**PROJECT II CAMERA SUBSYSTEM & MOCK DESIGN**

Team 3

Joseph Page, Joshua Silva, Natalia Jenkins, Garrison Veith

02/28/2025

# **Executive Summary**

In this project we work with implementing image processing as a tool to be used in future project applications. This involves the use of a camera as a sensor. We also brainstorm and showcase crucial design concepts to be used going forward using a complex decision matrix. We make use of customer requirements to come up with a list of engineering requirements to be fulfilled through various aspects of testing. We work on developing a website to showcase current and future projects. This project begins to set our eyes on the scope of our overall prototype and the extensive testing and design process that will be involved to ensure an efficient and accurate final product.

**Engineering Requirements for Camera Subsystem**

| Customer Requirements | Engineering Requirements |
| --- | --- |
| Golf Balls 0.5" Diameter | Camera must be able to identify the ball seeing only 0.2 sq. in. |
| Must be able to view the entire board. Camera should be placed where it captures as much of the board as it can |
| range of 1-2 ft target distance | Our camera in pair with a program should be able to identify distances from ball to hole and communicate this to the overall system. allowing for increased precision in hitting |
| 3 Possible holes | Camera must be able to identify each of the three holes and keep track of each holes location |
| Choose hole from interface | This adds to the previously mentioned requirement. User can now call on a specific hole and program can now make trajectory calculations using this hole |
| Interface should show user the identified holes in a graphic |
| efficient | Camera should have high detection rate given the board is well held in location |
| user friendly | program interface for start button and selection of holes should be easy for the user to understand |
| must use camera | should use a good enough camera to detect the balls and holes and be able to decipher between the two |

Table 1: Camera Subsystem Customer and Engineering Requirements

Table 1 gives an overall summary of requirements and generally what we plan to do in terms of implementing these requirements. A few of these we have already achieved but several others will require further development to ensure the requirement is met. This project specifically worked on developing the camera subsystem as a detection method. We most importantly were able to almost guarantee accurate detection given uniform lighting conditions. This will be very important for overall system performance.

**Document Hardware**

Hardware used involved the use of the Brio 100 Camera as our sensor. Electrically all else we needed was a simple computer with USB-C connection to create a link with our MATLAB program for image processing. We also needed to build a simple frame to hold the camera steady for the implementation of background subtraction. This project involved dealing with a number of physical factors like lighting and noise in our imaging. We also went through a few different detection techniques before landing on our final technique, that being Background Subtraction and color differentiation.

The Brio 100 camera has a maximum resolution of 1080 x 720. For our design we wanted to use as much of this total resolution as possible to capture the board. This involved a bit of playing around with the camera to find what height would allow us to capture as much of the overall board as we could. The height we found to use as much of the camera's resolution was approximately 2 feet 9 inches above the board's surface. This would allow the board to be taking up the full length of the camera. Following some cropping of our output photo we found the resolution to be about 612 x 306. This is due to the way MATLAB deals with cropping of photos. We cropped the photo to reduce any noise that could appear outside of the board to have the system be more accurate with detection.

Given the resolution of the cropped image and the size of the board that takes up the entire image we can simply deduce that the camera has resolution of 650 pixels/square inch. Using this and some other simple measurements we can find the pixels/ball to be roughly 127 while the holes have around 390 pixels. Through various trial and error using size detection we found this to be inaccurate. The fabric on the board right around the holes causes the pixels per hole to be significantly smaller. Aligning more similarly to the size of the balls. This caused a change in approach regarding image processing and differentiation between balls and holes.

We also made a relatively primitive camera mount shown in Figure 1. This mount needed to serve as an initial reliable design for holding our camera in a position where it can reliably detect the board.This design was intended to hold the camera at roughly 2’ 9” above the board and have the camera lens be centered at the middle of the board to capture the board the best. While it will likely undergo change in the future it showed us that with proper positioning, our camera can almost always detect the balls and holes.



