Home Depot Connector Type Identification

ECE4012 Senior Design Project

Team Name: Sensor Team Six
Website:
Project Advisor: Dr. Matthieu Bloch
Project Sponsor: Keith Ruckstuhl

Team Members
Joseph Doughty, jdoughty3@gatech.edu
Christopher Fox, chrisfox@gatech.edu
Matt McBrien, mmcbrien3@gatech.edu
Nicholas Korzik, nkorzik3@gatech.edu
Lindsey Robirds, lindseyro@gatech.edu

Submitted

December 11, 2018

Table of Contents

Executive Summary

Sensor Team Six was tasked with creating a device that can aid Home Depot customers in identifying connectors such as screws and bolts that are brought into the store. Customers will place a connector inside the device and the best match will be presented on the display. By providing customers with a fast, reliable system for identifying connectors, customers can move more quickly through the store and eliminate the need for an employee to help with such a need. A previous team had created a proof of concept prototype and the current team was tasked with improving upon that design.

Sensor Team Six's prototype uses a Raspberry Pi computer, a camera module, and server-side computing to capture and analyze the images. The images are preprocessed to reduce total computation. After preprocessing, tailored image processing techniques are used to determine the head width, body width, total length, body length, head width location, and thread count of the connector supplied. Once the parameters are extracted, a random forest classifier is employed to determine the closest match. This match result produced by the algorithm is displayed to the customer on an LCD display. Performance goals that factor into the design process include: the device must be intuitive enough to be successfully used without instruction, accurate enough to build the customer's trust, and quick enough to hold the customer's attention. The component cost of creating the prototype design has been estimated at approximately $200.

Home Depot Connector Type Identification

1. Introduction

Sensor Team Six was tasked by Home Depot to design a connector identifier for in-store use by customers. Designing a prototype of this device cost approximately $200 in components.

1.1 Objective

Sensor Team Six extended the work of previous groups and laid the groundwork for future work to be done at Home Depot. In previous years, work was done to identify some key characteristics of screws and bolts (e.g. height, width, thread count) using image analysis. Prior work included building a model, capturing images with an Android phone, and processing images to determine properties of the screw. The Fall 2017 team was able to produce a proof of concept but the vision system implemented did not produce the accuracy required to become a full-fledged in-store model. From their tests, they achieved a thread count measurement accuracy of 53.75% but were unable to identify the head type of the screw [1]. Sensor Team Six expanded on that work by utilizing more sophisticated computer vision techniques for parameter extraction and classifying screws using a random forest classifier. An ideal device prototype would be able to identify screws sold by Home Depot and be easily expandable to work on additional connector types such as bolts, nuts, and washers and give customers instructions on where to find these items in the aisle.

1.2 Motivation

A successful in-store implementation of this prototype will provide a more enjoyable shopping experience for customers and will increase efficiency of Home Depot employees. By using this product, customers would be able to quickly procure information about the connector along with any other information Home Depot wishes to supply the customer based on this query. Additionally, helping customers locate screws and bolts takes up employee time that could be spent on other job duties, so this product will eliminate the need for an employee to direct customers around the store.

1.3 Background

1.3.1 Image Capture

The device must capture images of screws and bolts given by the customer. The resolution and contrast of the image must be of sufficient quality to produce consistent results from the analysis part of the process. In previous years, an 8MP camera on an Android phone was used capture images [1]. Sensor Team Six chose to use a Raspberry Pi 3 Model B+ computer primarily because of its onboard Wi-Fi and Ethernet connectivity and its built-in camera support which enabled the team to use an 8MP Raspberry Pi Camera Module.

1.3.2 Image Analysis

To determine the type of item supplied by the customer, the product must analyze the characteristics of the item from the image captured. Image analysis, and more generally computer vision, is a large field of study that includes both digital signal processing (DSP) techniques and machine learning algorithms. DSP generally involves applying mathematical transformations of images and using the new representation to more easily characterize the image and extract data. The previous team's analysis started by resizing the image to focus on the screw, followed by a Hough Line Transform to find the longest part of the screw for rotating the image and the height calculation [2]. Their techniques were not successful enough to reliably identify the screw for the customer. Machine learning is a growing field of research that has found success in image analysis. The vision side of machine learning heavily relies on neural networks or other deep learning techniques. These networks usually include millions or hundreds of millions of layers and nodes. To train a network of this size, an inordinate amount of data is required. The time to acquire the data and train the network made this approach unfeasible for a project of this scope. A University of Toronto team that researched deep convolutional networks took "five to six days" to retrain their network with fresh data [3]. Additionally, machine learning's vision techniques are typically used when there are a wide variety of images and large variation in the images. The University of Toronto team used their system to analyze 1.2 million images in 1000 different classes. For this project, machine learning's main benefit would be to allow for less preprocessing.

