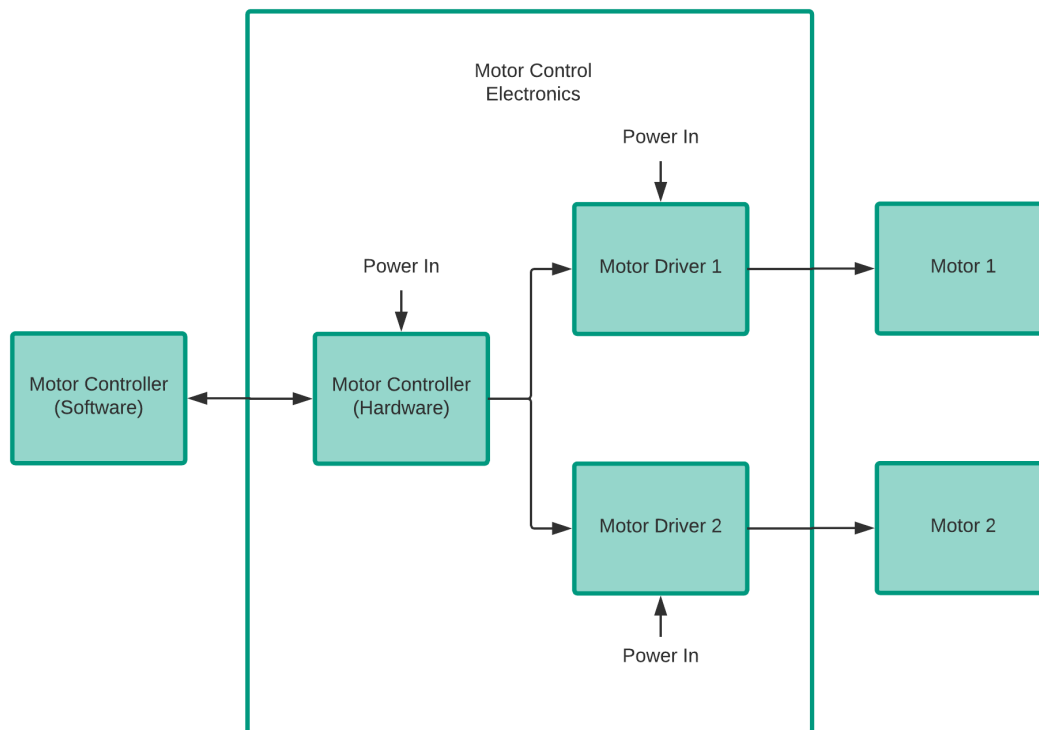


**Figure 18:** Schematic of full electronics control system for control over laser power, and both motor positions, with feedback to track position.



### 5.2.5 Alternate Motor Control System

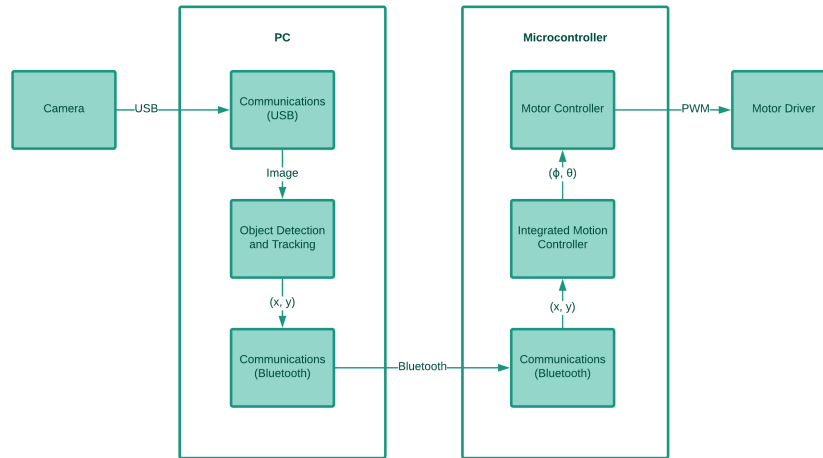
Should higher resolution or power requirements be needed, Trinamic motor control electronics can be used as depicted in the [Figure 19](#). Taking some of the effort from the software team, we can use the TMC429 controller which will take in position and velocity commands and produce required waveforms. The waveforms will be read in using the Trinamic TMC249A motor driver, which offers finer resolution and lower vibration with 256 step microstepping which may be necessary to meet REQ 1.1.4-BC. Additionally, it is worth noting that the Trinamic driver does not include the H-bridge circuit. Therefore, an external H-bridge can be designed using appropriate power MOSFETS to meet the motors' power requirements. Being external, the H-Bridge would not be restrained by the size of the chip, and heat dissipation is less of an issue.



**Figure 19:** Flowchart illustrating motor control system.



## 6 Software Design



**Figure 20:** Flowchart illustrating flow of information throughout the required software modules.

### 6.1 Proof of Concept

In the proof of concept version, the Eagle Eye Tracker currently does not support recognition of a specific object. The present codebase uses OpenCV's `cv2.goodFeaturesToTrack()`, which is based on an algorithm called Shi-Tomasi Corner Detection. Each frame of the live video from the webcam is processed and corners that meet a specific quality level in the image are marked [2]. Then, we choose to keep one of these corners as a point to track. We use the Lucas-Kanade optical flow algorithm `cv2.calcOpticalFlowPyrLK()` in order to compute its position in the next frame; this will be kept and improved in the prototype to meet REQ 2.3.1 - 2.3.4.

Communication between the PC and NXT module is done using either Bluetooth or USB. Bluetooth was the initial method that Eagle Eye Systems had went with, but ultimately discovered issues with regards to latency. Hence, USB is the recommended choice in order to meet REQ 2.1.1. Making use of the NXT-Python library, the communication methods were set up, which included functions to locate and connect to the brick, as well as to read and write data.





Control of the servo motors is performed by the NXT brick. The program is written in a language called NXC and executed directly on the module. The NXC API provides a function, `OnFwd()`, that drives the desired output port containing the motor in either a forward or reverse direction. As the input coordinates are calculated from the tracker module and continuously sent to the NXT brick through USB, they will be converted to an angular position in terms of phi and theta. From these, the power values are calculated and used to drive the motors.

## 6.2 Prototype

Object recognition in our prototype will be much more sophisticated than in the proof-of-concept, and will involve the use of deep learning. Specifically, we will be using a real-time detection algorithm called You Only Look Once (YOLO) developed by Joseph Redmon. This algorithm is consistently ranked in the top with others of its kind for its accuracy and speed; version 3 boasts a mean average precision (mAP) of 57.9 on the COCO dataset when the intersection over union (IoU) is 0.5 or greater [3]. We are confident that this algorithm will help meet our product's requirements of object recognition, specifically REQ 2.2.4. Without diving into implementation details of YOLO itself, our team will be able to utilize this by installing and training it on our own dataset using Darknet, an open source neural network framework that is provided by the creator. First, we'll have to gather and label a sufficient dataset of the specific object that we want to track. A useful tool for this task may be BBox Label Tool to draw our ground-truth bounding boxes, which must be accurately done to ensure a robust model. We would ideally want a few hundred correctly labelled images in order to meet REQ 2.1.3. We will train the model for as many iterations as it takes until the average loss remains stable to ensure REQ 2.2.1 is met. This may take thousands of iterations; fortunately, the weight files can be re-used to continue where training left off and also retrieved from an earlier point in case overfitting occurs [4].

