Jan 28, 2019

Andrew H. Rawicz
School of Engineering Science
Simon Fraser University
V5A 1S6

Re: ENSC 405W Requirements Specification for NightEagle

Dear Dr. Rawica and Professor Craig Scratchley,

We have thoroughly prepared EagleVision's Requirement Specifications report of our NightEagle device for review. Our unwavering goal is to create a portable, efficient, and affordable night-vision human and animal detector to alarm drivers, in order to greatly reduce vehicular fatalities during night time. We will combine bleeding-edge Machine Learning technology into an extremely efficient and portable hardware form-factor. The device will take in visual data from an embedded night vision camera and pass the data through our in-house developed AI in order to correctly identify pedestrians under any lighting conditions.

This document will detail the system overview of said device from its functionalities, system integration, software design, and hardware implementation. It also prescribes all of the mandatory requirements from a general, hardware, and software perspective. In addition, this report will include engineering standards, responsibilities, sustainability, and safety measures to ensure performance, operational ecosystem, and ease-of-handling.

We will follow the development sequence listed in this document and achieve each of the requirements in proof-of-concept, prototype, and final product, based on the outlined priorities. **Please note that the requirements marked as PoC will be strived to be demonstrable at the end of ENSC 405.**

EagleVision currently consists of three engineering students: John Xing, Billy Luo, and John Zhang, all coming from different engineering divergences to ensure the robustness of the final end product. We would like to thank you for taking the time to review our product and giving valuable feedback.

Sincerely,

John Xing
CEO

Enclosed: Requirements Specification for EagleVision's NightEagle

Requirement Specification

for EagleVision's flagship device, the NightEagle

Project team:

John Xing

Billy Luo

John Zhang

Correspondence:

Billy Luo

Email: luosiyal@sfu.ca

Submitted to:

Mr. Craig Scratchley: ENSC 405W

Dr. Andrew Rawica: ENSC 440

School of Engineering Science, SFU

Issue date:

Jan 28th, 2019

Version 1.0.0

Confidential content

Abstract

In a society of rapidly advancing vehicle safety for passengers, manufacturers continue to consider the safety of pedestrians as an afterthought. In fact, nearly 1 in 5 people killed in car crashes every year in BC are pedestrians, and most of these deaths are preventable, according to the Insurance Corporation of British Columbia [1]. The same article quotes that "Although late night/early morning pedestrian collisions (midnight to 6:00 am) are relatively infrequent, they have almost double the risk of fatality." The sad reality of this sobering reality is the fact that even though the automobile has been on the market for over a century and vast technological advances have been made, the indirect safety of the pedestrian has been forgotten in wake of engineering innovation. To further supplement this, it has been reported that hit-and-run fatalities has been increasing at an average rate of 7.2 percent per year since 2009 [2]. This highlights the detrimental nature of night-driving, as drivers rather abandon the dying victim than face jail time, in hopes that the cover of darkness will aid their escape from prosecution. Evidence clearly shows that the safety of innocent pedestrians has, unfortunately, not proportionally grown to vehicular innovation, and there is a gigantic void of public concern waiting to be filled by technology.

While such collision prevention mechanisms may exist on a small selection of top-end luxury cars, they utilize the main cameras of the front and back sections, which are only adapted for road conditions in broad daylight or well-light night roads. Very few mainstream vehicles implement a very complex pedestrian detection system, and are usually no more than simple proximity sensors. In addition, the majority of drivers are still drive older mid-range options, which do not even offer any type of automated alerts. This specification document will serve as a high-level exploration of the proposed solution to pedestrian fatalities due to vehicular collisions, especially at night, for high-speed vehicles of all kinds. EagleVision's mission is to minimize or greatly reduce night-time vehicular collisions with pedestrians by implementing a high-effective yet affordable device, which will recognize pedestrians even in the darkest of circumstances and alert the driver. EagleVision will construct this compact and fully embedded device by combining a high quality night-vision sensors, a micro-PC board, and cutting-edge self-developed machine learning software which reacts in real time. Looking from a top-perspective, the system will pass in the input data through the camera sensor and be decoded by the on-board CPU of our motherboard. The decoded string will then be sent to the concurrently running AI, which will detect and identify pedestrians based on their facial features and send an external audio/visual alert to the user.

This document will serve as an overarching guide to both the design language and technical specifications needed to complete the implementation process on each individual piece of the puzzle. However, completion of the end goal is impossible without an integration plan, thus multiple sections will be also dedicated to the systems design of the overall architecture. Finally, the sustainability and safety of the end product will be explored.