1.3.3 Classification

Once the parameters of the screw in an image are determined, this data must be used to identify the screw. Since the prototype allows for the screw to be placed within the stage at any orientation, the parameter extraction will produce results that vary from image to image as not all perturbations can be accounted for in the image analysis. To handle these variations, a method for classification must be employed. Statistical classifiers that have been trained on an identified set of data is a common technique for this task. By capturing images of the same screw in different orientations, a database of parameters and their corresponding screw can be created which can train the classifier. A random forest classifier is popular choice for identification as it avoids overfitting and can be tailored to produce low prediction times.

1.3.4 Data Transfer

To analyze the image, DSP algorithms must be applied to the image. These techniques are computationally intensive and may not run quickly enough for the customer on the small computer that is used to capture the images. To improve processing time, the image can be transferred to a more powerful computer with the results then returned to the device. The previous year's team performed all processing on the local device which may have hindered their processing time. Off-device processing can be achieved by using an external commercial cloud server or building an in-store server. The commercial server provides the hardware and networking to perform off-site computations, and companies such as Amazon and Google have businesses set up around this [4, 5]. Instead of a cloud server, an in-store Linux server can be built that handles the computation.

2. Project Description and Goals

Sensor Team Six researched and developed the hardware and software required to supply customers with near instant information about screws provided to the device. This included a physical device that includes a stage for items, a camera, a user interface, and a controlling computer. The software developed controls the user interface and camera. The analysis of the image may take place either on a server located in the store that can produce results quickly enough that the lag time associated with transferring the data will be negligible or on the local device's processor. The analysis will return a best guess about what item was provided and then provide this information to the customer via the user interface. The device's use and operation must be easily understandable by a new customer and produce accurate enough results to replace the intervention of a store employee. Additionally, the device is powered by a 5V AC wall adapter rated for up to 2 amps. Design aspects of the system include:

- Touch screen user interface and results display
- Real-time image capturing
- Device enclosure with bay for placing connector
- Connection to external processing system via Wi-Fi network

3. Technical Specifications

Enclosure

- Maximum dimensions 20” x 10” x 10”
- 3-D printed materials

Raspberry Pi Model B [6]

- Broadcom BCM2837 64bit ARMv7 1.2GHz Quad Core Processor
- 1GB RAM
- BCM43143 Wi-Fi on board
- 40pin extended GPIO
- CSI camera port for connecting the Raspberry Pi camera
- DSI display port for connecting the Raspberry Pi touch screen display
- Micro SD port for loading your operating system and storing data
- Upgraded switched Micro USB power source (now supports up to 2.4 Amps)

Raspberry Pi Camera Board v2 [7]

- 8 Megapixel Sony IMX219 for Raspberry Pi
- Image resolution: 3280 x 2464 pixels

7” LCD Touchscreen [8]

- 5-12 V Power supply required
- 1024 x 600 Resolution

External Battery

- Battery life should last for a 16 hour work day.

RB-Phi-203 100g Micro Load Cell [9]

- Max value of 100g before failure.
- Rated Output 600 microvolts/V
- Rated Output Error Max 150 microvolts/V

4. Verification

Enclosure

- The dimensions of the final prototype are 11.4” x 7.6” x 4.7”, which satisfies the size requirements put in place by The Home Depot which were 20” x 10” x 10”

Raspberry Pi 3 Model B+ [11]

- A Raspberry Pi 3 Model B+ was used instead of a Raspberry Pi 3 Model B, so all related requirements were met or surpassed.

Raspberry Pi Camera Board v2 [7]

- The selected camera module met all specifications and performed as required.

7” LCD Touchscreen [12]

- 5V operating voltage provided by GPIO pins on the Pi.
- The 800 x 480 resolution is slightly lower than the specification.

External Battery

- This design constraint was removed by Home Depot.