Figure 1: Initial Camera Mount Design

**Document Software**

With the subsystem’s hardware & frame being established, the software can be effectively implemented. The software used for the camera subsystem uses image processing techniques to find locations of the golf balls and golf holes while removing outside noise that could make the subsystem inaccurate in its calculations. Most of these techniques are given from example code of similar problems, with thresholds being adjusted in order to suit the project’s specifications.

To find the locations of the golf holes & golf balls, it is necessary to use various image processing techniques to isolate the centroids and eliminate noise in the Design Lab setting. Although various image processing are viable, it was decided to use explored techniques such as Cropping, Background Subtraction, Binary Conversion, Erosion, and Dilation to implement the subsystem. The techniques are implemented in that order, as seen below in Figure 2.

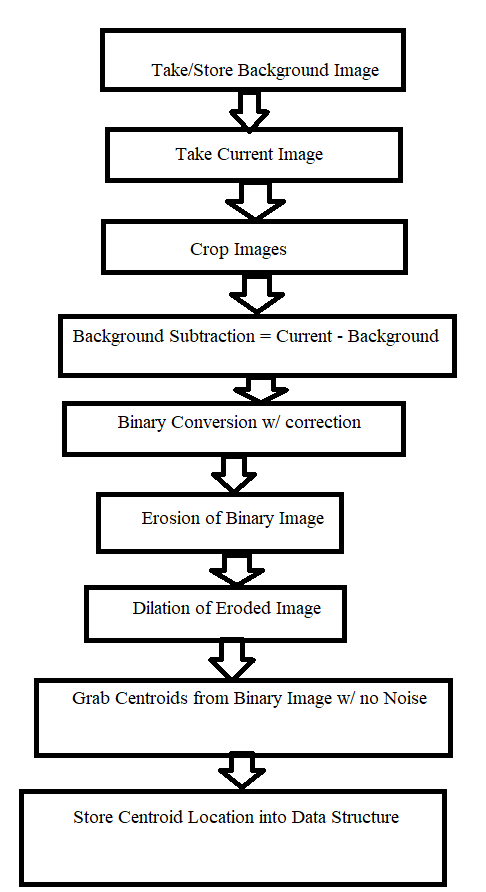


Figure 2: Flowchart of Software Implementation

To give further detail, Background subtraction is done to find the differences between one image & another. In this case, a golf board with no golf holes & balls compared to a random placement of golf balls and holes. Implementing this in software is done by taking or storing a background image, depending on user input, and then taking a picture of the “Current” or random assortment of golf balls & holes. Cropping is done on both images to remove some of the noise outside of the board, with the software saving a rectangle that captures just the board’s area.

Binary Conversion is done on the Background Subtraction before the other techniques in order to more accurately display data. Normally, this is one line in MATLAB, however with our subsystem relying on differing colors between the golf balls & holes in order to distinguish between the two, correction on the Background Subtraction must be done in order for the binary image to pick up less extreme colors. This is done by a simple loop that goes through every RGB pixel of the Background Subtraction to determine if it is light enough, far enough from the black background, and converts it to a light green. From there, binary conversion is able to pick up the differences between the two images.

With the binary conversion done, regionprops can be called already to find centroids, however there is still a noticeable chance of noise found through experimentation. In order to remove the noise, erosion on the binary image is done, to remove any small particles or small board movement that was picked up. Now however, the eroded image might have removed too much of the golf balls & holes to find a reliable centroid, in some cases it removes enough to detect several centroids. To ensure reliable calculations, dilation is done on the eroded image in order to restore/enhance the golf balls & holes.

From there, regionprops, a MATLAB function, is called on the binary image with noise removed. Regionprops, for the project’s purposes, will calculate the centroids of the objects, however it cannot distinguish between golf balls & holes. As stated earlier, the differences between the two are determined by color, with golf balls being white and a surface underneath the golf holes being any other color but white/black. To determine this, a loop going through each centroid is made, going to each centroid coordinate on the current image, and determining if the color is white for golf balls or any other color for golf holes. Various testing was done to determine the correct threshold, RGB [0 0 0] is black while [255 255 255] is white, which was found to be a RGB value that accounted for values slightly lower than white to pick up the golf balls with room lighting.