Table of Contents

List of Figures

List of Tables

Glossary

Term	Definition
Opencv Library	An open source coding library containing training datasets for software integration
Machine learning AI	Artificial Intelligence based on repeated training cases of software to recognize visual queues
Night-Vision Device	Conversion to visible light of both visible light and near-infrared using an image intensifier tube, to produce a monochrome feed
Feature descriptor	An algorithm which takes an image and outputs feature descriptors/feature vectors
RPi	Raspberry Pi board

1 Introduction

1.1 Background

Driving is a very complicated task that requires the driver to not only clearly identify the various road conditions, but to respond quickly in an instant. Especially at night, driving visibility is greatly reduced, which will impact the driver's ability to recognize and respond to their environment. The most notable observational survey shows that the severity, which is defined as the ratio of the number of fatal collisions per 100 collisions, is doubled at night.[3] The ratio of fatal incidents per 100 million miles at night in 1990 is high at 10.37, yet only 2.25 during the day.[4] Therefore, the darkness can be confidently theorized as the main factor which impedes drivers' abilities to identify and avoid road collisions. For many years, this issue has caused more than half of all traffic accidents, especially for pedestrians. In 1912, the first traffic light was invented by Lester Wire, a Salt Lake City policeman.[5] Although there are traffic lights on almost every major street, there are still many trails and crosswalks without any pedestrian or street lights. As a result, research shows that the overall severity of injuries is tripled on roads without lights,[6] Even though certain high-end vehicle brands in modern times incorporate some form of frontal collision detection technology, the R&D industry is still in its development stages. In addition, advanced human detection technologies only exists for concept self-driving cars of the future, and are nowhere to be found on common modes of transportation, such as mass-production cars and motorcycles.

EagleVision aims to provide a safer environment for the current traffic ecosystem. The company is committed to enabling all drivers - not only high-end luxury vehicle owners, to drive safely with its patented frontal human detection device. OpenCV, a library built for developing real-time computer vision systems, provides Haar cascade classifiers. Haar cascade classifiers are based on a machine learning approach, which can be trained from both positive (image with faces) and negative (image without faces) sample spaces. The Haar classifier will take in thousands of samples and the code will attempt to search for the face by analyzing each pixel in the image.[7] Through countless iterations of incremental improvements, the AI will become fast and accurate at recognition. The code will also use Histogram of Oriented Gradients (HOG), a feature descriptor for computer vision and image processing, to count occurrences of gradient orientation in localized portions of an image and detect the human body.[8] In terms of hardware input, a night-vision capable camera will be used to record the video stream and output frame-by-frame bitmaps, so that the software component can detect the faces and bodies in each sample. As for portability and ease of usage, the system will incorporate a micro-computer such as Raspberry Pi as the computer at the heart of the device. It is a single-board computer consisting a CPU, GPU, RAM, Ethernet ports, GPIO pins, Xbee sockets, UART, power source input, along with various interfaces for connecting other external peripherals.[9] Raspberry Pi gives the development team at EagleVision a powerful set of tools found only on full-sized PCs, but in a minimal size and cost factor.

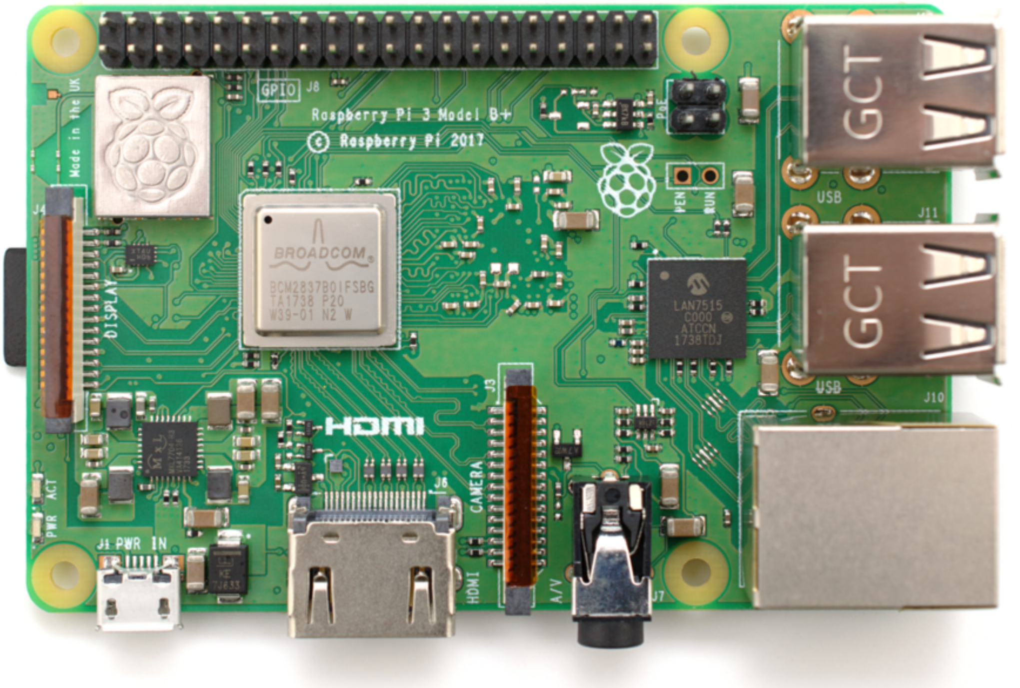

Figure 1: Raspberry Pi 3 Model B+ [10]

As machine learning seeps deeper and deeper into today's technological innovation, its various promising uses are applied in all sectors. However, as society rapidly improves quality of life with tech, one must always consider not only the safety of the end-user, but also the population affected by its usage. EagleVision is confident that its flagship device will disrupt the market with its affordability and ease of usage, and more importantly, help shape a future of absolute road-safety aided by AI.