RB-Phi-203 100g Micro Load Cell [9]

- Due to complications explained below, the load cell was not used in the final implementation.
- The load cell reported accurate data outside of the enclosure, however, when attempting to attach the load cell to the weighing plate, the readings were inaccurate.

5 Design Approach and Details

5.1 <u>Design Approach</u>

Sensor Team Six chose to focus on improving the previous design implemented by the Fall 2017 ECE 4012 team, Screws and Volts, rather than overhauling the design component for this project. The fundamental configuration of the connector identification prototype is similar in size and form, however there were substantial changes to the hardware component and the software component. Figure 1 shown on the next page displays the relationship between the different hardware and software elements of the connector identifier.

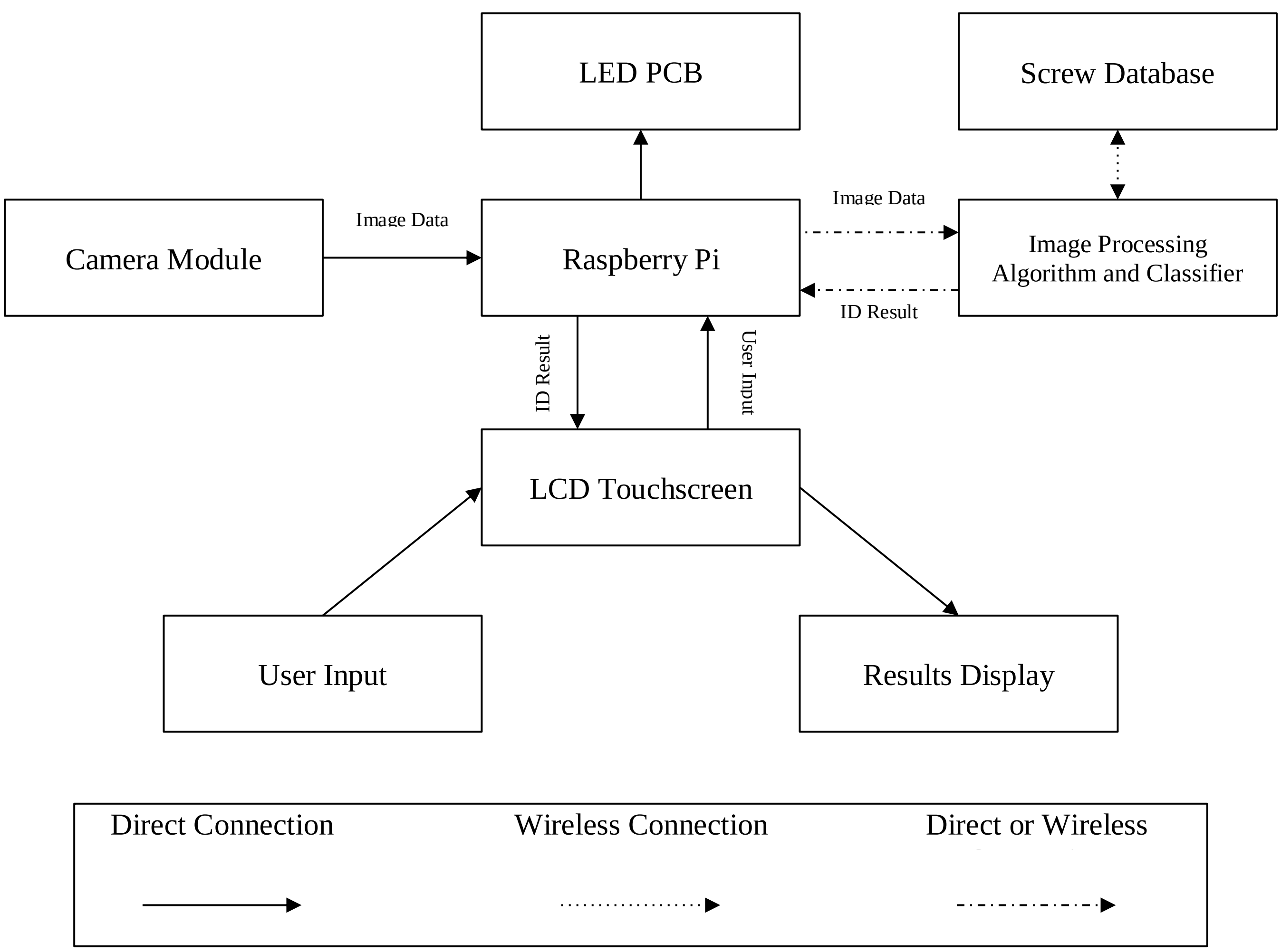


Figure 1. Block Diagram depicting system interdependence.

