Senior Design 1

Automatic Face Tracking Camera

High Level Design Rough Draft

3/13/22

Winter 2022 Semester

Team 12: All Stars

Nicholas Frazer, Hesham Alaidarous

Advisor: Paul Watta

1. Introduction

1.1 Purpose

The automatic face tracking camera is designed as a proof of concept for a hands free device that will automatically follow and record the user.This working prototype will be designed with taking stationary pictures without the user having to physically interact with the device. With a more powerful processor the camera could be able to stream video as the device physically follows the user which will be useful in this era of video conferences and zoom classes. However with our prototype it will be designed more for consumers who take pictures of themselves often and alone.This device will also serve as a proof of concept for further development with more powerful processors.

1.2 Definitions

Computer Vision: a field of artificial intelligence that trains computers to interpret and understand the world

1.3 System Description

The Automatic Face Tracking Camera is a device designed to recognise the user's face and track their position so that they are always in the center of the frame and automatically take their picture. This will allow the user to set the device down and walk away while the camera automatically repositions to have the person centered and take a photo after they have been still long enough. The device will begin tracking the user once their face has entered the camera's field of view. This device is designed with a single user in mind.

System objectives

1. Automatically shift the camera to follow the user's face
2. Take a photo
3. Notify that the task has been completed.
4. Read information from a video source

5. Recognise faces within frames of the video feed
6. Relay that positional data
7. Motors will physically reposition the camera to follow the user's face based on where they are moving within the frame of the taken images..
8. Check how long the user has remained in the same spot so that the camera can capture a picture
9. Picture will be stored within the device's memory to be transferred later.