1.2 Requirement Classifications

NightEagle's requirement classification is shown below:

[Domain-(num)Priority]	Requirement description

Table 1: Requirement Classification Format

Where num is the order of the requirement, with 1 being the highest

Domain is classified as:

- **GE** - General
- **HW** - Hardware
- **SW** - Software
- **ES** - Engineering Standard
- **SS** - Safety/Sustainability Standard

Priority is classfied as:

- **H** - High: Proof of Concept - Basic requirement and functionality which need to develop firstly
- **M** - Medium: Prototype - Completed functionality which need to improve
- **L** - Low: Final Product - Professional features and performance improvement.

1.3 Scope

This document serves as an overview and analysis of the various functional requirements of the NightEagle device, which EagleVision will strive for during the various design stages. The analysis will be divided into Systems Overview (integration), detailed Hardware, and detailed Software. Finally, the specific requirements that will be met for each category will be explored, listed, and justified. The requirements will also indicate whether it is meant for PoC, Engineering Prototype, or Final Product. Lastly, the Engineering Standards and Sustainability/Safety aspects, as well as the Acceptance Test Plan will be thoroughly accounted for.

1.4 Intended Audience

This report functions as the guiding principles at all design stages for EagleVision's employees, and every prototype will be evaluated according to the outlined specifications. In addition, this document is ready to be evaluated by Professor Craig Scratchley, Dr. Andrew Rawicz, the teaching assistants, as well as potential clients and investors.

2 System Overview

2.1 Scope and Functionality

The device uses a night vision camera as the main input device. The video feed recorded through the camera lense will produce an encoded and continuous digital signal which is then passed along through the micro-motherboard into the concurrently running code. The software element embedded into the board will then decode the image feed frame by frame into a format recognized by the code and begin analyzing it. The complex software backend will detect all the humans in the frame and draw an alert window around the detected objects. It will then re-encode the analyzed feed and transmit it through the hardware, which is connected to a mini-screen. The end result will be an extremely efficient clockwork of hardware and software components, as the screen will display the video feed as the camera sees it, with all the humans detected in real time. Additional hardware alerts can be implemented, which includes sound and visual alerts when the software picks up a specified object.

2.2 Proposed Design

2.21 Software/Hardware Integration Systems Overview

This section provides a brief summary of the methods designed to take full advantage of the software and hardware components. As can be seen from Figure 1, the *final product design* will implement such an embedded system in order to compact all the various pieces into a small and portable factor. There are several types of boards that can be used to integrate software from hardware, and Raspberry Pi is amoungst the most robust and developed of the lot. While more detailed technical analysis will be performed in Design Specifications, this document will mainly use the Raspberry as a base to explain the fundamental requirements.

Looking at the diagram, the Power Input can be interpreted as the main "on/off" switch of the entire system. After power is supplied, the main computer board should send the signal to all peripherals, including the camera, the screen, and additional hardware alert options. The micro-computer also concurrently loops the code, which is designed to output to the various hardware components. The code will decode the camera feed and perform human detection on it. The video is then re-encoded and passed onto the external screen in real-time, so the driver can see the recognized objects being highlighted. More detailed inspections of the hardware and software component will be discussed in the following section.