The prototype will also consist of an overhauled design for the motor and motion control systems in that the Lego Mindstorms NXT kit will no longer be used; this also means that all the current NXC codebase and modules related to communicating with the NXT brick will have to be rewritten to support our new



system. However, many of the interfaces in the current code can be reused. The improved motor control software will be run on an Arduino and a PC, which may be replaced with a Raspberry Pi; direct control of the motor drivers will be done by the former while more complex workload, such as motion control calculations, will be offloaded to the latter. Further improvements will have to be made to the implementation of functions while they are ported over from the prototype, including the use of feedback control and prediction algorithms to provide even better results. The Arduino will be communicating with the TMC249A motor driver via the SPI interface or the A5984 via GPIO, which are the protocols supported by the drivers. Arduino provides an SPI library for all functionalities related to using SPI, and will act as the master device.

The diagram in [Figure 20](#) shows the initial flow of information that we had planned for the proof-of-concept. For the prototype, much of the design remains the same except a serial connection will be used in place of Bluetooth for communication between the PC and microcontroller to reduce latency.

## 7 Conclusion

The Eagle Eye Tracker represents a new leap in multipurpose technology. While versatility is indeed the goal, precision and efficiency cannot be sacrificed. After all, a single device that can perform multiple tasks less effectively than several devices performing a single task is not worth very much at all. The Tracker's design as described in this document takes a bottom-up approach. Instead of trying to solve multiple problems by cramming separate independent functionalities into one device, we designed the simplest base model we could imagine that had the most general (yet precise) functionality, and built upwards. Following this design model, we envisioned the potential additions that could be made to the device to open doors to new applications — without ever altering the core of the system. Thus, by designing the fundamentals of the device as precisely as possible, it could perform a wide variety of tasks with only minor changes to its superficial composition.

Our elegant two-motor and camera system, combined with our deep learning recognition and customizable identification models not only ensures this core



## EAGLE EYE SYSTEMS

functionality, but provides a path on which the device may evolve naturally as users become more familiar with it and test its capabilities. This in turn will improve the system as we receive feedback and enthusiastic users develop their own models to apply on the device. We hope to accelerate this feedback loop by working with our customers and providing a network on which these functional ideas may be shared. Further, as new potential applications arise from this network, we exclusively will be able to supply the compatible modular components that will give the core device the metaphorical limbs it needs to perform said tasks. We believe this business model combined with our bottom-up design approach will cement the Eagle Eye Tracker as a unique and — most importantly — truly useful product.



## 8 Test Plan Appendix

### 8.1 Mechanical tests

<b>Component</b>	<b>Test</b>	<b>Desired Result</b>
Motors	Motion with different loads, velocities, and accelerations	No stalling or skipping steps
Motors	Apply large number of rotations at varying speed	Exact number of rotations is completed and encoders are accurate
Motors	Apply microstepping	Angle divisions are evenly spaced and motors exhibit minimal vibration
Motors	Run motor for long duration of time	Performance does not noticeably degrade
Stage	Apply large loads and large accelerations	Gears do not visibly wear or slip
Stage	Run both stages for long duration of time	Bearings are able to effectively reduce friction (and thus, wear) on contact points
Laser	Mount laser in various locations that would not interfere with the camera	Laser is able to accurately mark an object at a distance of 100 m

### 8.2 Hardware tests

<b>Component</b>	<b>Test</b>	<b>Desired Result</b>
Power Supply	Power on and run motors for long period of time	Voltage supplied remains stable



## EAGLE EYE SYSTEMS

Motor Driver	Motors are stepped at maximum frequency and speed for long period of time	Correct power is consistently outputted and motors do not stall
Motor Driver	Motors are turned on and loaded, and connections to microcontroller are monitored	Microcontroller does not experience voltage spikes due to motor
Laser	Microcontroller turns laser on/off and video is recorded of this	Laser turns on and off at precisely the specified time intervals
Motor Encoder	Motors are driven at various frequencies for large time periods	Output from motor encoder matches expected rotation count

### 8.3 Software tests

Component	Test	Desired Result
General	Unit tests	Code coverage of 80% and no less!
Connection	Ping test between PC and microcontroller	Average ping time <5 ms and peak ping time <10 ms
Motion control system	Arbitrary point far away from current orientation is given	Tracker changes orientation quickly towards target location
Motion control system	Given arbitrary, erratic paths	Motor output is stable and remains within tolerance of target path
Motion control system	Path through singular points (e.g. north pole)	Fast, accurate rotations about pole while remaining within tolerance of target path



## EAGLE EYE SYSTEMS

Motion control system	Path around boundaries (e.g. below horizon)	Motors do not attempt to go below angle
Spatial orientation	Move motor in arbitrary paths for long period of time	Remains calibrated and does not drift over long periods of time
Image recognition	Video of drone is fed into neural network	Drone is successfully recognized and its coordinates are correctly determined
Image recognition	Video of non-drone objects are fed into neural network	No drone is recognized

### 8.4 Integration tests

<b>Component</b>	<b>Test</b>	<b>Desired Result</b>
Camera	Object is placed within focal range	Object is sufficiently in-focus, has enough pixel resolution needed for object recognition, and video feed is delivered to microcontroller at 30 fps
Overall system	Object is moved across camera's field of view	Object is recognized and is kept within tolerance radius of center of frame
Overall system	System is run for long period of time (e.g. one day), then object is moved across camera's field of view	Object is recognized and is kept within tolerance radius of center of frame



## 8.5 User tests

<b>Component</b>	<b>Test</b>	<b>Desired Result</b>
GUI	Observe typical user setting up and using the Tracker	Setup time is within margin and user does not get stuck or frustrated at any stage
CLI	Observe advanced user setting up and using the Tracker	Setup time is within margin and user does not get stuck or frustrated at any stage

**ENSC 405W Grading Rubric for User Interface Design  
(5-10 Page Appendix in Design Specifications)**

Criteria	Details	Marks
<b>Introduction/Background</b>	Appendix introduces the purpose and scope of the User Interface Design.	<b>/05%</b>
<b>User Analysis</b>	Outlines the required user knowledge and restrictions with respect to the users' prior experience with similar systems or devices and with their physical abilities to use the proposed system or device.	<b>/10%</b>
<b>Technical Analysis</b>	Analysis in the appendix takes into account the "Seven Elements of UI Interaction" (discoverability, feedback, conceptual models, affordances, signifiers, mappings, constraints) outlined in the ENSC 405W lectures and Don Norman's text ( <i>The Design of Everyday Things</i> ). Analysis encompasses both hardware interfaces and software interfaces.	<b>/20%</b>
<b>Engineering Standards</b>	Appendix outlines specific engineering standards that apply to the proposed user interfaces for the device or system.	<b>/10%</b>
<b>Analytical Usability Testing</b>	Appendix details the analytical usability testing undertaken by the designers.	<b>/10%</b>
<b>Empirical Usability Testing</b>	Appendix details completed empirical usability testing with users and/or outlines the methods of testing required for future implementations. Addresses safe and reliable use of the device or system by eliminating or minimizing potential error (slips and mistakes) and enabling error recovery.	<b>/20%</b>
<b>Graphical Presentation</b>	Appendix illustrates concepts and proposed designs using graphics.	<b>/10%</b>
<b>Correctness/Style</b>	Correct spelling, grammar, and punctuation. Style is clear concise, and coherent. Uses passive voice judiciously.	<b>/05%</b>
<b>Conclusion/References</b>	Appendix conclusion succinctly summarizes the current state of the user interfaces and notes what work remains to be undertaken for the prototype. References are provided with respect to standards and other sources of information.	<b>/10%</b>
<b>CEAB Outcomes:</b> Below Standards, Marginal, Meets, Exceeds	1.3 Engineering Science Knowledge: 4.1 Requirement and Constraint Identification: 5.4 Documents and Graphic Generation: 8.2 Responsibilities of an Engineer:	





## 9 UI Appendix

### 9.1 Introduction

Given the Eagle Eye Tracker's wide array of applications, it is critical that the user interfaces are consistent, and do not become complex with specialized uses. A failure to develop this proper interface would result in a universal lack of motivation to purchase the product, let alone pursue creative and alternative applications to share with the community. Hence, Eagle Eye Systems will prioritize the development of a streamlined user interface. The ability to modify object detection parameters will be realized in a manner that is usable by an individual with the basic technical skills needed to operate a cell phone. Data presentation will also be streamlined to only produce outputs that are relevant to the user's application. This commitment will require significant development on the software end, but is absolutely necessary to realize the useful product we envision.