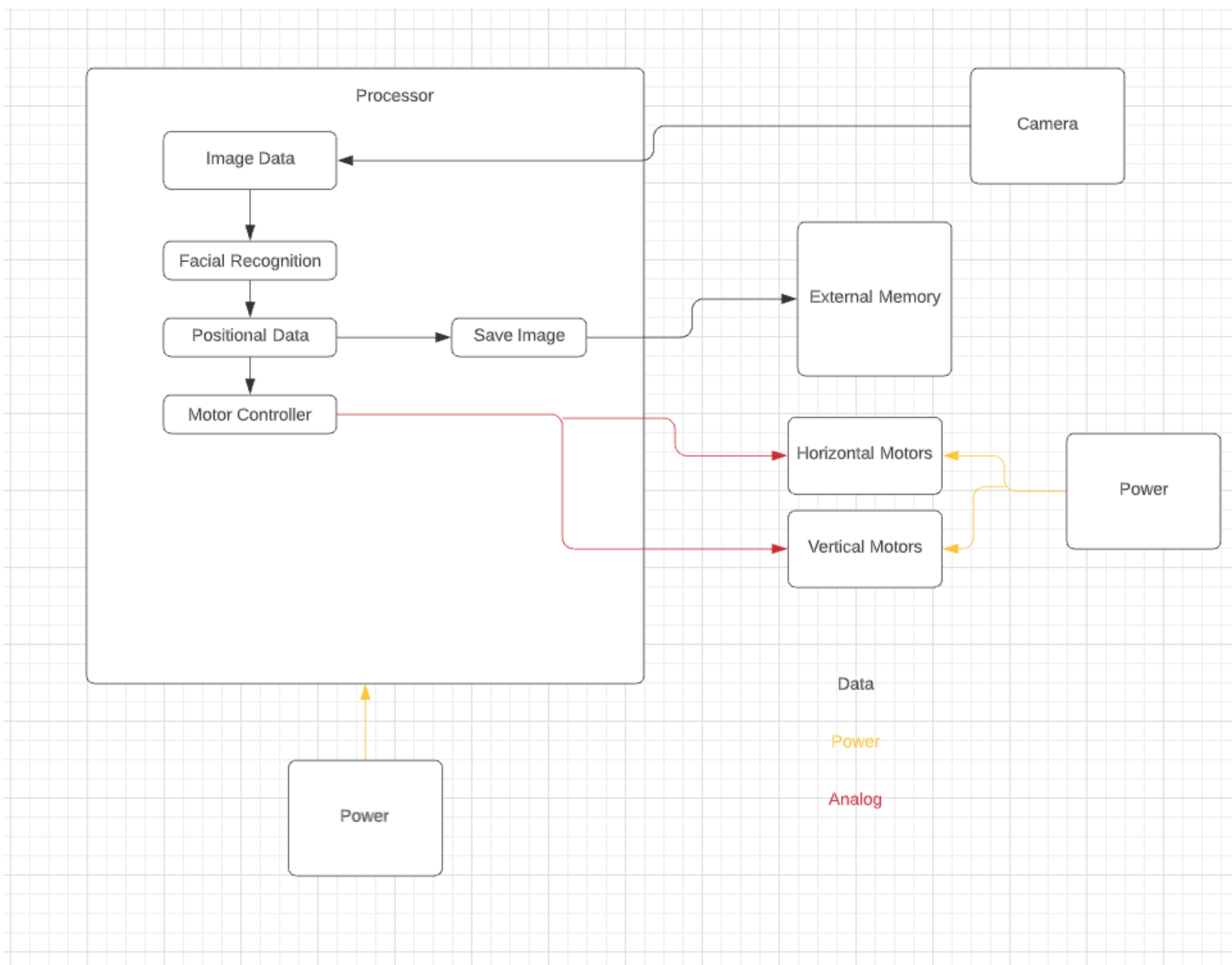

Figure 1 Module Block Diagram

Modules

1. **Camera** will capture images within its field of view
2. **Facial Recognition**: the image data will be analyzed and faces within these images will be identified.
3. **Parse:** Parsing the positional data and interpret coordinates to send to motors
4. **Motor controlling**: The motors will be controlled to follow the user
   Vertical and horizontal motors will receive analog data from the motor controller and spin accordingly
5. **Memory management**. The device will save certain frames to the memory when the user has stood still for long enough.

1.4 References

Python standard library: https://docs.python.org/3/library/

OpenCV vision libraries: https://github.com/opencv/opencv

2. System Constraints

2.1 Environmental Constraints

This device will be designed as a test bench prototype that will undergo no extreme environmental constraints. This will work under normal room temperature and will not be waterproof. It will undergo no major shock and vibration. It will work under ideal lighting conditions in a well lit room.

2.2 size, weight, cost, power, constraints

The device will be approximately 10x10x10 inches and weigh around 0.75 pounds. The device itself will be designed to be either mounted on a tripod or sit on a flat surface. These are not included within the size or weight of the device. As seen in the block diagram, the microprocessor and motors all require power. These 3 components will draw less than 20 watts of power each.

2.3 reliability and safety considerations

This device has no major safety concerns due to it using only low voltages and requiring no large power needs. This also applies to any concerns relating to the damaging of components as we are working with low voltages. These parts are also fairly replaceable except for the microcontroller.

2.4 Site information

This will be designed as a test bed prototype in an ideal laboratory environment under the conditions described under environmental constraints

3. High Level System Design

3.1 Acquiring image traces to requirement 3.1

The camera will capture images and send them to the raspberry pi via the DSI connection and stored in the capture structure for use in the other modules.

3.1.1 Hardware: Camera
Vendor: Arducam
Model: Arducam 5mp camera OV5647
Cost: $9.99

3.1.2 Software: **Processing Images** : The python library picamera will be used to interface the camera with the raspberry pi and will allow us to control aspect ratio, limit FPS, and retrieve data. The data is continuously captured using Picamera's video capture function and stored within Picamera's capture structure. The video capture function continuously captures and overwrites frames, which will be stored in a frame structure using Picamera's read function. That will be used by the other modules such as the facial recognition module

3.2 Facial Recognition: Traces to requirement 3.1 and 3.4
This module will analyze the images gathered from the camera module. The images will be analyzed looking for faces within the images. From this positional information will be gathered from each frame. Using the HAAR frontal face and upper body detection algorithm paired with the detectmultiscale functions within the OpenCV libraries we will detect faces. OpenCV inputs the frames from the Image acquisition module and outputs an image structure. The relevant data within this image structure is the x and y coordinate position of the upper left and corner of the detected face as well as the height and width of the face.

3.2.1 Hardware: Graphical microprocessor
Model Raspberry pi 4
Vendor: Raspberry pi
Cost: $35

3.2.2 Software: **Face detection setup**:
The cascade path is set using the cascade clarifier function of OpenCV. This allows us to set the algorithm that will be used once the facial detection begins. The algorithm used will be the HAAR frontal face and upper body algorithms which will detect the user's general position within the frame from their upper body and the front of their face. The cascade method analyzes the photo using line/edge detection.