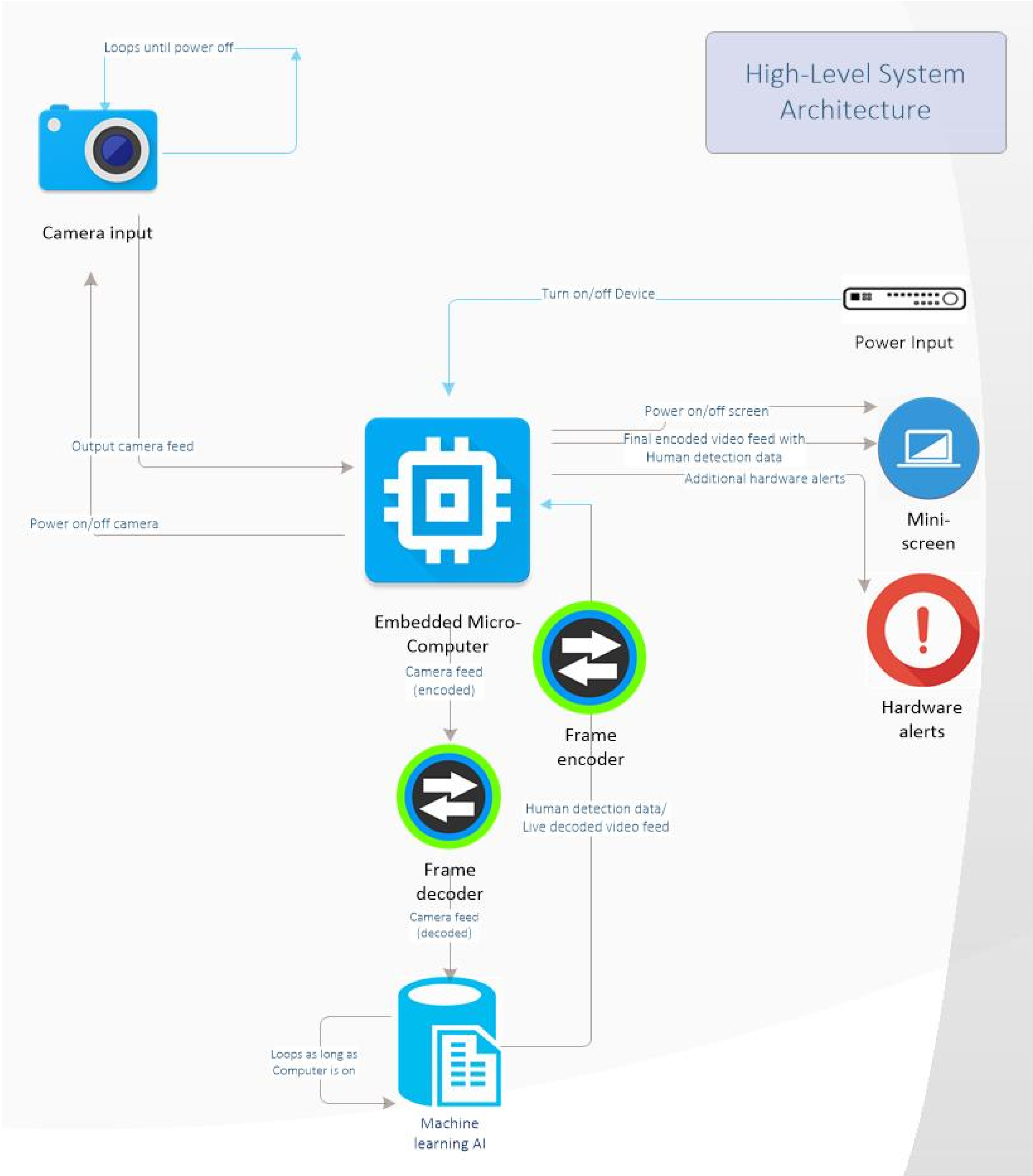

Figure 2: High-level systems architecture of entire device recreated through Visio

2.21 Hardware Indepth Overview

As mentioned above, this document will use the Raspberry Pi 3 as a basis for the hardware core of the device. Since it is a very versatile and developed board, it comes with many advantages. As can be seen from Figure 3, there is a specified I/O for camera modules. This means that most of the hardware components can be connected directly to the board, which would in turn be fed through to the software.

Figure 3: Common camera module I/O for Raspberry [11]

In order to keep all the components packed in tightly, a hard outer shell will be applied. There exists various aftermarket options such as in Figure 4, which are designed to encase a Raspberry board with the appropriate ports. The screen and other hardware components will be secured to the outside of the casing, so it would be a rigid and solid device after production.

Figure 4 - Casing option for final production [12]

In addition, there will be an external screen which will serve to transfer the feed processed live by the software. There are several smaller 5 inch options which would work well with the current model. As the Raspberry comes with HDMI options, it will be compatible with most models. The screen will be secured to the top of the device to ensure stability.

Figure 5 - Mini-screen for Device [13]

After examining the nature of each hardware component, the hardware architecture can now be combined to form the hardware core of the device. Taking a look at Figure 6, all the different components have been defined and allocated on the board. As seen in the previous section, Power Input will regulate the on/off status of the system, and the CPU/memory block houses the software, which will control the data fed to the other hardware options. This should create the end result of an efficient and compact system, where all the fringe components are secured close to the main board, which will prevent parts from coming loose.