### 9.2 User Analysis

It is important that the Tracker is able to operate with minimal supervision. In the case of perimeter security, supervising officers may already have a plethora of security cameras broadcasting video to their surveillance hub. In addition to monitoring the cameras, officers likely have a multitude of other duties. Therefore, much consideration must be placed on the simplicity and autonomy of the Tracker's operation. An officer already well acquainted with security devices should have no difficulty comprehending the video feed from the Tracker(s). When a threat is detected by the Tracker, visual and auditory indicators will alert the appropriate personnel of a possible intrusion. This reduces the officer's overhead on constant monitoring of video feeds, allowing them to focus on other duties. The interface of the Tracker will be specifically designed for security scenarios to minimize supervision, and will focus on delivering critical data when required. [Figure 21](#) illustrates a preliminary visualization of this scenario.

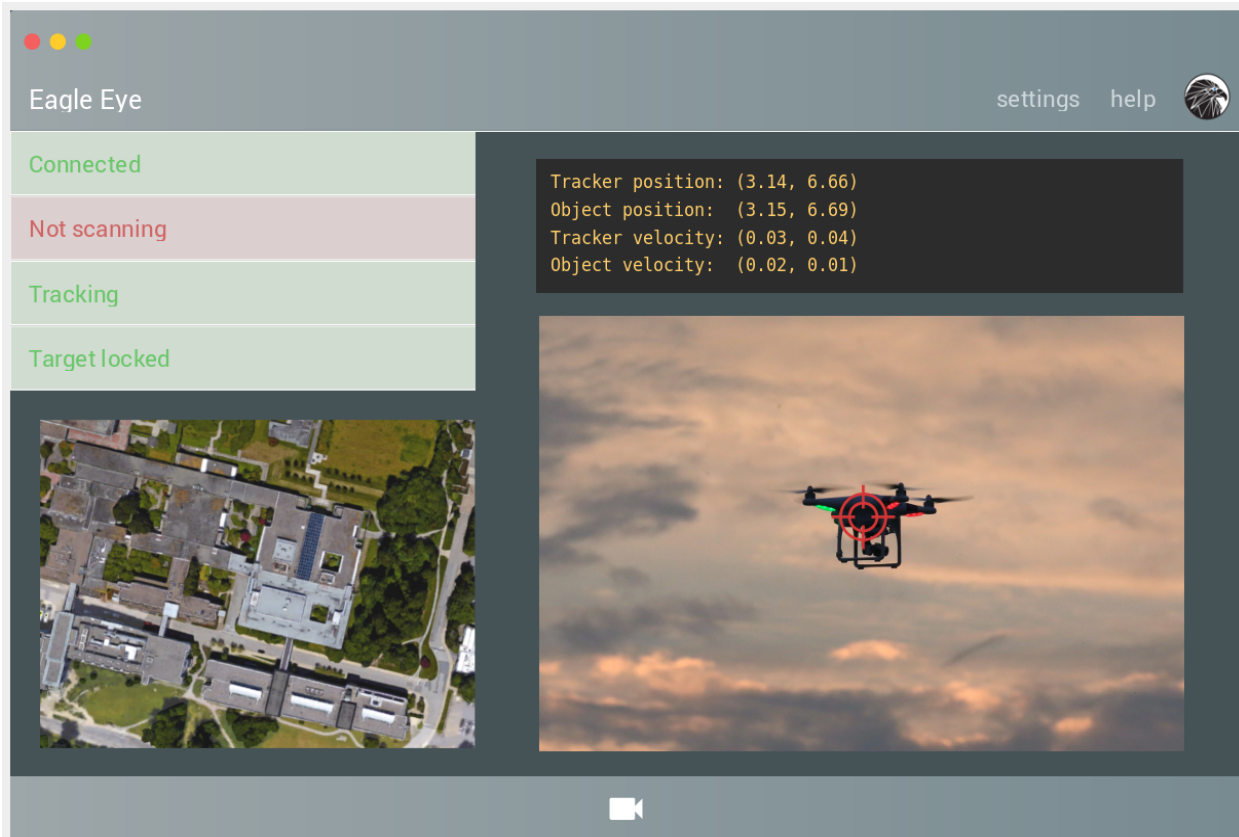
The integrity of the Tracker's internal security system is extremely important



## EAGLE EYE SYSTEMS

for any application involving surveillance or security. In the event of an attempted hack of a Tracker to circumvent security, a typical user likely does not have the expertise to regain control. Therefore, it is the responsibility of Eagle Eye Systems to implement sufficient security measures, or to provide the user with necessary guidance if a case arises where the default security measures prove insufficient.

Regarding installation of the Eagle Eye Tracker, the user must be familiar with basic knowledge of electronic devices such as supplying the Tracker with main power and connecting it to a computer. Experience with drills and screwdrivers are also necessary for fixing the tracker to a surface. Additionally, any supplemental software not included in the base model must be downloaded — a simple process for those who are technically adept enough to know that they require additional software.



**Figure 21:** GUI mockup.



## 9.3 Technical Analysis

### 9.3.1 Discoverability

The Eagle Eye Tracker shall be controlled from a PC via USB connection. It will have a CLI and a GUI. The CLI allows ease of deployment and is preferred by some users. The GUI provides an interface for a less technically inclined user, and is full-featured like the CLI.

The API will also provide documentation which will include examples. For example, an example of a code example:

```
from eagleeyetracker import Tracker

tracker = Tracker(port='usb')
tracker.attach_object_recognizer(user_specified_recognizer)
tracker.start()
```

### 9.3.2 Feedback

In the GUI ([Figure 21](#)), the control panel will display status indicators for connectivity, whether the tracker is enabled, an object is detected, current position, and other useful information. The CLI will be just as functional:

```
$ eagleeyetracker --status
Ping: 4ms

Connected: yes
Scanning: no
Tracking: yes

Tracker position: (3.14, 6.66)
Object position: (3.15, 6.69)
Tracker velocity: (0.03, 0.04)
Object velocity: (0.02, 0.01)
```



### 9.3.3 Conceptual models

The CLI and GUI will follow standard design guidelines (e.g. -help and a clutter-free GUI main layout). Any advanced GUI functionality can be hidden to reduce information overload, but with sufficient hints to allow the user to discover.

### 9.3.4 Affordances

The GUI will present a live video stream and status indicators to demonstrate that the tracker is running and functional. A test demo can also be run to ensure that the product is setup correctly. This is important for applications such as drone tracking since the user needs to confirm that the product will work properly when the need arises.

### 9.3.5 Signifiers/Mappings

Colors and GUI indicators will be used to indicate status. A live video stream will be provided.

### 9.3.6 Constraints

These example CLI commands exhibit some constraints in what can be done with the Tracker:

```
eagleeyetracker --help
eagleeyetracker --version
eagleeyetracker --port=usb --status
eagleeyetracker --port=usb --application=drone --start
eagleeyetracker --port=usb --application=bird --start
eagleeyetracker --port=usb --application=custom --start
eagleeyetracker --port=usb --get-video-stream
eagleeyetracker --log=log.log --verbose
```



## 9.4 Engineering Standards

The following engineering standards are relevant in the design and implementation of all user interfaces included in the Eagle Eye Tracker.