Lastly, the software stores the data from regionprops, in particular centroids, into a data structure called “golf.” Based on earlier color differentiation, centroids are assigned to a “ball” and “hole” array, in order to prepare for the Final Design’s requirements to hit a ball into a hole. The data structure also stores the previous images taken, including the binary image with no noise. And for the user’s convenience, the locations of the golf balls and holes are displayed onto the current image, as seen below in Figure 3.

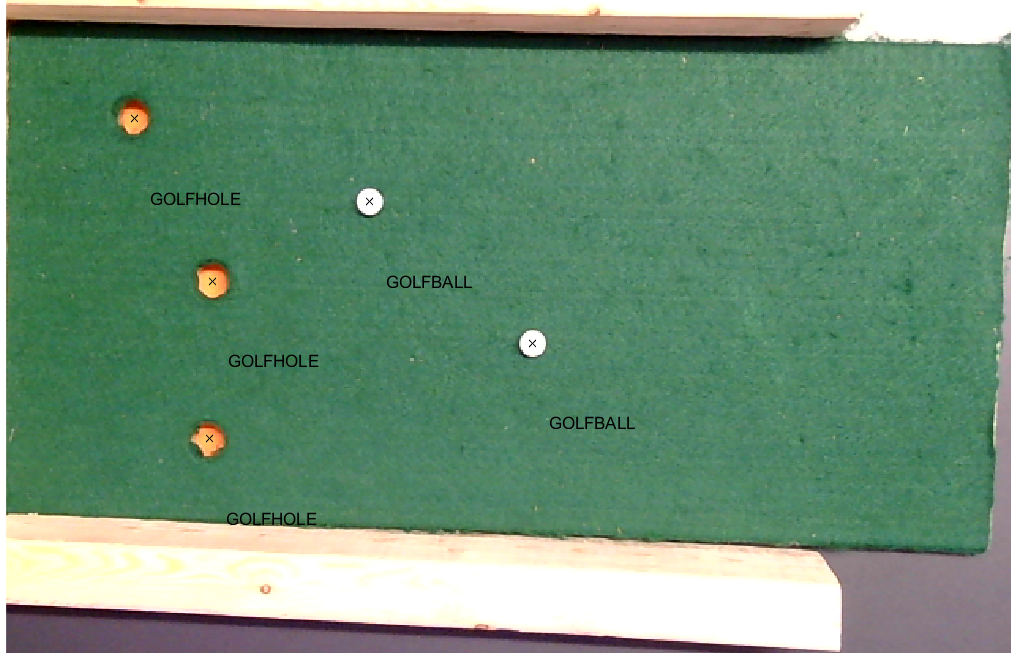


Figure 3: Image plotting the locations of Golf balls and Golf holes

**Engineering Requirements for the Entire System**

| Customer Requirements | Engineering Requirements | Tests to be conducted |
| --- | --- | --- |
| Sink a 0.5” diameter “golf ball” into a hole at a distance  ranging from 1 foot to 2 feet | Camera subsystem must be able to identify golf ball and hole location to a 98% degree of accuracy and store information for use in later calculations | This has been tested using the camera subsystem we have already developed. Centroids and locations of holes and ball locations have been perfectly identified during 10 tests |
| Device must be able to deliver enough force to hit a ball to these distances 95% of the time | Conduct system ball striking tests and measure distance to ensure system meets requirement' |
| 3 Possible holes | System must be able to correctly distinguish between three holes for use in calculations of positioning system with 95% accuracy | This will involve extensive testing of our prototype and overall system tuning |
| Choose hole from interface | User should have an easy GUI interface to be able to choose which hole they want to sink the ball in | We will implement a program interface and verify functionality and accuracy |
| Implement system start button | System start button should be able to be pressed once the user has chosen the hole | Implement another interface to allow user to start process and will conduct system tests to ensure button is implemented correctly |
| Inform user when the task is completed | Receive a notification of completion and run-time information pop-up once striking motion has been completed | Conduct system tests to ensure notification is implemented correctly |
| system should be within 2.5x1 feet | Overall system('putter') should fit within 2.5x1 foot area | After deciding on a final design we will need to come up with more technical drawings of the space our design is likely to take up |
| Can enter up to 2 in into work space | System should enter into work space no more than 2" | Using measuring tape during system tests to ensure system does not enter further than 2 in |
| Quiet | System should not exceed 70 dB | Perform noise tests using audio software (Audacity) |
| Efficient | Max power should not exceed 15W | Measure power with multimeter during system tests |
| Quick Response | System should move to desired location and strike the ball within 2 minutes | Use timer to time overall system tests |
| Safe/User friendly | Motors, gears and anything else similar should be covered or out of easy reach | This will be ensured with the use of coverings that will hide away any of these elements |
| interfaces should be easily understood | Will make the interface clear and unambiguous |
| Closed loop control of at least one axis | Use PID based closed loop control for accurate motor positioning within 3mm | Measure position during several tests to ensure accuracy |
| Use DC motor, Encoder, Camera | Must implement DC motor, Encoder, Camera | This will be achieved simply by the use of these elements in our design |