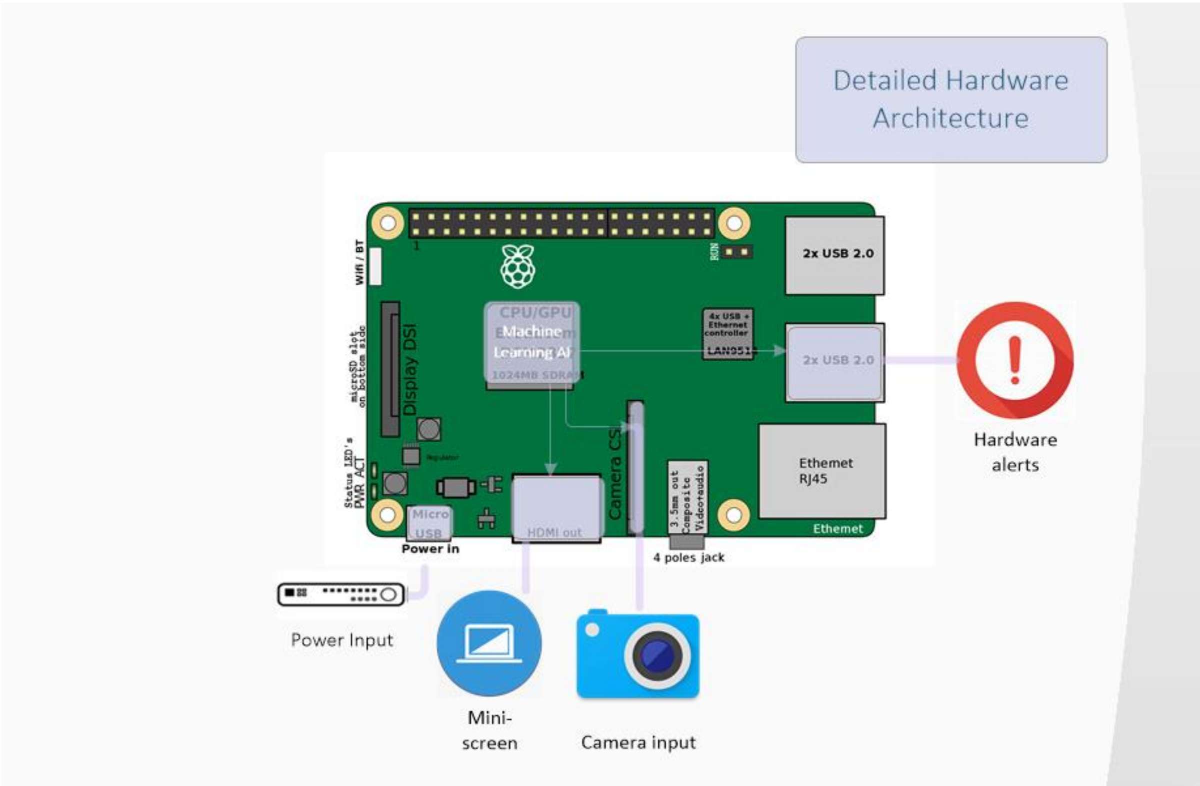

Figure 6 - Hardware implementation recreated through Visio; based on Raspberry Pi 3 diagram [14]

2.22 Software Backend Indepth Overview

In terms of software, the code will focus on human detection technology from the emerging deep learning industry. With proper integration, it is a versatile computer vision technology that identifies the locations and sizes of human faces or bodies with high accuracy automatically.

The system will make use of Haar feature-based cascade classifiers from the OpenCV library, which is a collection of programming functions mainly aimed at real-time computer vision. Recognized by the industry, Haar cascade classifier is one of the most efficient and robust object detection ecosystems which allows users to train AI from a variety of positive (image with particular objects) and negative (image without the specified objects) samples.[15] After going through a large number of error-elimination iterations, the code will be able to detect any target object accurately.

When the camera turns on, the detection system on the software backend will read in the captured video frame-by-frame, convert the frames into grayscale, and store them into a dynamic list. Afterwards, it will draw a rectangular outline on each target object in the video to accurately indicate the location and distance of the object in the real time on the hardware screen. More importantly, the software will send an alert to the hardware side, so that the driver will take notice when a pedestrian is nearby.

For the final product, additional useful software features such as Google Maps, GPS, and

Dash Cam functionalities will be implemented to improve the versatility of the design.