3.2.3 Software: **Detection**:

Input image data from previous modulus

DetectMultiscale used after the cascade clarifier is set to begin analyzing individual frames using the set algorithm

If a face is detected within a given frame a new image structure will be saved and used in further modules for its X, Y, Height, and Width values.

3.3 Interpreting coordinate data: Traces to Requirement 3.1 and 3.2

The X, Y, width, and height values will be transferred from the previous module to a separate microcontroller through UART Serial communication. This is to alleviate some of the processing being done by the Raspberry pi, allowing it to only input and process new images and not interpreting data and controlling motors. Once the positional data is sent to the Arduino it will be converted from planar coordinates to angles to be used in motor control.

3.3.1 Hardware: Microcontroller

Vendor:Arduino Mega

Cost: $30

3.3.2 Software: **Serial Communication:**

The Raspberry pi will open serial communication with the arduino through a USB UART Serial connection at a baud rate of 9600. The arduino will begin serial communication at the same rate using the arduino's built in serial libraries.

3.3.3 Software: **Coordinate Conversion:**

The arduino will convert the X and Y coordinates sent from the Raspberry pi to angles using the equation X/(Width/180) used for the horizontal axis and Y/(height/180) for the vertical motors. 180 is used as it is the maximum angle that the motors can turn at.

3.4 Motor Control: Traces to requirement 3.2

Servo motors will receive signals through the arduino servo library based on the input coordinates. These signals will be used to control two servo motors, one that controls horizontal movement and one that controls vertical movement. These motors can easily be driven through the Arduino alone, as was demonstrated in our first excursion.

3.4.1 Hardware: Vertical Servo Motor

Model: MG996R

Vendor: Deegoo

Cost: $10

3.4.2 Hardware: Horizontal Servo Motor

Model: MG996R

Vendor: Deegoo

Cost: $10

3.4.3 Hardware: Mounting

The Horizontal servo motor will be mounted to a stationary object to keep it standing upright and the vertical motor will be mounted to the horizontal servo motor. The camera will then be attached to the vertical servo motor.

3.4.4 Software: **Control Signal:**

Using the servo.h library that comes default within arduino’s library it will send signals through the PWM using the servo write function given the angles from the previous module.

3.4.5 Hardware: Microcontroller
Vendor:Arduino Mega
Cost: $30

3.5 Memory Management: Traces to requirement 3.3
The positional data will be parsed and interpreted within a second microcontroller after the data has been transferred. Data will be saved once the user has remained still for at least 5 sill

3.5.1 Hardware: SD Card
Model: 32GB 3D NAND High Speed MicroSD Card
Vendor:Silicon Power
Cost: $10

3.5.2 Software: **Data Saving**:
Will save images to the SD card within the raspberry pi
Saves images after the user has remained centered for 30 seconds
Will save at least twenty images before the user needs to take further action
User can view images and manage data

3.6 Power
The system will be powered by 5v AC power adapters.

3.6.1 Hardware: 2 Power Adapter
Model: Universal AC/DC Adapter Multi-Voltage Regulated Switching Power Supply
Vendor:SoulBay
Cost: $15

4. Interface Description

The central microprocessor will be the graphics processor (Raspberry Pi 4). From this central processor the camera will be connected to it via a DSI connection. From the processor the image data can be sent to the external memory via the Raspberry Pi’s built in SPI SD card port if the conditions have been met for saving images. After the positional data has been acquired using the OpenCV library within the Raspberry Pi that data will be transferred via UART serial communication to the secondary processor (Arduino Mega) through the USB connection. From the secondary processor the data will be parsed and interpreted within the Arduino IDE and sent to the Servo Motors through a PWM connection. All components will be powered by mains electricity and limited through 5 V AC power adapters.