5.1.1 <u>Hardware Components</u>

The previous configuration used a mobile Android phone mounted on top of the frame. The Android phone took a picture of the connector type and used the cellular network connection to analyze the image. In the updated design a Raspberry Pi serves as the main hardware component rather than the mobile phone and utilizes multiple hardware peripherals. A camera module was attached to take the picture of the connector inside the frame, an ethernet cable facilitated communication between the Raspberry Pi and the processor, and an LCD screen was mounted to the front of the housing to provide a user interface to display the instructions and results. A rough sketch of the early device design is shown in Figure 2. As the Raspberry Pi is attached to the LCD screen on the inside of the box, it is not shown in the drawing.

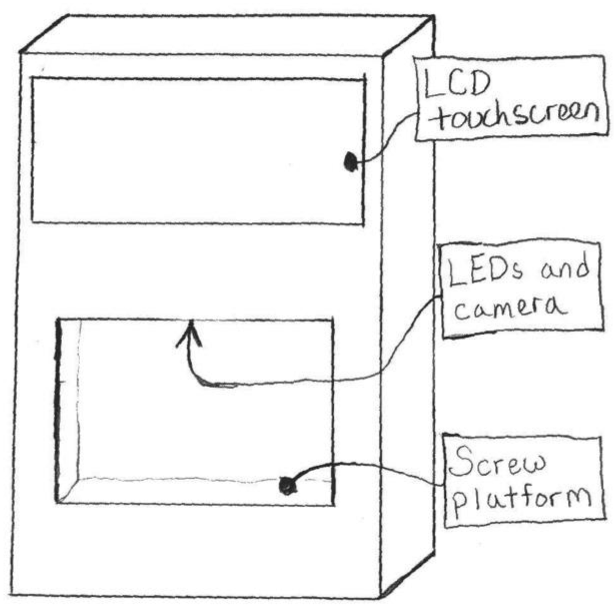


Figure 2. Hand drawn sketch of the device design.

5.1.2 Image Processing

Once the image of a connector is captured, it is analyzed by software to determine what type of connector the customer has supplied. The previous team used edge detection to calculate the height, width, thread count, and head type of the connector. This information was used to estimate what type of connector was present.

Sensor Team Six's algorithm performed image normalization to create uniform alignment and used cropping to reduce the image size. This preprocessing step reduces overall execution time by computing the second stage of DSP transformations on a smaller image. The ultimate goal of the image processing is to segment the object from the background, extract the outline of the object into a 1-dimension vector for faster processing, and estimate parameters from the outline. The first step is segment the object and align it vertically. Figure 3 shows an example of an original screw image (left) and the image after difference filtering and segmentation (right).

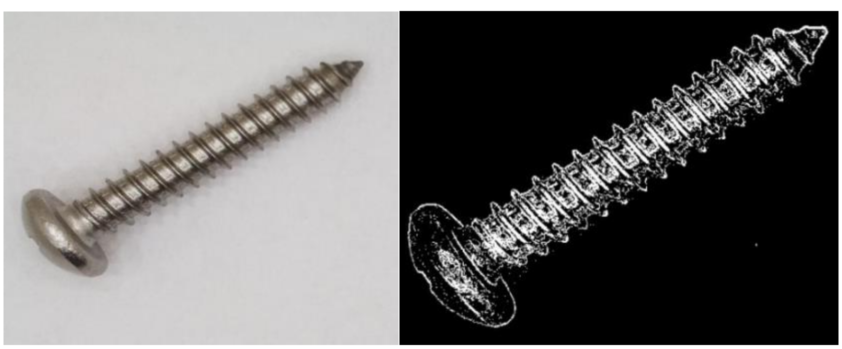


Figure 3. Original image of screw (left), image after difference filtering and segmentation (right).

The second stage of processing focuses on reducing spurious pixels that will negatively impact accurate parameter measurement. The primary source of these pixels are defects in the background such as dirt or smudges as well as shadows cast by the object. A projection mask is used to eliminate any stray pixels outside the largest cluster of pixels. Figure 4 illustrates the projection mask.