Table 2: Overall System Customer and Engineering Requirements and Tests To Be Conducted

**Design Solutions**

As a group, everyone came up with a solution on their own and we had a brainstorming session to discuss which solution we should go with. Though there are four of us in the group, two of the designs were very similar so we grouped them together and we were left with three great designs. Ultimately we decided on a hybrid of two of the solutions, which will be talked about later, for now, a discussion about the three solutions. To discuss them more efficiently we gave the designs nicknames: pool cue, hanging paddle, and golf club.

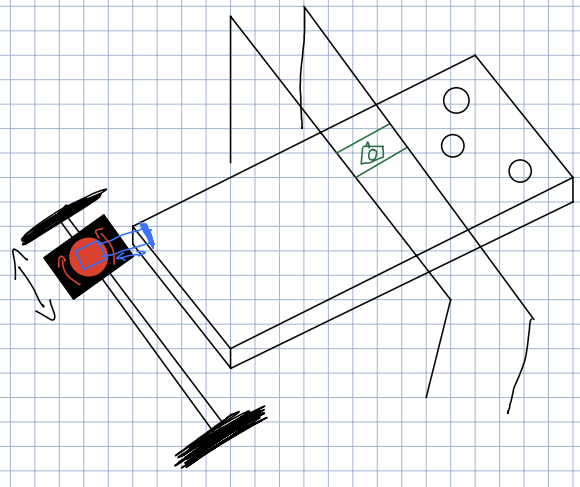


Figure 4: Sketch of “Pool Cue” solution

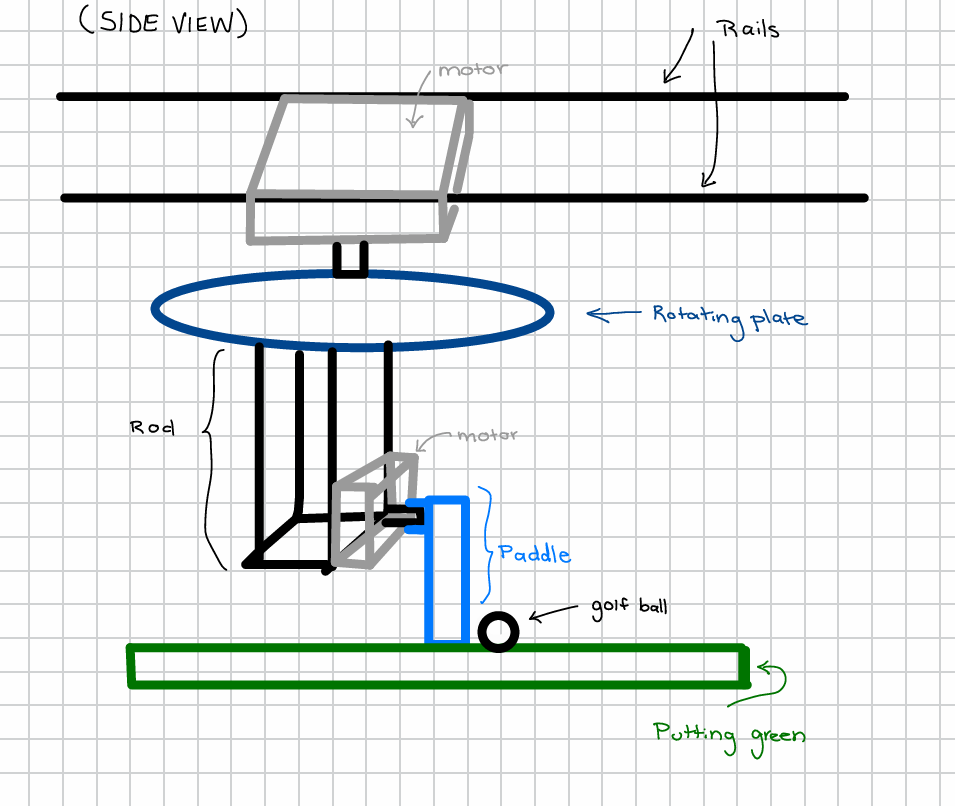


Figure 5: Sketch of “Hanging Paddle” solution

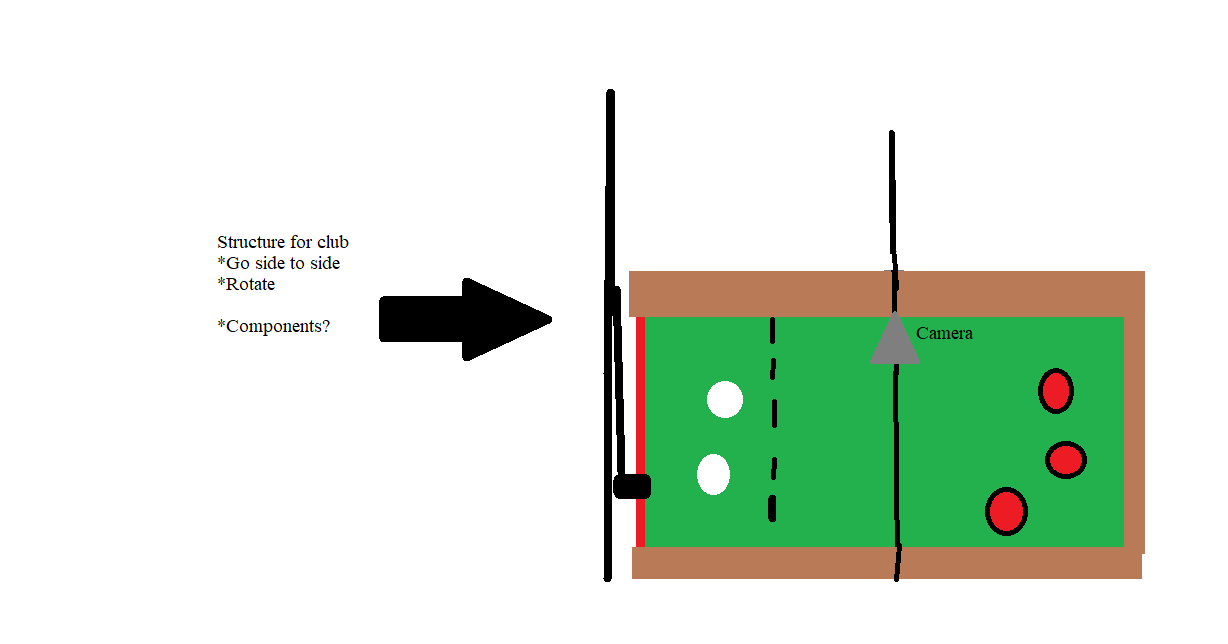


Figure 6: Sketch of “Golf Club” solution

Starting with the pool cue, this design features a solenoid as the instrument to hit the ball. The sketch that was made is in Figure 4. This design uses a rail to hold a carriage that moves the width of the board to align itself with the ball. The carriage uses a DC motor that has a wheel to move itself. This would have to go past the edges of the board in order to ensure that the ball can be hit into any hole, even in the most extreme cases. On the carriage there is a turntable that rotates the solenoid to align it behind the ball and in line with the hole that is chosen. This design is very small compared to the other designs because of the solenoid. It is also low to the ground which means it has a low center of gravity which helps lower the complexity of the design. It uses a solenoid to hit the ball which would require the solenoid to have a controllable speed and long enough stroke in order to hit the ball predictably and repeatably, this adds cost and potentially complexity. Another potential issue of this design is the wheel to move the carriage back and forth. This wheel might not be consistent because of slipping, in order to mitigate this, more effort would have to be put into the code.