Hardware interfaces

1. Camera-raspberry pi: DSI connection
2. Raspberry pi-Memory: SPI connection
3. Raspberry pi-Arduino: UART Serial Communication
4. Arduino-Servo motors: PWM
5. Power-Motors/Arduino/Raspberry pi: 5v AC power supply

Software Interface

1. Image Capture-Facial Recognition: Image structure
2. Facial Recognition-Serial Communication: X, Y, h, w floating point
3. Serial Communication-Coordinate Conversion: X, Y, h, w float
4. Coordinate Conversion-Motor Control: Angle float

Hardware Interface

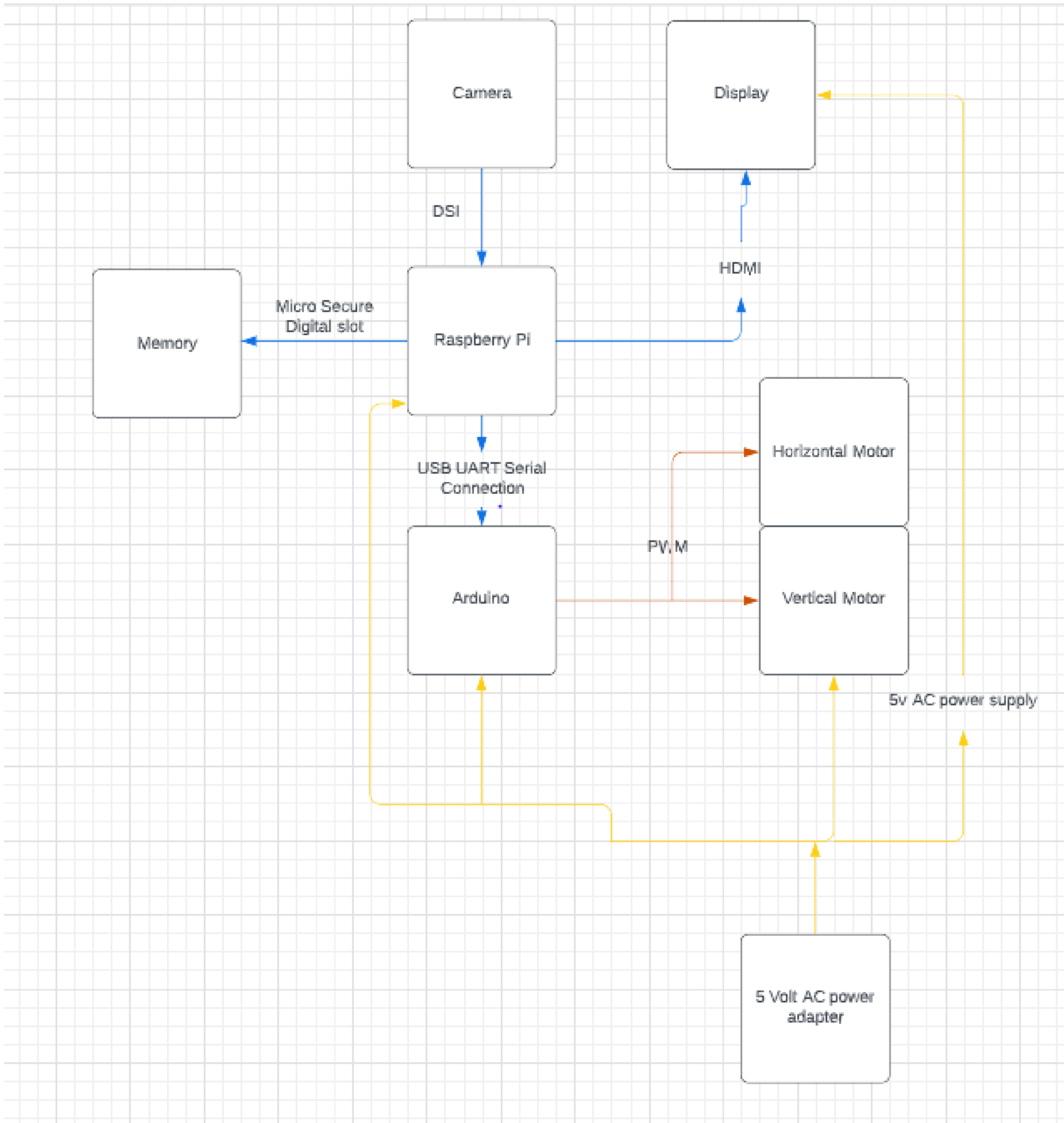

Figure 3 Hardware Interface

Software Interface