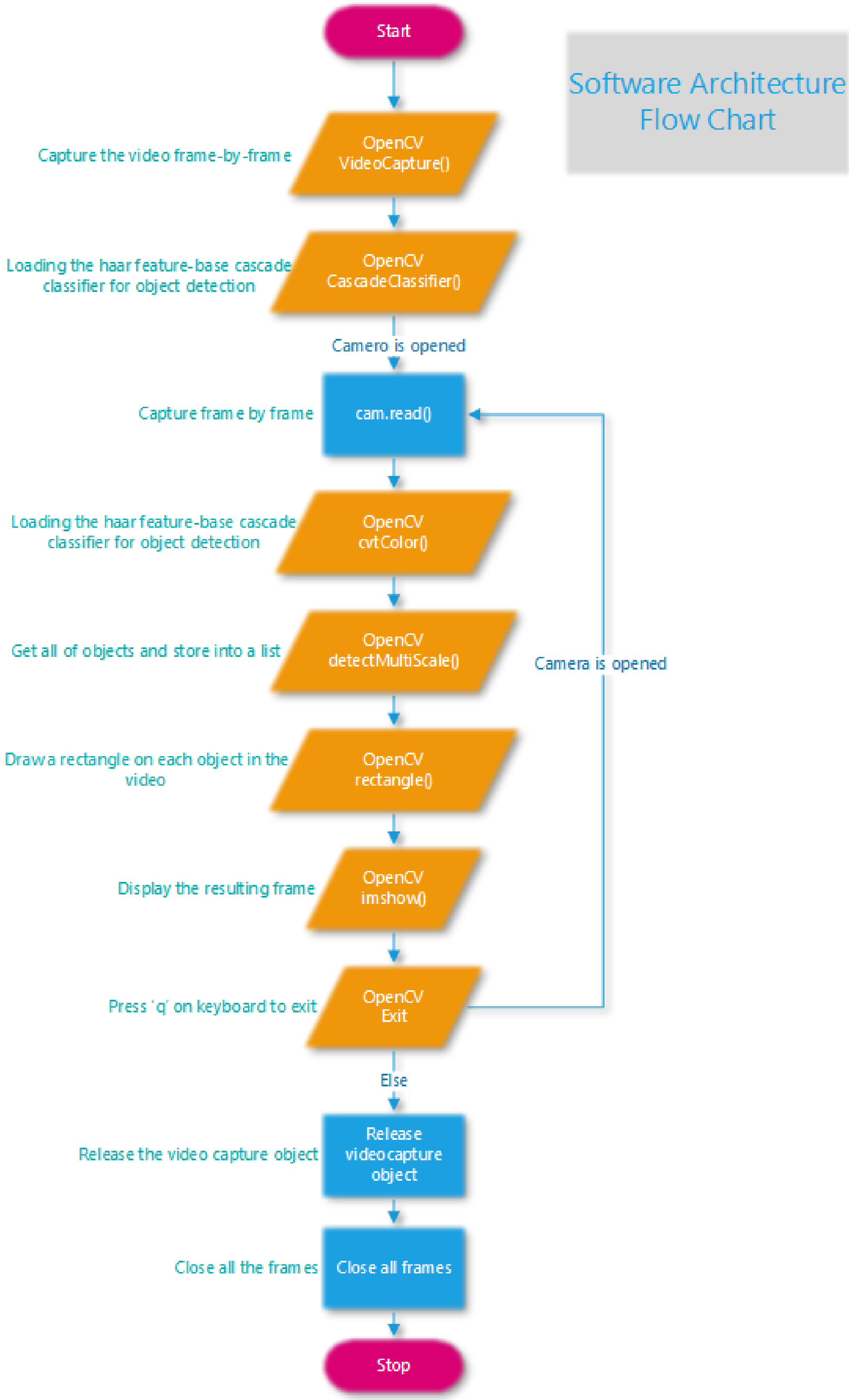

Figure 7 - Software Architecture Flow Chart of Machine Learning

3 Requirements

3.1 General Requirements

Where P = PoC, E = Engineering Prototype, F = Final Product

Requirement	Detail	Justification	P	E	F
GE-1H	Device shall highlight and alert user when pedestrian detected in any dark environment	Main purpose of device	x	x	x
GE-2H	Device shall have very small error margin and be able to recognize humans without fail	Need device to have low error rate to guanratee safety		x	x
GE-3H	Device should be able to recognize objects with very little delay (near real-time)	Critical for situations when user is driving at high speeds		x	x
GE-4H	Device should be able to recognize objects under unstable camera conditions	Critical for situations when user is turning corners or driving on bumpy roads			x
GE-5M	Device should be rigid and all hardware components should be well protected	Need to ensure device can hold up under all driving conditions			x
GE-6M	Device should be compatible with most vehicle power extensions	Necessary to be ready-to-use for a wide range of vehicles			x
GE-7L	The UI should be simple and easy to operate by average consumers	Should limit usage cases of Raspberry board and OS			x
GE-8L	Device should be suitable to use with variety of car mounts	Provides more options to consumers			x
GE-9L	Additional feature - Google map	Feature improvement			x
GE-10L	Additonal feature - GPS system	Feature improvement			x
GE-11L	Additional feature - dash cam	Feature improvement			x

Table 2: General requirements

3.2 Hardware Requirements

This following requirements shall be used as a guideline for the product hardware design. They will be achieved for the necessary hardware operation.

Requirement	Detail	Justification	P	E	F
HW-12H	Raspberry Pi shall operate properly	Hardware core, the brain of the device	x	x	x
HW-13H	Night vision camera shall work properly	Main input component	x	x	x
HW-14H	Signal shall have accurate feedback	Main input component	x	x	x
HW-15M	Stable battery with sustainable life cycle	Performance - Safety requirement			x
HW-16H	Additional feature - touchscreen	Feature improvement			x

Table 3: Hardware requirements

3.3 Software Requirements

This following requirements shall be used as a guideline for the system software design. They will be achieved for the necessary software functionality.