Any remaining points in the "Projection Shadow" are handled by applying two Morphological transforms. The first transform uses a 3x3 + shape to dilate the image, and the second transform uses a 3x2 rectangle to erode the image back to normal size. The end result of this procedure removes small clusters pixels but leaves large structures unchanged.

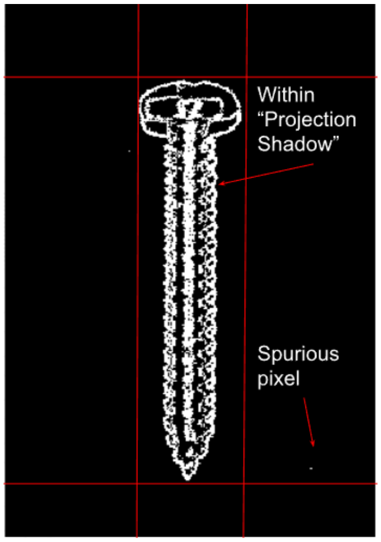


Figure 4. Projection mask applied to remove spurious pixels in background.

Once the preprocessing occurs, the parameter estimation algorithms are applied to the extracted outline. These techniques are used to acquire identifiable characteristics about the connector. First, difference filters are applied across rows and down columns to determine object boundaries. Then, the boundary data is treated as a one-dimensional signal. Filtering the boundary data in turn allowed for the axis of alignment for the screw to be found. Once the outline of the object is determined, the algorithm extracts six parameters used for classification: total length, body length, total width, body width, head width location, and thread count.

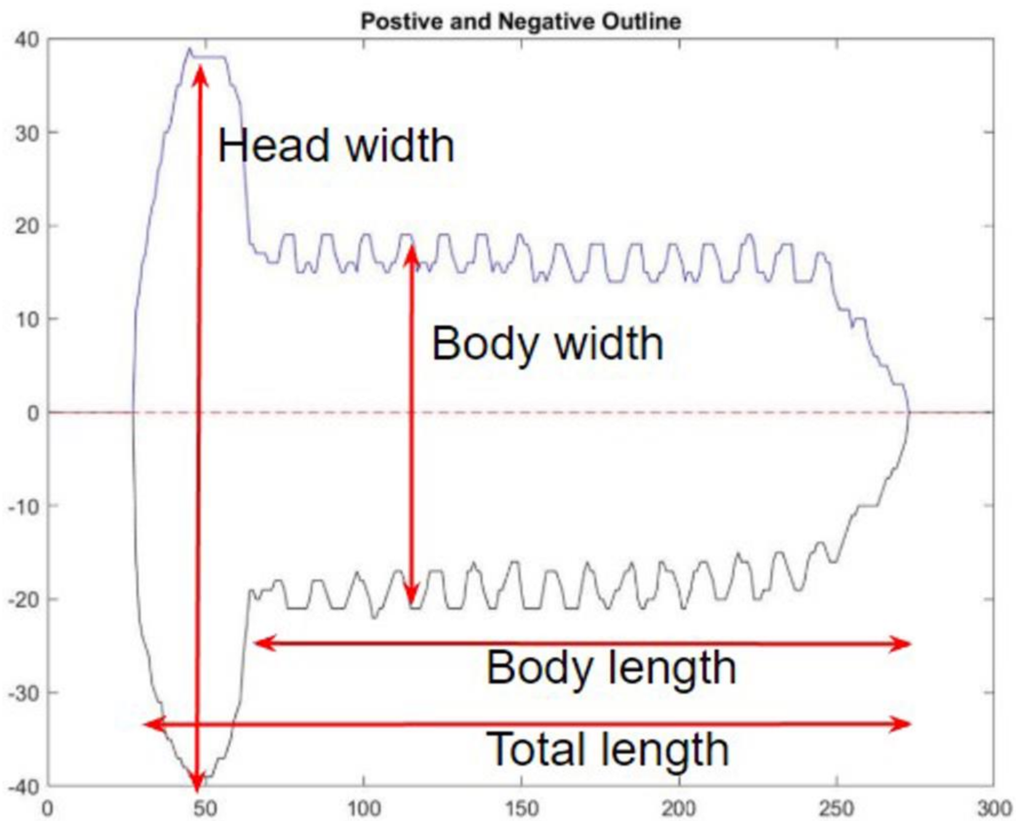


Figure 5. Parameters obtainable from the vector formed from the segmented screw image.