- |                      |   |
|----------------------|---|
| <b>IEC60065:2014</b> | Audio, video and similar electronic apparatus — safety requirements [13]                      |
| <b>ISO13854:1996</b> | Minimum gaps to avoid crushing of parts of the human body [14]                                |
| <b>ISO12100:2010</b> | Safety of machinery — general principles for design — risk assessment and risk reduction [15] |

## 9.5 Analytical Usability Testing

We will be using a heuristic-based approach for analytical testing of our system. The evaluation will be done by our design team, following a subset of cognitive engineering principles compiled by Gerhardt-Powals [5]; we will be judging our product depending on its compliance with each principle. This process will be done in parallel to implementation tasks, and therefore can give us a good sense of the product's usability starting from the early stages of the prototype. Doing so will allow for adequate time to address any changes that will have to be made to the design. The list of principles and how we will attempt to address them is listed below:

1. **Automate unwanted workload:** As a solution that is envisioned to be fully automated, offloading work from the user to the system is an inherent requirement of the Eagle Eye Tracker. All complex algorithms and calculations involved in the operation of the tracker will be abstracted behind the interfaces to minimize the cognitive load experienced by users. We will be evaluating how successful our product is in accomplishing this by testing these individual interfaces and the functionality that they expose.



## EAGLE EYE SYSTEMS

2. **Reduce uncertainty:** As a detection and tracking system, accuracy is a crucial aspect of the Eagle Eye Tracker and this information must be conveyed clearly to the user. The laser mounted on the system will help reflect how accurately the product is tracking the object. As well, useful information such as whether the system is idle or tracking will be displayed to reflect the current state of the tracker.
3. **Fuse data:** Lower level data will be accumulated and presented in a way that is easy to understand for any user so that they will not need an advanced technical background in order to operate our product. Status indicators will be displayed in a GUI, alongside a live view of the camera feed.
4. **Present new information with meaningful aids to interpretation:** Status updates will be displayed with color coded labels so that users are aware of any changes. Furthermore, bounding boxes will be used on the detected object to indicate that it is being tracked.
5. **Use names that are conceptually related to function:** Information in the interfaces will be labelled clearly. As well, we will refrain from using terms that are too technical so that the interface can be understood by the average user.
6. **Group data in consistently meaningful ways:** Information in the interfaces will be logically laid out. For example, information related to numerical data like position and velocity should be grouped together. Status on the tracker itself should be in a separate group.
7. **Include in the displays only that information needed by the user at a given time:** Ensure that the information presented in the interfaces is actually valuable for the user or for diagnostic purposes. Do not clutter the view with unnecessary data such as stock quotes.
8. **Provide multiple coding of data when appropriate:** Users will be given an option of using a CLI or GUI depending on their preferences and technical background. An API will also be available for those who want to get into development.



## 9.6 Empirical Usability Testing

As the end user will ultimately determine the success of the Eagle Eye Tracker, empirical testing is crucial in identifying problems and biases in the design of our product. We will have to gather sufficient feedback from these sessions in order to judge its usability, including the specific areas that are satisfactory or needs improvement. By incorporating real users that reflect our target market into the testing process, we will also be able to gain further insight into their needs. Furthermore, having people beyond our team test the product will increase the likelihood of discovering bugs and other issues that need to be fixed before release of the prototype.

Currently, we have not conducted any testing of our product with real users but are planning to do so once the proof-of-concept is completed. Hence, the first session will be planned for the presentation on April 9th. The evaluation will be informal, consisting of a simple demo and questionnaire aimed to collect feedback on the overall design, as well as to gauge the level of interest in the Eagle Eye Tracker. We will start with questions for the users, such as initial impressions on the design, but will also be encouraging them to raise any questions and critique about the product. Since our proof-of-concept currently only supports tracking of an arbitrary object that enters the field of view, users will also be encouraged to try out the tracking system for themselves. We will be actively keeping written records of our observations during this procedure; for example, we will take note of the tracking accuracy, what situations the tracking is unsuccessful, and how well the system recovers from errors. Lastly, we will ask for direct feedback from the users, including what improvements they would like to see and what they liked or didn't like about the product. Since our user interface components have yet to be implemented, opinions on the UI design mock-ups that we have created will also be collected.

The second session of empirical testing will be done in the latter portion of the prototype stage, ideally after many of the core features have been implemented. This will involve a more structured approach in comparison to the first. We will gather users to conduct a series of tasks that will involve utilizing the product and its interfaces. Because of our product's vast applicability to many industries, our criteria for choosing the participants will be mostly open but will be restricted



## EAGLE EYE SYSTEMS

to a certain age range and basic technical background. As the operation of the Eagle Eye Tracker is automated for the most part, our tests will be focused on user-facing features such as the graphical user interface (GUI), command line interface (CLI), and usage of the application programming interface (API). The tasks will range from simple, such as navigating the interfaces and retrieving certain information, to slightly more complicated, such as setting a custom object to track. Direct guidance will be minimized so that users will be encouraged to use the documentation provided. During this process, we will be maintaining close observation on the users' experience and the steps that they take to complete the specified task. Specifically, details such as ease of use, errors, and overall user satisfaction will be taken into account.

A crucial factor that affects the usability of any product is its vulnerability and response to user errors. Therefore, the team will be paying particularly close attention to any occurrences of slips and mistakes during the testing procedure. A mistake is defined as an error in choosing the method to carry out a specific task, while slips are errors during the execution of an intended method[6]. To minimize both variations of these errors, documentation will be provided that details the exact features that are supported in the interfaces, as well as examples to guide the user; this will come in the form of manual pages for the GUI and CLI, and a reference for the API. Recovery from error is also an essential part of the process, as slips and mistakes are inevitably bound to happen at some point. The user interface of the Eagle Eye Tracker will provide instantaneous feedback if such errors occur through the use of descriptive error messages. Furthermore, logs will be continuously recorded and stored to aid users and our support team in debugging issues. Such logs will contain helpful information that describes the state of the system throughout its operation including API calls made, parameter values, and diagnostic messages.

Lastly, an anonymous survey will be sent out to the participants following the evaluation session to garner final opinions on the product. Through extensive review of our users' experience with the product during the testing sessions, the design team will meet to identify notable issues and define a plan for improving affected areas. These changes will then be implemented and incorporated into the final prototype.





## 9.7 Conclusion

User interfaces can undoubtedly make or break a product. This is certainly the case for the Eagle Eye Tracker, especially due to the wide range of markets we aim to insert the Tracker into. This variety gives rise to an inherent variation in the needs of the users, and the interfaces we develop must adapt to and reflect this reality. As such, in this document we have outlined the analytical and empirical steps we have taken and intend to take to ensure the viability of our product's usability. The analytical principles we have outlined are certainly required to fulfill our commitment to our diverse user group; however, empirical testing as outlined herein is the true measure of how our users respond to the design of our product and its interfaces. To date, we have laid out conceptual frameworks for how our various interfaces will be interlinked, and have begun the development of the main GUI.