Requirement	Detail	Justification	P	E	F
SW-17H	Program shall detect objects properly	Main purpose of the entire program; Need to recognize target object	x	x	x
SW-18H	Program shall alarm the driver the detection objects properly	Main purpose of the program; Need to send alarm signal	x	x	x
SW-19H	Program shall detect objects accurately with no error	Necessary for device to function properly		x	x
SW-20H	Program should be able to recognize objects in real-time	Necessary for device to function properly			x
SW-21H	Program should be able to alert the driver in real-time	Necessary for device to function properly			x
SW-22M	Program should require minimum power consumption	Increases battery life			x
SW-23M	Program should have minimum size and run time	Safety requirement - Prevent overheating; Minimize power consumption			x
SW-24L	Program should have a simple user interface	User experience			x

Table 4: Software requirements

4 Engineering Standards & Responsibilities

Night Eagle will follow engineering standards shown below for both of design and safety process in the development to produce a powerful, stable, and safe product. It includes safety, product material, and software development areas in the table.

Requirement	Detail	Justification	P	E	F
ES-25H	All designs shall comply with the IEEE safe engineering methodologies. [16]	General safety standard	x	x	x
ES-26H	Software development shall follow Google's coding standards for programming and scripting languages. [17]	Quality of software development	x	x	x
ES-27M	All major components, including the outer shell, shall adhere to UL 94 - the Standard for Safety of Flammability of Plastic Materials for Parts in Devices and Appliances Testing. [18]	Material safety component			x
ES-28M	All electronic components and the final product shall adhere to RSS-102 - Radio Frequency (RF) Exposure Compliance of Radio communication Apparatus. [19]	Necessary to ensure signal radiation exposure to end user is safe and sustainable, as well as avoid interference with external radio waves			x

Table 5: Engineering standards

5 Sustainability & Safety

The product will strictly adhere to the electronic product sustainability standards and user safety standards. The details are described below.

5.1 Sustainability Standards

Sustainability standards strive to ensure materials consumed in product manufacturing are able to be reused in the development cycle. In order to achieve this goal, engineers have to not only consider developing a robust product, but also maintain product material sustainability.

Requirement	Detail	Justification	P	E	F
SS-29H	Checking component manufacturer sustainability and fair labour policies	Environmentally-friendliness and other social ethics shall be a priority towards business partner determination	x	x	x
SS-30H	Reusing components in the PoC prototype to build the prototype	Waste reduction	x	x	x
SS-31H	Reusing components in the prototype to produce the first-piece device	Waste reduction		x	x
SS-32M	Providing environmental impact and recycling information to distributors and end-users	Environmental awareness			x

Table 6: Sustainability standards

5.2 Safety Standards

Another most important part of the product is safety. Developers have to ensure that the product is completely safe for use. It should not have any possible risks to cause physical damage to the end user. The following requirements shall be used as a guideline for the product safety standards.

Requirement	Detail	Justification	P	E	F
SS-33H	Prevent short circuit	Basic operation requirement	x	x	x
SS-34H	Prevent battery overheating and battery blowing up	Main safety requirement	x	x	x
SS-35H	Product exterior shall not have sharp edges to avoid injury risk	Ensures no injuries if product falls during operation			x
SS-36M	Product shall be encapsulated in case for the moving part injury risk	Product assembly requirement			x

Table 7: Safety standards

6 Conclusion

This document introduced the backbone of the overall device with the system overview, prescribing the product functionality, integration, hardware, and software aspects. It also details the requirement specifications of the general, hardware, and software components for the various stages of development. Lastly, the engineering, sustainability, and safety standards will be stated to ensure the quality, safety, and environmental sustainability of the final product.

EagleVision Inc. plans to finish the proof of concept (PoC) and Appearance Prototype in April 2019, and the final product in August 2019. The acceptance test plan is included in the appendix at the end of this document. It will be used as a testing standard to guide the development of the product in the intended direction.

Through the different skill sets and attributes of each members in the team, EagleVision strives to create a safer environment for all drivers and pedestrians. The NightEagle device will serve as an affordable and easy-to-access product for any vehicle on the road, ensuring the reduction of night-time vehicular collision fatalities.

7 References

[1] “Pedestrian Injury from Motor Vehicle Crashes in BC”, BC INJURY, 2017. [Online]. Available: https://www.injuryresearch.bc.ca/wp-content/uploads/2017/05/Injury-Insight-Pedestrian-Safety-May-2017-FINAL.pdf

[2] “Hit-and-Run Crashes: Prevalence, Contributing Factors and Countermeasures”, Foundation for Traffic Safety, 2018. [Online]. Available: https://aaafoundation.org/wp-content/uploads/2018/04/18-0058_Hit-and-Run-Brief_FINALv2.pdf

[3] S Plainis, I J Murray, and I G Pallikaris, “Road traffic casualties: understanding the night-time death toll”, NCBI, 2006. [Online]. Available:https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2564438/