Moving on to the hanging paddle design, this design uses a stepper or servo motor with a paddle attached to it to hit the ball into the hole. The sketch of this design is in Figure 5. Similar to the pool cue design, it uses a rail and carriage to move across the base of the board and to align itself with the ball. The main difference between the carriage in the pool cue and this design is that this design hangs above the board. This carriage is also driven by a belt from a motor fixed to the structure, not on the carriage. From the hanging carriage it has an arm to extend the paddle and the motor running the paddle to hover above the ball. From this there is a motor that turns the paddle and paddle motor to align it behind the ball and in line with the chosen hole. This design is larger than the pool cue since it hangs, this also means the design is more complex since the center of gravity is high. Since a stepper or servo motor is controlling the means of hitting the ball there is a high level of control over the speed, torque, and position of the motor through the swing.

Finishing up with the golf club design, this design uses a stepper or servo motor with a hanging golf club to swing and hit the ball into the hole. The sketch of this design is in Figure 6**.** This design is similar to the hanging paddle, because the system is hanging above the board. It has a rail that the golf club slides along to align it behind the ball. This is controlled by a DC motor. There is then a motor to pull the club back and swing the club to hit the ball into the hole. This design keeps the spirit of golf with the golf club as the instrument for hitting the ball, but this is only aesthetic. The major flaw with this design is the inability to rotate the club if the ball needs to be hit at an angle other than perpendicular to the board. This could be worked around though with the addition of a motor to rotate the club. Like the hanging paddle, this has a high center of gravity which adds complexity to the implementation of the design. Another potential downside to this design is depending on how heavy the club is, a high torque motor would be needed to control it. This design also has safety concerns because of the size of the club and worry that it might hit something behind it.

**Concept Evaluations**

|  |  | Option 1 | Option 2 | Option 3 | Option 4 |
| --- | --- | --- | --- | --- | --- |
| Criteria | Weighting Factor | Hanging Paddle | Pool Cue | Golf Club | Hanging Paddle and Pool Cue |
| Safety | 5 | 4 | 5 | 3 | 5 |
| Cost (low=5) | 5 | 2 | 2 | 3 | 2 |
| Size | 2 | 3 | 5 | 2 | 5 |
| Speed/Position Control | 4 | 5 | 2 | 4 | 5 |
| User Friendly | 3 | 5 | 3 | 2 | 5 |
| Ease of Implementation | 2 | 4 | 4 | 3 | 3 |
| Noise | 2 | 4 | 3 | 3 | 3 |
| Score | | 93 | 86 | 72 | 102 |

Table 3: Complex Decision Matrix of Our Initial Designs

As shown in Table 3, the criteria in the concept evaluation matrix are safety, cost, size, speed/position control, weight distribution, user-friendliness, ease of implementation, and noise. In addition, each criterion has a weighting factor. Safety has a weighting factor of five, as the safety of those around the system is imperative. The cost is also key, so it has a weighting factor of five. The size has a weighting factor of two because the main concern is building an effective system, so even if the system is large, the functionality is the focus. The speed/position control has a weighting factor of four; the better the position and speed control, the more likely the ball can accurately fall into the correct hole, which is key. User-friendliness is also relevant to the project and mentioned in the customer requirements. User-friendliness does not directly contribute to the functionality of the system, so it has a weighting factor of three. The ease of implementation criteria has a weighting factor of three, as the team is willing to design a more technically challenging system as long as it is effective. Ideally though, the system would be as simple as possible. The noise is mentioned in the customer requirements but does not directly affect the functionality of the system, so it has a weighting factor of three.