Once the parameters are extracted from the image, they are sent to a random forest classifier. Training data was produced by taking ten images of each screw from a set to be classified and extracting the parameters from each image. The classifier was then trained on this database, which included the parameters extracted and the correct associated screw. The random forest classifier used ten trees to make its prediction. A low number of trees keeps the prediction times low. Additionally, the classifier used the mean squared error to determine when to split. These parameters were determined by testing on early prototypes with smaller sets of screws. Since these early tests produced accurate results and low prediction times on the smaller sets, the parameters selected to be sent to the classifier were also used in this way for the final prototype. The result from the classifier is then presented to the user by the interface on the LCD touchscreen.

5.1.3 <u>Graphical User Interface</u>

A vital part of the success of this project was ensuring that the final product was simple to use for Home Depot customers. The physical interface is imperative to this, so the LCD screen used for the prototype was selected as it is large enough to display the instructions and results of the identification. The initial screen (Figure 6, left) displayed by the user interface shows a Home Depot logo as well as a button that says “Begin”.

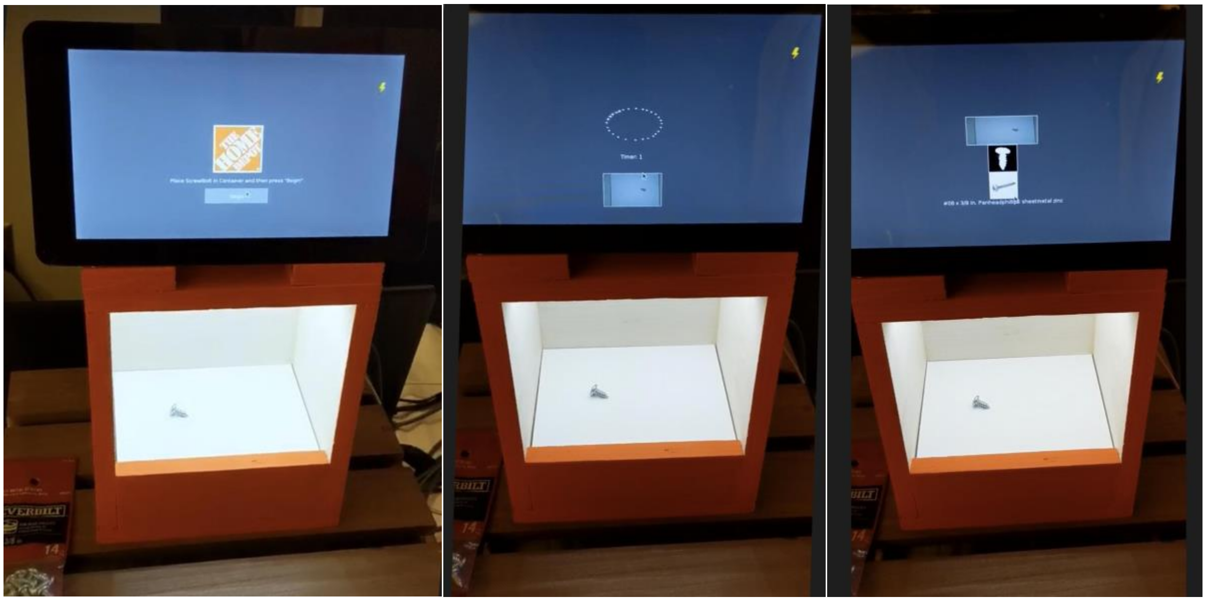


Figure 6. Prototype with initial screen (left), processing screen (center), and results screen (right).

The customer places their unknown part into the device and then presses the button on the touchscreen interface to initiate image capture and processing. While the image processing algorithm and classifier are executing, the touchscreen displays a loading icon as well as the picture that the camera module took of the screw placed in the box (Figure 6, center). Finally, the LCD screen displays the initial camera module picture, a segmented image of the screw, the identification result which includes the name and Home Depot stock images of the screw (Figure 6, right). Once the user is finished with the result display screen, they can click on a button that says “Restart” in the upper left of the result screen that returns the user to the initial screen. Integrating the prototype with store specific information would mean that the interface would eventually display the cost of the connector and its location in the store.