[4] Dawn L. Massie and Kenneth L. Campbell, “ANALYSIS OF ACCIDENT RATES BY AGE, GENDER, AND TIME OF DAY BASED ON THE 1990 NATIONWIDE PERSONAL TRANSPORTATION SURVEY”, deepblue.lib.umich.edu, 1993. [Online]. Available:https://deepblue.lib.umich.edu/bitstream/handle/2027.42/1007/83596.0001.001.pdf?sequence=2;analysis

[5] “Traffic light”, En.wikipedia.org, 2019. [Online]. Available: https://en.wikipedia.org/wiki/Traffic_light

[6] S Plainis, I J Murray, and I G Pallikaris, “Road traffic casualties: understanding the night-time death toll”, NCBI, 2006. [Online]. Available:https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2564438/

[7][15] “Face Detection using Haar Cascades”, OpenCV, 2018. [Online]. Available: https://docs.opencv.org/3.4.1/d7/d8b/tutorial_py_face_detection.html

[8] “Histogram of oriented gradients”, En.wikipedia.org, 2019. [Omline]. Available: https://en.wikipedia.org/wiki/Histogram_of_oriented_gradients

[9] “Know all about Raspberry Pi Board Technology”, EDGEFX KITS & SOLUTIONS, 2017. [Online]. Available: https://www.edgefxkits.com/blog/raspberry-pi-technology-with-applications/

[10][14] Raspberry Pi 3 Model B+ [Digital image]. Available: https://en.wikipedia.org/wiki/Raspberry_Pi

[11] Common camera module I/O for Raspberry [Digital image]. Available: https://www.google.com/url?sa=i&rct=j&q=&esrc=s&source=images&cd=&ved=2ahUKEwjgwtTkwongAhUEiFQKHVVuBPIQjRx6BAgBEAU&url=https%3A%2F%2Fcore-electronics.com.au%2Fraspberry-pi-wide-angle-camera-module-seeed-studio.html&psig=AOvVaw3lriLLIHJpcBbvXu0nRb54&ust=1548525906045421

[12] Raspberry Pi 3 Case Black and Grey [Digital image]. Available: https://www.umart.com.au/Raspberry-Pi-3-Case-Black-and-Grey_39940G.html

[13] Raspberry Pi Official Screen Case - White [Digital image]. Available: https://www.robotistan.com/raspberry-pi-official-screen-case-white

[16] "PSES IEEE - Product Safety Engineering Society", Ewh.ieee.org, 2019. [Online]. Available:
http://ewh.ieee.org/soc/pses

[17] "Google Python Style Guide", Google.github.io, 2019. [Online]. Available:
https://google.github.io/styleguide/pyguide.html

[18] “Standard for Tests for Flammability of Plastic Materials for Parts in Devices and Appliances”, ul.com, 2019. [Online]. Available:
https://standardscatalog.ul.com/standards/en/standard_94_6

[19] “RSS-102 — Radio Frequency (RF) Exposure Compliance of Radio communication Apparatus (All Frequency Bands)”, ic.gc.ca, 2019. [Online]. Available:
https://www.ic.gc.ca/eic/site/smt-gst.nsf/eng/sf01904.html

8 Appendix

Acceptance Test Plan

General Requirement Test		
Device detects objects and alerts user in dark		Comments:
☐ Yes	☐ No	
Device operate without error		Comments:
☐ Yes	☐ No	
Device detects object in real-time		Comments:
☐ Yes	☐ No	
Device detects object under unstable camera		Comments:
☐ Yes	☐ No	
Rigid and well protected		Comments:
☐ Yes	☐ No	
Compatible most vehicle power extensions		Comments:
☐ Yes	☐ No	
Easy to install		Comments:
☐ Yes	☐ No	
Addition feature - Goolge map		Comments:
☐ Yes	☐ No	
Additional feature - GPS system		Comments:
☐ Yes	☐ No	
Additional feature - dash cam		Comments:
☐ Yes	☐ No	

Hardware Requirement Test	
Raspberry Pi operate properly	Comments:

☐ Yes	☐ No	
Night vision camera work properly		Comments:
☐ Yes	☐ No	
Accurate signal feedback		Comments:
☐ Yes	☐ No	
Stable battery with sustainable life cycle		Comments:
☐ Yes	☐ No	
Touchscreen operates properly		Comments:
☐ Yes	☐ No	

Software Requirement Test		
Program detects objects properly		Comments:
☐ Yes	☐ No	
Program alerts properly		Comments:
☐ Yes	☐ No	
Program detects objects without error		Comments:
☐ Yes	☐ No	
Program detects objects in real-time		Comments:
☐ Yes	☐ No	
Program alerts in the real-time		Comments:
☐ Yes	☐ No	
Minimize program power consumption		Comments:
☐ Yes	☐ No	
Minimize program size and runtime		Comments:
☐ Yes	☐ No	
Simple UI		Comments:
☐ Yes	☐ No	