# ENSC 405W Grading Rubric for ENSC 440 Planning Appendix

## (5-10 Page Appendix in Design Specifications)

Criteria	Details	Marks
<b>Introduction/Background</b>	Introduces basic purpose of the project. Includes clear project background.	<b>/05%</b>
<b>Scope/Risks/Benefits</b>	Clearly outlines 440 project scope. Details both potential risks involved in project and potential benefits flowing from it.	<b>/10%</b>
<b>Market/Competition/Research Rationale</b>	Describes the market for the proposed commercial project and details the current competition. For a research project, the need for the proposed system or device is outlined and current solutions are detailed.	<b>/10%</b>
<b>Personnel Management</b>	Details which team members will be assigned to the various tasks in ENSC 440. Also specifically details external resources who will be consulted.	<b>/15%</b>
<b>Time Management</b>	Details major processes and milestones of the project. Includes both Gantt and Milestone charts and/or PERT charts as necessary for ENSC 440 (MS Project). Includes contingency planning.	<b>/15%</b>
<b>Budgetary Management</b>	Includes a realistic estimate of project costs for ENSC 440. Includes potential funding sources. Allows for contingencies.	<b>/15%</b>
<b>Conclusion/References</b>	Summarizes project and motivates readers. Includes references for information from other sources.	<b>/10%</b>
<b>Rhetorical Issues</b>	Document is persuasive and demonstrates that the project will be on time and within budget. Clearly considers audience expertise and interests.	<b>/10%</b>
<b>Format/Correctness/Style</b>	Pages are numbered, figures and tables are introduced, headings are numbered, etc. References and citations are properly formatted. Correct spelling, grammar, and punctuation. Style is clear, concise, and coherent.	<b>/10%</b>
<b>Comments:</b>		



## **10 440 Planning Appendix**

### **10.1 Introduction**

Development of the Eagle Eye Tracker has thus far been parallelized. The software team has been using the proof of concept model as a platform on which to develop and test the object tracking algorithms. Meanwhile, the hardware team has been focusing on setting the framework for the prototype, thus reducing transition time into ENSC 440. Despite the Eagle Eye Tracker being a very integrated and unified device in terms of interrelated hardware and software components, this strategy has proven successful, as we have thus far met the deadlines set out at in the preceding months. In this appendix, we describe specific details concerning the evolution of the project to where it is now, and set a course for continued progress into the future. This includes scope changes, risks and benefits, market analysis, specific application analysis, as well as administrative content including personnel management and budget/funding plans.

### **10.2 Scope/Risks/Benefits**

#### **10.2.1 Scope**

Since the inception of the idea for the Eagle Eye Tracker, the team has decided to expand the scope of the project to encompass a wider variety of applications. Our vision is to create a universal solution to the problem of object tracking. By generalizing the recognition algorithm and designing for modularity, the product will be highly customizable and able to track different objects in real-time. Beyond drone tracking, other examples of applications that the Eagle Eye Tracker could be used for include pest deterrent for farms, ball tracking in sports, and photography for hobbyists. To support this, we will also be writing an API that will allow users to specify whether to use built-in models for the deep learning algorithm or to incorporate their own pre-trained model for custom objects. In the future, we hope to establish an online community to allow users to share their creative ideas and experiences with our product.



## EAGLE EYE SYSTEMS

For the next phase of the project, our primary goal will be the completion of our prototype. In order to accomplish this, we will be focusing heavily on the features that have yet to be implemented in the proof of concept, such as the base platform and object recognition algorithm. On the hardware side, we will be starting construction of the rotating stages that will support and drive the camera module. This will include a sophisticated motor control system that will allow the platform to move precisely and smoothly with two degrees of freedom. In terms of software components, we will begin to integrate our chosen deep learning algorithm into the product. This will involve various subtasks such as gathering a sufficient dataset and training the model to recognize a chosen object with high accuracy. In addition, we will continue to refine the tracking algorithm that we had already implemented in our proof of concept. By the end of the semester, we hope to have a functional fully prototype.

### 10.2.2 Benefits

With the scope for the Eagle Eye Tracker broadened, our confidence in its potential benefits have also expanded. Our team believes that a modular, customizable design will enable the product to be incorporated into various industries with little effort. As a detection system for drone defense, the Eagle Eye Tracker will help maintain public safety by countering misuse of UAV technology. This will have vast applications from preventing contraband smuggling to protecting the airspace from rogue drones.

With a proper laser mounted, the Eagle Eye Tracker can also act as a pest deterrent for farms. Wildlife such as birds are an ongoing problem for farms, causing substantial damage to crops every year [7]. The tracker will be a much less obtrusive solution in comparison to other methods such as the usage of chemicals or loud noises.

Additionally, hobbyists using the Eagle Eye Tracker for photography or videography will enjoy its lightweight and modular design. This will allow the product to be highly portable to a variety of environments. Taking it a step further, the team will be looking to build an ecosystem around the Eagle Eye Tracker in the future, including a suite of development tools, services, and an online community to help



nurture creativity and expand the usages of the product.

### 10.2.3 Risks

Misidentification is a significant risk of any recognition system; even with advanced algorithms such as deep learning, unexpected errors will still occur and may lead to unfavorable outcomes. Eagle Eye Systems is very aware of this risk and will work to create the most robust system possible. Mitigation and recovery from these errors will be an essential component of our product's design from an early stage. We will be training our models with a sufficient and high-quality dataset in order to ensure excellent accuracy.

Latency is also another notable risk that affects any video processing system, specifically one that works with live streaming data like the Eagle Eye Tracker. This was an apparent issue in our proof of concept when we were working on the tracking algorithm, which led to overshooting of the motors and inaccurate results. However, by discovering this issue early into the implementation phase, we were given sufficient time to address the problem. In order to reduce latency, communication between the modules in the POC were switched from using Bluetooth to USB. USB has inherently faster data transfer speeds than Bluetooth and our changes led to a 25x increase in this area.

Lastly, laws and regulations regarding drone defense and laser safety may limit the marketability of the Eagle Eye Tracker. Currently, destruction of drones, which fall under the category of an aircraft, is illegal and may only be limited to certain government bodies. As a pest deterrent, safety concerns might also be raised if a user decides to mount an inappropriately high-powered laser for the job, which will lead to more harm than good. In fact, as a modular object tracking system that encourages customization, its most significant benefit may also be its biggest risk and it is undeniable that there will be a potential for misuse of this product. Therefore, it will be important for us to understand and consult all our customers with respect to their intended usages of the Eagle Eye Tracker.



## **10.3 Market Competition/Research Rationale**

The largest selling factor of our product is its wide applicability. The Eagle Eye Tracker uses deep learning for recognition of a variety of objects and animals. As a result, our market includes governments, farms, state-owned enterprises, private businesses, and individual citizens. Although competition exists for each of the applications described below, such a versatile system does not. For ENSC 440, our plan is to narrow our application to drone detection and pest deterrent. The markets for various applications are described below.

### **10.3.1 Drone Detection**

The usage of drones for illegal purposes is increasing. As a result, governments feel increasing pressure to find solutions for law enforcement. Recently, a news article published by CTV News Vancouver reported that a drone had dropped a package containing \$26,500 worth of prohibited goods into a medium-security prison [9]. Smuggling of narcotics and weapons across borders and into prisons is an act to be taken very seriously and for this reason, we believe that governments will take a serious interest in our product. In particular, a country's border or prison walls could benefit from anti-drone technology. Though the Canadian government is a big market, multiple times more revenue would be generated by marketing the Eagle Eye Tracker to the United States government (federal and state level). With 3000 prisons in the United States, sending 10 trackers to each prison would provide a revenue on the order of magnitude of 10 million dollars. Furthermore, our product may be marketed to the Trump administration to be used for border security. The 3100km US-Mexico border can have a tracker set up every 100m for 24 hour surveillance. Though this would also cost them many millions of dollars, it is a small dent in comparison to the wall that the administration plans to build.

Another market for this application is for state-owned enterprises, where provincial or state governments have significant ownership over the enterprise but do not control its day-to-day decisions. These are the kinds of companies that own and operate public venues such as BC Place Stadium. Such venues are often targets of terrorist attacks because of the overwhelmingly large quantity of people attending



## EAGLE EYE SYSTEMS

the events. The Eagle Eye Tracker seeks to improve the safety of the public by installing our product at these venues

The drone defense system market was worth around 801.8 million USD in 2016 and is estimated to grow to 15 968.2 million USD by 2022 [8]. Typical drone tracking and detection systems do not include the full functionality of our system. For instance, current solutions do not use any form of automated countermeasures. A company named DeTect has developed a technology called DroneWatcherRF [10] which detects only commercially available drones in a 1-2 mile radius. Our product, although it cannot track drones 2 miles away, will be able to detect all types of drones, not only commercially available ones. DroneWatcherRF is an electric box that is installed around the perimeter of a facility and requires the user to take action once an intrusive drone is identified. The Eagle Eye Tracker does not take up nearly as much space and can include a high powered laser to decapitate the drone. As outlined in the mechanical specifications, our product only takes up 37cm by 37cm of ground space, meaning installation is much simpler and the cost is lower.