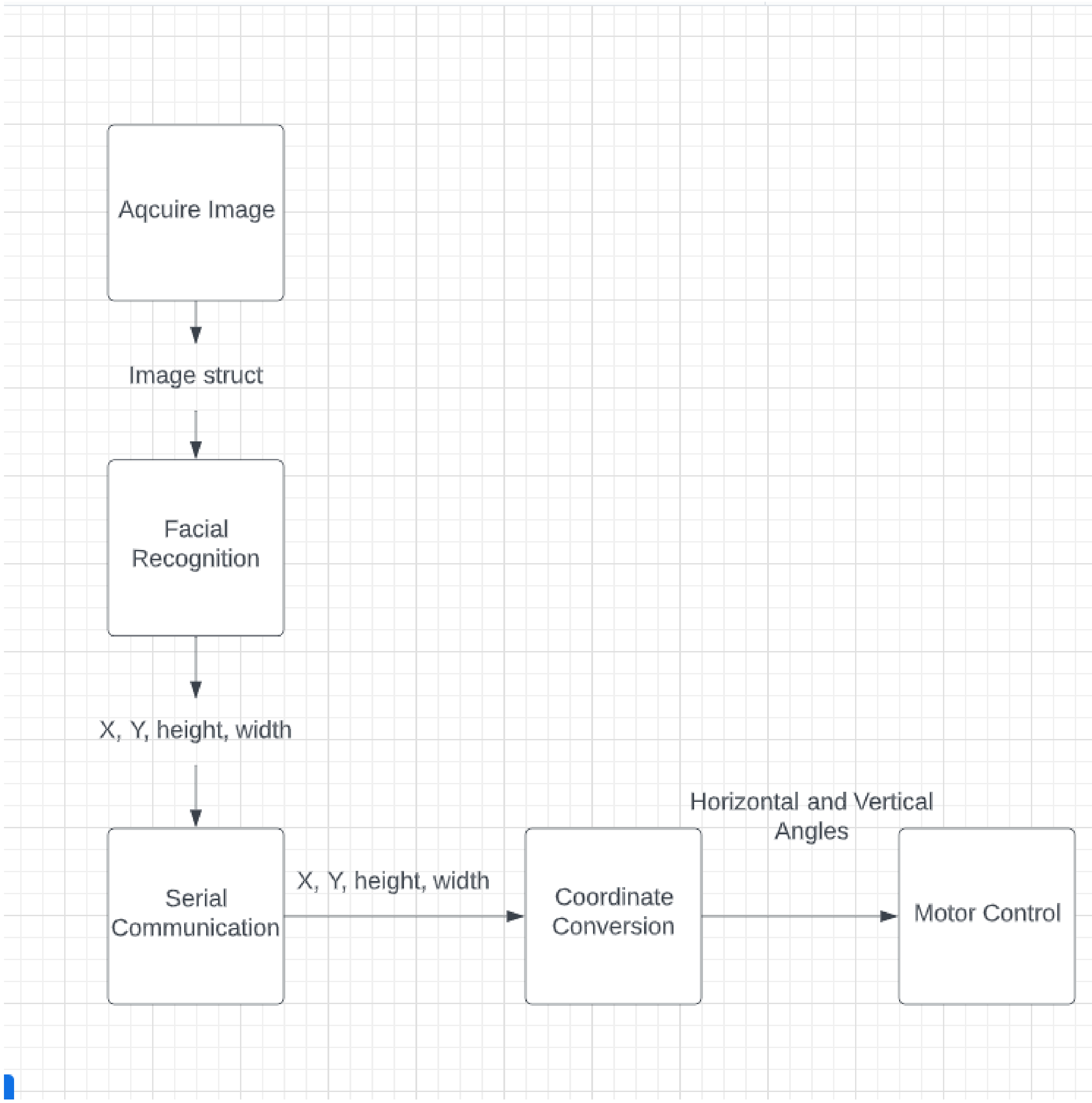

Figure 4 Software Interface

Figure 3 Interface block Diagram

5. Risk Assessment/mitigation

5.1 Functional and Performance Risks

Processing speed and detection accuracy are our main risks. If the system can not process the images fast enough or the accuracy is too low then it will result in the camera lagging behind the user and not following them correctly.

5.2 Programmatic Risk

Currently all of our hardware components have been decided on and we have remained under budget. The only issues related to schedule or budget that we could run into are major parts breaking such as the raspberry pi.