Initially, the team only had three ideas: the hanging paddle, the pool cue, and the golf club. After a discussion, it was decided that combining the hanging paddle and the pool cue would be best. The concept evaluation matrix reflects this - the score for the combination of the hanging paddle and the pool cue is the highest. The final mockup demonstrates the proposed concept as seen in Figure 7. The system has three ranges of motion: the striking system moves side-to-side along a linear rail, it can move forwards and backward, and can fully rotate in order to be in line with the ball and hole. This system is robust. The system will use several motors, which are expected to perform consistently over time, especially when applying controls techniques. Many other components such as rails, the paddle, and the arm will likely be 3D printed and are robust.

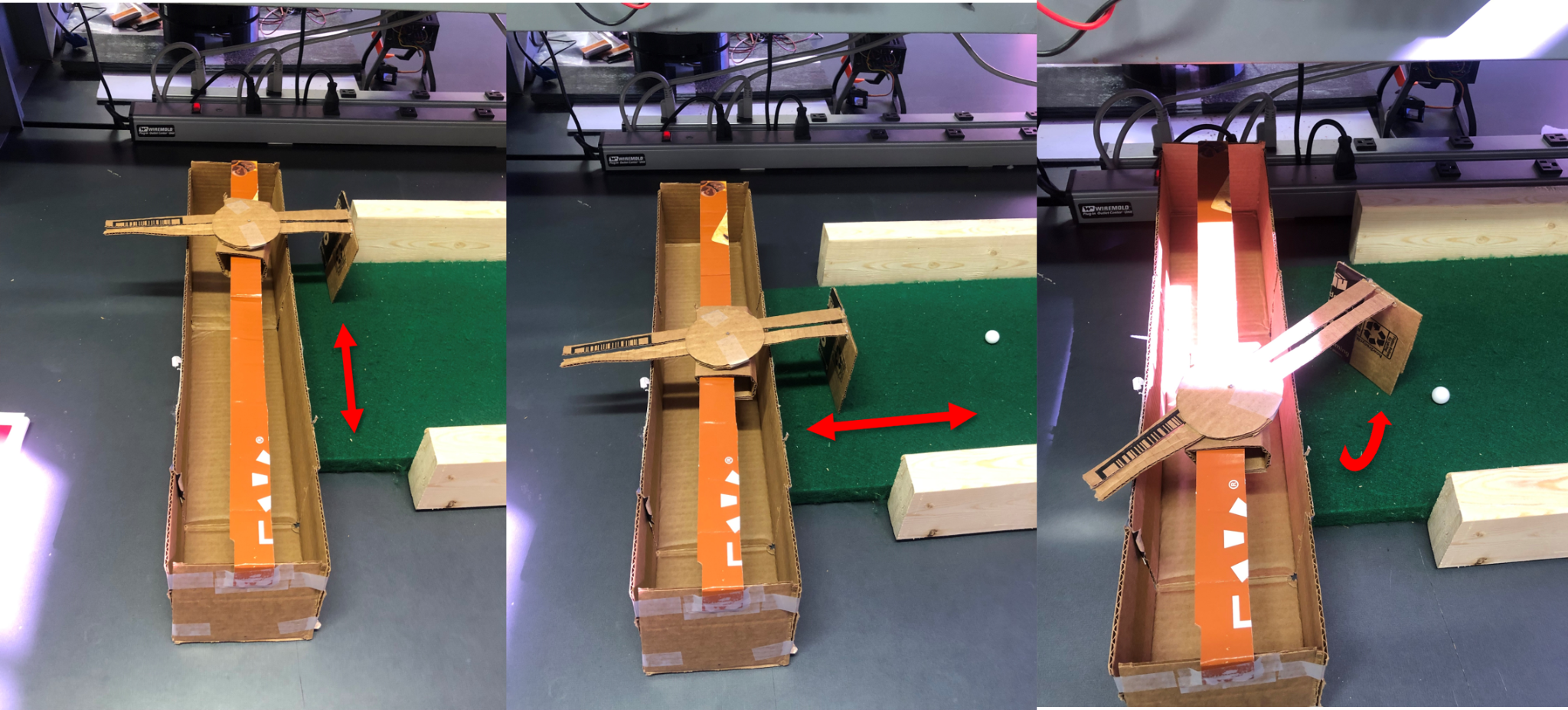


Figure 7: Ball-Striking Mockup

ECE 4950 Project 2 – Sensor and Website Rubric

Group Number (and Name): Group 3 (Joseph Page, Joshua Silva, Natalia Jenkins, Garrison Veith)