### 10.3.2 Bird/Animal Deterrent

For generations, farmers have been using scarecrows to deter harmful birds from their farms. With modern technology, more effective solutions exist. In addition to drone detection, our solution is flexible enough to have such agricultural applications. Our product will aim a low power laser towards the birds' eyes to scare them away. The typical laser class used for this purpose is the class III B laser, which have a power rating between 5mW and 500mW.

This application can also be implemented at outdoor venues. Unwanted bird excretion can be very problematic at outdoor events. For this reason, we are including event organizers in our market. Other industries that may be interested in bird deterrence include aviation, industrial sites, and oil and gas fields.

The company Bird Control Group has developed multiple solutions for this very niche market with their most similar product to ours being the Autonomic [11]. The Autonomic is not as compact as our product taking up about four times as



## EAGLE EYE SYSTEMS

much area. Unfortunately, since this company is a multinational organization, the Autonomic costs over \$10 thousand. It also does not have such a versatile system. A farmer may not have the funds to purchase such technology.

In addition to this, low-tech solutions exist such as scarecrows, or kites resembling birds of prey. Unfortunately, these methods underestimate the intelligence of many birds. Birds quickly realize that these scarecrows and kites do not pose any threats. Our fully automated system will not encounter such a problem because although birds may realize the laser does not pose direct harm, the lasers will act as a temporary vision impairing deterrent.

The Eagle Eye Tracker can be further extended as a general animal repellent. For example, look no further than the SFU Burnaby campus! Raccoons are frequently entering SFU buildings and wreaking havoc by spreading germs and knocking over garbage cans in search of food. A company named HavaHart has developed a series of electronic animal repellents. However, in contrast with our fully automated recognition and aiming, these devices need to be manually aimed [12].

## 10.4 Personnel Management

### 10.4.1 Technical/Design Roles

#### Hardware Team



**Bud Yarrow** — Chief Executive Officer; Hardware Team Lead. In charge of overseeing the mechanical and electrical Design and developments. Assisting mechanical and electrical system leads to ensure successful completion of the hardware tasks meeting necessary system requirements. Providing the resources necessary for the hardware team to successfully complete their tasks.

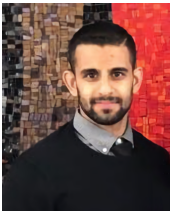




## EAGLE EYE SYSTEMS



**Martin Leung** — Chief Operating Officer; Electrical Systems Lead. In charge of the design and development of all electronics systems including the motor control system, laser control system, electronics power supply system. Also will be responsible for design and assembly of the PCBs in the final system.



**Naim Tejani** — Chief Financial Officer; Mechanical Systems Lead. In charge of the design and Construction of the Mechanical system. This will include the stage with 2 degrees of angular freedom. Will need to ensure smooth working of the rotational components under load. Sufficient space and strength to mount the and support the camera load and provide enclosure space for system electronics.

### Software Team



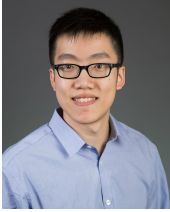
**Mateen Ulhaq** — Chief Technical Officer; Software Team Lead. Responsible for the object identification software using deep learning neural networks as required. In charge of overseeing the completion of other software components and making sure tracking and system integration leads have sufficient resources to complete their tasks.



**Arman Athwal** — Chief Communications Officer; Tracking Systems Lead. Responsible for developing the algorithms for tracking the identified objects. Ensure smooth and precise tracking of objects using predictive algorithms and feedback control system techniques as necessary to minimize latency and ensure stability. May be involved with training the neural net and acquiring training datasets.



## EAGLE EYE SYSTEMS



**Victor Yun** — Chief Information Officer; Systems Integration Lead. Responsible for integrating different software systems together with the hardware to create the final product. This will include the handshaking required to transfer data between different hardware including the camera, hardware accelerated neural network, CPU for object tracking, and the MCU for motor control. May also be involved with the hardware team to do the low level hardware programming for the motor control system.

### 10.4.2 External Contacts

#### **Jie Liang**

Associate Director, School of Engineering Science

Deep-learning expert

Co-founder and the President of AltumView Systems Inc.

Email | [jiel@sfu.ca](mailto:jiel@sfu.ca)

Phone | 1-778-782-5484

Website | <http://www.sfu.ca/~jiel/>

#### **Marinko V. Sarunic**

Professor, School of Engineering Science

Photonics expert

Email | [msarunic@sfu.ca](mailto:msarunic@sfu.ca)

Phone | 1-778-782-7654

Website | <http://borg.ensc.sfu.ca/>

#### **Terrance**

Information Manager, EIM PCB

PCB design and printing company

Email | [info@eimtechnology.com](mailto:info@eimtechnology.com)

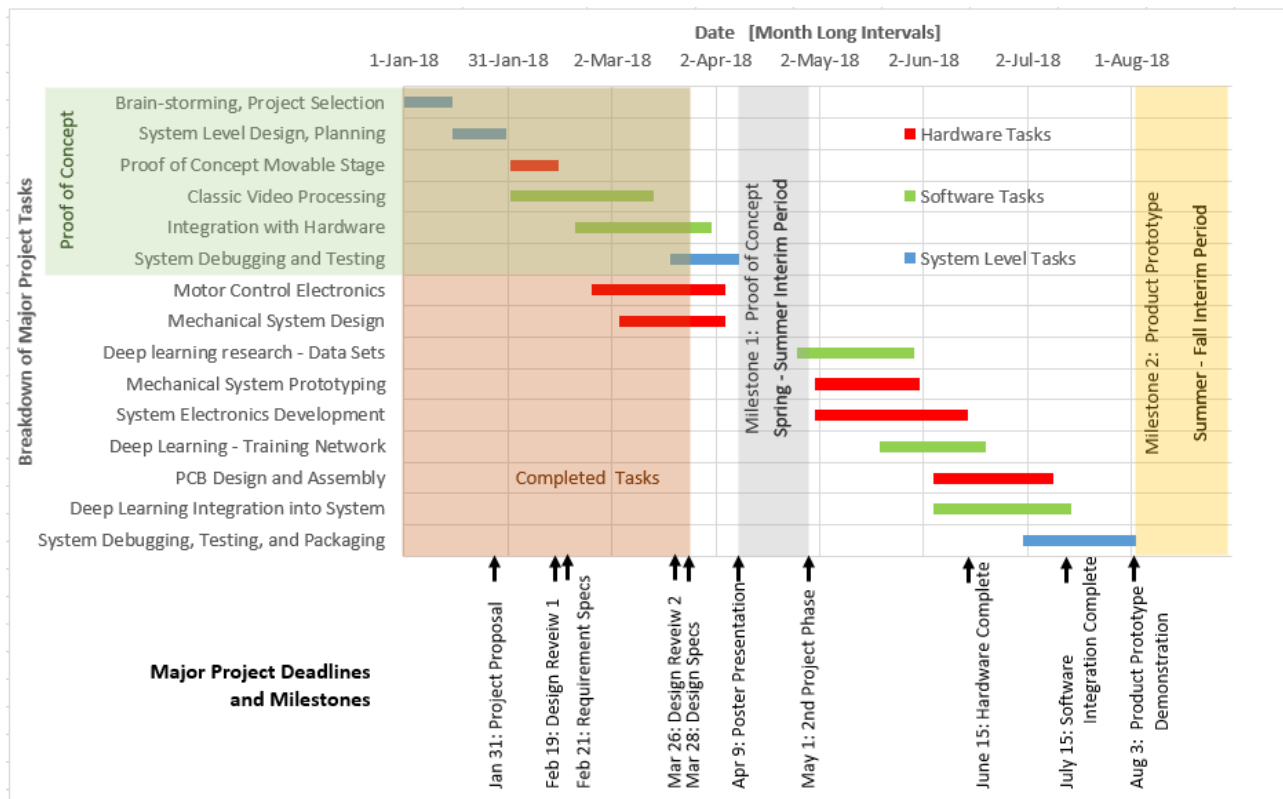
Phone | 1-778-321-1369

Website | <http://pcb.eimtechnology.com>



## 10.5 Time Management

In order to reduce development time, we have divided ourselves into teams to work on separate subsystems in parallel. The main categories include hardware and software, which will be done in parallel by software and hardware teams. Furthermore, the Hardware team is broken into Mechanical and Electronics components which will also follow parallel development stages. System integration will follow, leaving adequate time for testing and debugging. The detailed proposed time-line can be seen in the Gantt chart in [Figure 22](#).