5.3 Safety and Reliability

Minor health and safety risks are possible due to working with electricity. This is not a very high risk since we will be working with relatively low voltage. The greater risk is any electrical damage to any equipment.

5.4 Knowledge Base and Uncertainty

Our team does not know the most effective way to maximize either our accuracy or the processing speed under our current design. We have an overall good understanding of how facial detection works at a basic level. However, where we lack knowledge currently is how to increase speed and accuracy.

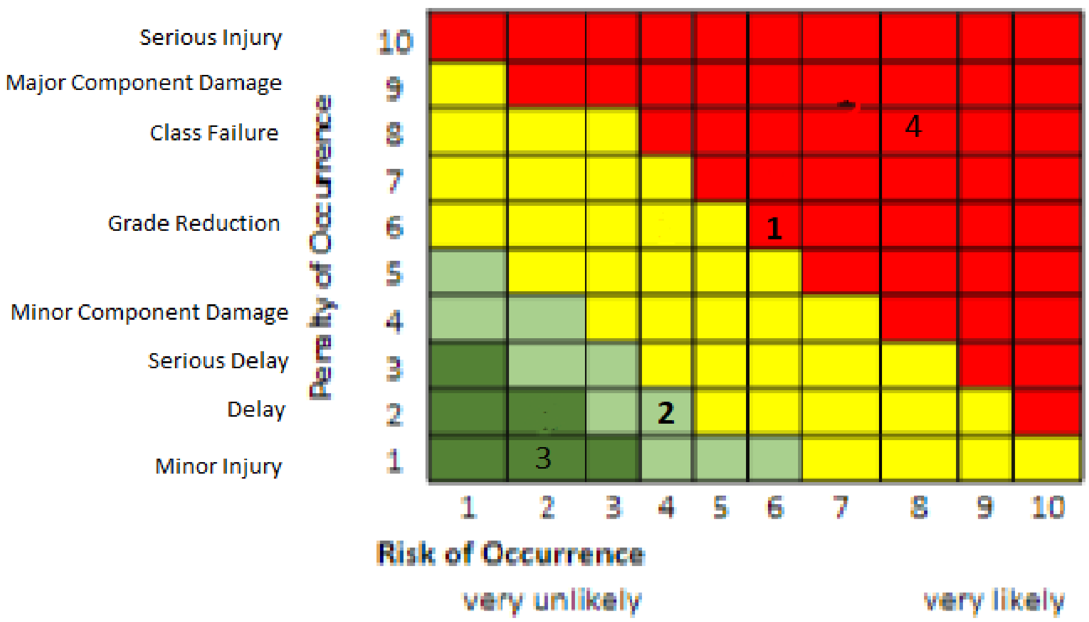

Figure 5 RAM Table

5.5 Risk Mitigation Steps

5.5.1

To mitigate our processing speed and accuracy issues we will be performing excursions to test the effectiveness of other algorithms and how the aspect ratio and FPS limiting effects the accuracy and speed.

5.5.2

To alleviate our programmatic risks we will be treating our main components with care as well as preparing other backup components from our personal contacts and sources.

5.5.3

Since our project has such low safety risks due to using such low voltages we mainly just have to follow basic safety precautions to prevent any major or minor bodily harm.

5.5.4

To mitigate any of our knowledge based risks we will be performing excursions as is needed. Since our main roadblock so far has to due with processing speed and accuracy we will be focusing on that first. However, we will continue to perform excursions as gaps in our knowledge are identified.

6. Budget

no	Item	Unit Cost	Sub Total Cost
2	Microprocessor	$70	$30
2	Servo Motor	$20	$90
1	Breadboard	$5	$95
TBD	Wires	$10	$105
3-4	Power Source	$20	$125
1	Camera	$10	$135
1	Tripod	$7	$145

Figure 6 Budget

7. Master Schedule

Phase	Feb Week 1	Feb Week 2	Feb Week 3	Feb Week 4	March Week 1	March Week 2	March Week 3	March Week 4	April Week 1	April Week 2	April Week 3	April Week 4
Concept Analysis												
Requirements Analysis												

Design Analysis												
Milestones	Concept Pres.					SRR	Excursion 1			Excursion 2	PDR	

Figure 7. Master schedule