| Score | Pts |  |  |
| --- | --- | --- | --- |
|  | 5 | General Format - Professional Looking Document/Preparation (whole document)  a) Fonts, margins (11pt, times new roman, single spaced. 1" margins on all sides). b) Spelling and grammar are correct   1. Layout of pictures – all figures need numbers and captions and must be referenced in the text 2. Follows the page limitations below. 3. References. Use IEEE reference format. 4. This grading sheet is included as the final page. | O3-SA1:1 |
|  | 5 | Page 1: Title, Group Name, Group Members, and Date  Executive Summary (1 concise, well-written paragraph)  Provide an overview of this project. Briefly describe what you did and what you learned. | O3-SA1:1 |
|  | 20                20          20 | Subsystem Design  Page 2: Engineering Requirements for the Camera subsystem (~1 page)  In the context of just the Camera-as-a-Sensor, make a two column table that contains a column for the Customer Requirements (what are the functions of the sensing system?) and the resulting Engineering Requirements. Each row should contain a specific customer requirement and the resulting engineering requirement. One customer requirement may generate multiple engineering requirements. For example, the customer will want an “accurate” system, the Engineering Requirement could be 99.5% detection success.    Pages 2-3: Document Hardware (1 page)  Describe and show images of the equipment used, connection diagrams, calculation of resolution – pixels per square inch/cm on game board etc. Is the camera an appropriate sensor?    Pages 3-4: Document Software (2 pages)  Using Flowcharts, state diagrams, data structures etc. describe how the software is implemented. There is no need to include the source code. | O1-SA5:2                O2-SA2:1          O2-SA2:1 |
|  | 30 | Overall System Design  Page 5: Engineering Requirements for the Entire System (~1 page)  Make a three-column table that lists the Customer Requirement in the first column and the resulting Engineering Requirement(s) in the second column. Note that a customer requirement might branch to multiple engineering requirements. The third column should list the test that will be done on the prototype to verify that the design chosen meets each requirement.    Pages 6-7: Design Solutions (~2 pages)  Use a brain storming session to generate concepts. Document your top three most feasible ideas with sketches and brief descriptions while explaining the main features of each concept.    Page 8: Concept Evaluation (~1 page)  Use a complex (weighted) Concept Evaluation Matrix to show how the final design was chosen from the three best ideas described previously. Include a description of the weighting factors. Make sure to use at least six of your most important criteria in the matrix and be sure to provide and include the weighting factor (i.e. importance) for each criterion. | O1-SA5:2              O2-SA1:4          O2-SA1:4 |
|  |  | Describe your final design choice (~1 paragraph)  The robot mock up will be graded based on the following criteria:   1. Does it demonstrate the proposed concept in sufficient detail? 2. Is it well thought out? 3. Does it look like the final construction using this concept will be robust? |  |
|  |  | Page 9: Grading Sheet |  |
|  | 20 | General Website Format – provide link here: [ECE-4950 Group 3 Website](https://sites.google.com/g.clemson.edu/senior-design-1-group-3/home)  Is the website:   1. Aesthetically clean 2. Complete. Does it include:    1. The Team Description?    2. Reports 1 and 2?    3. Outline of future sections?    4. Proper use of graphics? 3. Follow accessibility compliance at: https://siteimprove.com/en-us/accessibility/adacompliance-website/ | O3-SA1:1 |
|  | 10 | Demo Slideshow Presentation (~3 slides)   1. What did you do to complete this project? 2. How did all the team members get involved? 3. How do you believe your team dynamic is? |  |