**Figure 22:** Gantt chart outlining the development stages of the Eagle Eye Tracker, including important deadlines and milestones.



### **10.5.1 High-Level System Design**

In this phase, we plan the technologies we will use for the different subsystems. We also determine how various subsystems will cohesively fit together. This phase will already be completed as part of the design stages and brain storming done in the proof of concept development.

### **10.5.2 Hardware Design**

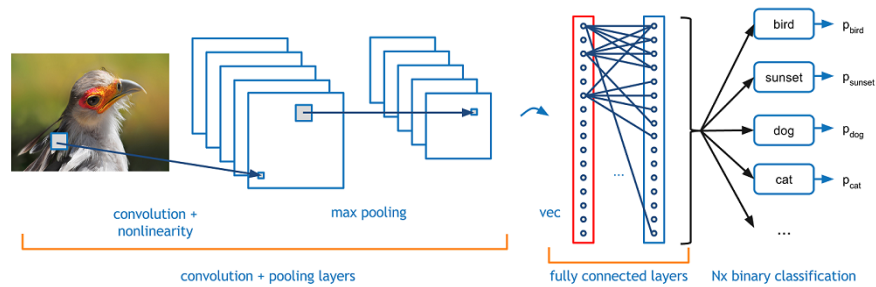
The prototype hardware design was started earlier than suggested in order to have as much time as possible for integration of the software into the system. This is due to the fact that integration of our several sub-systems is a crucial factor in the success of our project. Regarding the electronics aspect, the motor control electronics will be designed and tested before the end of the completion of ENSC405. The Mechanical System will be designed in parallel during this time. Starting in May the Mechanical System Development and fabrication will be initiated, with a completion date at the end of May, allowing for sufficient time contingencies. The system electronics development will also be done during the mechanical development, and will be integrated into the mechanical system for testing. Once the electronics system is working as required with the mechanics, the PCB design will be initiated and upon completion integrated into the mechanical system in place of the prototyped electronics.

### **10.5.3 Software Development**

The software development will occur in parallel to the hardware development wherever possible. Following the completion of the proof of concept, the upgrade to a prototype will be set in motion starting in the summer semester. However, many of the algorithms developed on the proof of concept for tracking can be reused in the final prototype, with some added upgrades to enhance the tracking precision, latency, and stability. One component of this will focus on the application of deep learning in this project. This will focus mainly on research and looking at data sets for training and suitable models for this project, as depicted in [Figure 23](#).



## EAGLE EYE SYSTEMS



**Figure 23:** Convolutional neural network. Takes an input image and classifies it. The input image in this case appears to be a bird of some kind.

The main software distinction between the proof of concept and the prototype is the inclusion of the deep learning algorithm, as the latter uses optical flow as the basis for identification and tracking. Once this software upgrade is complete, the more robust algorithms will be ported over to the upgraded hardware, which will require time for integration and testing. Upon completion the Eagle Eye Tracker, as visualized in [Figure 1](#), will be presented in a functional demonstration in milestone 2.

### 10.5.4 Contingency Plan

Most development sections are given some buffer time (i.e. more time than is expected to complete the tasks). There is still a chance that there could be delays in certain tasks.

In terms of the motor control electronics, if there is no time to design a PCB, prototyping boards may be used instead. Also, if the desired motor control system with high accuracy is too complicated to use and causes extensive delays, a simpler control system exists which we can fall back to.

In catastrophic failure regarding the mechanical system, we could fall back onto the hardware used in the proof of concept. Likewise for the software, should the deep-learning prove to be too ambitious for the time frame, we can fall back to classical techniques, similar to the ones used in the proof of concept. There would still be several optimizations that could be used to tailor the classical methods to work reasonably well for our applications.



## 10.6 Budget Management

We have spent about \$285 to date on components for our product. We plan to spend another estimated \$750 but are allowing for a 20% contingency fund allotting us \$900 to spend during the course of ENSC 440.

<b>Item</b>	<b>Cost</b>	<b>Purchased</b>	<b>Unpurchased</b>
<b>Arduino</b>	<b>\$20</b>	✓	
<b>Arduino (replacement)</b>	<b>\$20</b>		✓
<b>Bluetooth module</b>	<b>\$30</b>		✓
<b>Camera</b>	<b>\$400</b>		✓
<b>Electronics test accessories</b>	<b>\$30</b>	✓	
<b>Laser pointer</b>	<b>\$5</b>	✓	
<b>Motor control electronics</b>	<b>\$80</b>	✓	
<b>Motor 1</b>	<b>\$40</b>	✓	
<b>Motor 2</b>	<b>\$40</b>		✓
<b>Raspberry Pi kit</b>	<b>\$80</b>	✓	
<b>Stage materials</b>	<b>\$100</b>		✓
<b>Intel Movidius DNN USB</b>	<b>\$80</b>		✓
<b>Subtotal:</b>		<b>\$255</b>	<b>\$670</b>
<b>Tax (12%):</b>		<b>\$30</b>	<b>\$80</b>
<b>Contingency (20%):</b>		<b>N/A</b>	<b>\$150</b>
<b>Total:</b>		<b>\$285</b>	<b>\$900</b>

Table 1: Budget for ENSC 440

## 10.7 Funding

Our main funding options are the Engineering Science Student Endowment Fund and IEEE Student Project Funds; the parts library from the ESSS may also have useful materials that we may choose to borrow. Alternatives such as the Wighton Engineering Development Fund are also being considered. Our CFO has been in touch with the IEEE chair and is currently waiting for a response. Inquiries and



## EAGLE EYE SYSTEMS

applications for the ESSS and Wighton funds will be completed promptly in order to ensure that we have a solid understanding of our financial plans going forward. Any outstanding expenses will be distributed evenly amongst the team members; as many of the parts are perfectly operational independently, they can be seen as an investment and used in future projects as well.

### **10.8 Conclusion**

Progress thus far on the Eagle Eye Tracker is promising. The change of scope has complemented the design process of the product; subsequent market analysis has shown vast opportunity for the product to flourish; expenditures have hitherto fallen within an expected range and covered by contingency planning; but most of all, the functionality of the proof of concept is performing as expected, with transition into the prototype stage already under way. This appendix has served to describe progress made as well as anticipated future work. It is worth noting that the approach our group has taken - parallelizing proof of concept and some prototype work - has some drawbacks that we are aware of but are working to mitigate. Among these include the relative independence of the aforementioned project stages, which could result in difficulty in translating from one to the other. However, with the results achieved so far, we are confident that we will be able to carry our success forward into the next stage of the project, and deliver a prototype worthy of our vision.



## 11 Glossary

<b>Application Programming Interface (API)</b>	A collection of subroutines used to interface with external software components.
<b>Artificial neural network</b>	A computing model made up of interconnected nodes (neurons) which produces output in response to external inputs. It is inspired by the structure of the brain.
<b>Computer vision</b>	A field that explores algorithms for machines to gain visual understanding of images and video.
<b>Deep learning</b>	A machine learning technique based on learning data representations using neural networks, typically making use of multiple layers of neurons.
<b>Drone</b>	An unmanned aircraft controlled remotely or autonomously.
<b>Graphics Processing Unit (GPU)</b>	A processor capable of executing many instructions in parallel; particularly useful for image processing and neural networks.
<b>LIDAR</b>	A sensing method that uses light to measure distances [1]
<b>Machine learning</b>	A field of computer science and statistics that studies algorithms for learning and improving autonomously from data.





## References

- [1] National Oceanic and Atmospheric Administration; US Department of Commerce. [online] Available at: <https://oceanservice.noaa.gov/facts/lidar.html>. [Accessed 21 February 2018].
- [2] Shi-Tomasi Corner Detector and Good Features to Track [Online] Available at: [https://docs.opencv.org/3.0-beta/doc/py\\_tutorials/py\\_feature2d/py\\_shi\\_tomasi/py\\_shi\\_tomasi.html](https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_feature2d/py_shi_tomasi/py_shi_tomasi.html) [Accessed 31 March 2018]
- [3] YOLO: Real-Time Object Detection [Online] Available at: <https://pjreddie.com/darknet/yolo/> [Accessed 31 March 2018]
- [4] Yolo-v3 and Yolo-v2 for Windows and Linux [Online] Available at: <https://github.com/AlexeyAB/darknet> [Accessed 31 March 2018]
- [5] Heuristic evaluation [Online] Available at: [https://en.wikipedia.org/wiki/Heuristic\\_evaluation](https://en.wikipedia.org/wiki/Heuristic_evaluation) [Accessed 29 March 2018]
- [6] The Glossary of Human Computer Interaction: Human error (slips and mistakes) [Online] Available at: <https://www.interaction-design.org/literature/book/the-glossary-of-human-computer-interaction/human-error-slips-and-mistakes> [Accessed 29 March 2018]
- [7] Anderson, A., et al. (2013). Bird damage to select fruit crops: The cost of damage and the benefits of control in five states. Crop Protection, vol. 52, p. 103-109. [online] Available at: [https://www.aphis.usda.gov/wildlife\\_damage/nwrc/publications/13pubs/anderson132.pdf](https://www.aphis.usda.gov/wildlife_damage/nwrc/publications/13pubs/anderson132.pdf) [Accessed 30 Jan. 2018]
- [8] Marketsandmarkets Research Private Limited. [online] Available at: <https://www.marketsandmarkets.com/Market-Reports/drone-identification-market-85835738.html>. [Accessed 29 March 2018]
- [9] Mangione, K. (2018). Drone used to bring drugs, tobacco into B.C. prison.



## EAGLE EYE SYSTEMS

- [online] Available at: <https://bc.ctvnews.ca/drone-used-to-bring-drugs-tobacco-into-b-c-prison-1.3757360> [Accessed 12 Jan. 2018]
- [10] DeTect Inc. Drone Detection & Defense Systems. [online] Available at: <http://detect-inc.com/drone-detection-defense-systems/> [Accessed 26 Jan 2018].
- [11] Bird Control Group B.V. [online] Available at: <https://birdcontrolgroup.com/agrilaser-autonomic/> [Accessed 19 February 2018]
- [12] HavaHart. [online] Available at: <http://www.havahart.com/spray-away-hydro-remote-animal-repellent-sprinkler-5267> [Accessed 20 February 2018]
- [13] IEC 60065:2014, International Electrotechnical Commission, 2018. [online]. Available at: <https://webstore.iec.ch/publication/494>. [Accessed 19 February 2018].
- [14] ISO 13854:1996, International Organization for Standardization, [online]. Available at: [http://www.iso.org/iso/home/store/catalogue\\_tc/catalogue\\_detail.htm?csnumber=22598](http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=22598). [Accessed 20 February 2018].
- [15] ISO 12100:2010, International Organization for Standardization, [online]. Available at: [http://www.iso.org/iso/home/store/catalogue\\_tc/catalogue\\_detail.htm?csnumber=51528](http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=51528). [Accessed 20 February 2018].



## 12 Appendix: Requirement Specifications

- A: Requirement for proof-of-concept
- B: Requirement for prototype
- C: Requirement for production version

### 12.0.1 Mechanical Requirements

- REQ 1.1.1-ABC** Tracker shall have precisely two motors, each corresponding to an angular degree of freedom
- REQ 1.1.2-BC** Each angular degree of freedom will have a resolution of at most  $0.1^\circ$
- REQ 1.1.3-BC** Each motor will be fast enough to move each degree of freedom at least  $45^\circ/s$
- REQ 1.1.4-BC** Each motor will move smoothly and minimize vibrations induced to system and camera in particular
- REQ 1.1.5-BC** Modular product design that allows for easy upgrading

### 12.0.2 Electronics Requirements

- REQ 1.2.1-BC** Control system for motors will drive motors with smoothness, accuracy and speed required by the mechanical system
- REQ 1.2.2-BC** Control system for laser will drive laser as required by software
- REQ 1.2.3-C** System will have uninterruptible power supply, regulated to eliminate interruptions
- REQ 1.2.4-C** System will have options for solar power or directly from grid, both with battery backup



## EAGLE EYE SYSTEMS

### 12.0.3 Optical Requirements

- REQ 1.3.1-BC** Product will be capable of imaging objects 5 to 100 meters away with adequate resolution
- REQ 1.3.2-ABC** Objects will be resolved by at least 100 by 100 pixels to make object identification accurate
- REQ 1.3.3-C** Option for infrared camera for nighttime detection
- REQ 1.3.4-BC** Option for laser pointer to mark target
- REQ 1.3.5-C** Option for high power laser to destroy target

### 12.0.4 Software General Requirements

- REQ 2.1.1-ABC** Real-time video stream must be read in with low latency
- REQ 2.1.2-ABC** Integrated motion controller will ensure accurate mapping between detected object position and desired motor angles
- REQ 2.1.3-BC** Datasets of target objects used to train neural network will be sufficiently large to form a good internal representation
- REQ 2.1.4-C** For portability, the neural network may run on standalone system

### 12.0.5 Identification Requirements

- REQ 2.2.1-BC** Targets will be identified with minimal false positives
- REQ 2.2.2-C** Objects likely to be identified as false positives should be uniquely identified
- REQ 2.2.3-BC** Coordinates of object center must be accurately determined
- REQ 2.2.4-BC** Object detection must be fast to ensure low latency
- REQ 2.2.5-BC** Record position and timing of object identification



## EAGLE EYE SYSTEMS

### 12.0.6 Tracking Requirements

- REQ 2.3.1-ABC** Target objects must be brought quickly and smoothly to the center of the field of view through feedback control system with motors and camera
- REQ 2.3.2-BC** Identified target objects must be held accurately in the center of the field of view through feedback control system with motors and camera
- REQ 2.3.3-BC** Log position of object over time for duration of tracking to construct a smooth path of motion
- REQ 2.3.4-BC** Motor controller will enhance precision of tracking using prediction

### 12.0.7 Environmental and Safety Requirements

- REQ 3.0.1-C** Optional solar power for off grid stand alone system
- REQ 3.0.2-BC** Product will not incur harmful consequences to natural ecosystems in its vicinity
- REQ 3.0.3-C** A discriminator will be trained to help ensure that only actual target is tracked
- REQ 3.0.4-BC** Strong, corrosion resistant, environmentally friendly metals will be used in the construction of the stage
- REQ 3.0.5-BC** Rugged, environmentally friendly plastics will be used in the construction of the stage
- REQ 3.0.6-C** System electronics and mechanics must operate accurately under a broad range of temperatures from -30°C to 60°C
- REQ 3.0.7-C** Product will be water resistant
- REQ 3.0.8-C** Product shall be sturdy enough to withstand wind gusts of up to 60km/h