5.1.4 Device Enclosure

The enclosure that housed the hardware and software components measured 11.4” x 7.6” x 4.7” allowing it to be portable and compact. It was constructed out of one-half inch plywood and secured by nails. The previous prototype was 3D printed and while it was satisfactory for an initial prototype, the final enclosure was constructed out of wood so as to allow for more customization for attaching the hardware components such as the mounting hole for the camera module and a hole in the rear of the device to route the LED PCB leads.

An important aspect of the physical design was making the device appealing to use for the customer. To accomplish this the device enclosure was painted orange and embellished with The Home Depot Logo. These added design aspects were intended to increased appeal to users at the Georgia Tech Capstone Expo and make it clear to them what the device was intended for and who the company sponsor was.

The intended user for the final prototype is a Home Depot customer, so the design for everything that the customer sees when interacting with the prototype was simple and engaging. Constructing the enclosure out of painted wood provided the aesthetic of a hand-made project, which is something a Home Depot customer may appreciate. Based on feedback given from the many people who used the device during the Expo, the overall appearance and function of the final prototype was a large success.

5.2 Codes and Standards

Because this product is made for a specific problem occurring in Home Depot retail stores, the company sponsor has provided definite specifications on the features of the automated connector identifier. The entire component must be no longer than 10-20 inches and no wider than 4-6 inches. It must have an easily accessible flat surface inside the enclosure onto which the user can place a component. The sensors that are present on the platform must be non-obtrusive as well. A microcontroller or computer must take sensor input and provide an accurate identification of the component via a small LCD-type panel.

The Raspberry Pi module that controls the hardware of the project contains Wi-Fi capability that sends and receives the image processing data to the processor. The Wi-Fi capability standards are concealed by the Raspberry Pi module, but it is assumed that the module is compliant with IEEE 802.11i/WPA2 code.

5.3 Constraints, Alternatives, and Tradeoffs

One design tradeoff that required a decision was to choose between different power options. The first option powered the device directly from 120V AC wall outlets while the other option involved including a battery pack inside the system. The advantage of battery power is that the device would eliminate the need for installing new power lines throughout the store which is considerably costly. However, the disadvantage is that employees will need to maintain the machine by replacing the battery once it gets low on power.

Battery power was initially chosen as the more cost-effective option, but during the Fall 2018 semester, Sensor Team Six was informed by The Home Depot that power was no longer a constraint on the project. The cost of extending power lines to the prototype's location within the store was no longer prohibitive, so the project moved forward with more convenient wall power assuming 120V AC wall outlets would be readily available to power the final device.

The tradeoff between the number of LEDs used versus power consumption and complexity were apparent as shadows became an issue. The single LED consumed less power overall and is much simpler, however, it does not provide enough lighting to accurately classify images (Figure 7, left). The LED PCB consumes more power and is more complex to fabricate but provides enough lighting to substantially reduce shadows from the images and increase parameter extraction accuracy (Figure 7, right). To obtain the best results possible and to help secure the LEDs to the top of the enclosure in a neat and uniform way, the LED PCB was chosen.

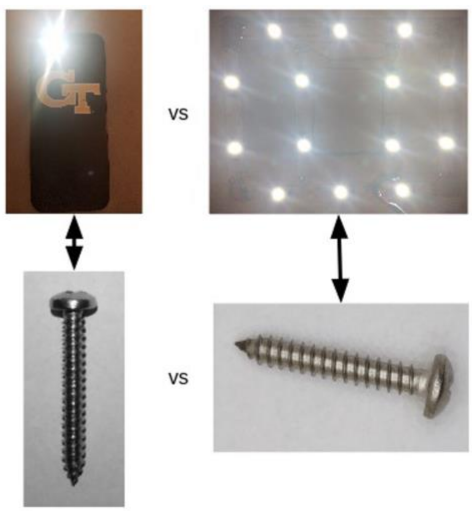


Figure 7. Single LED source and resulting image (left), 12-LED circuit and resulting image (right).

Another important tradeoff for the design is the price to performance comparison of a using an Android phone versus a Raspberry Pi module. An Android phone with a touch screen, Wi-Fi, processor, and camera would cost about $100 [2]. A Raspberry Pi 3 Model B+ (which has built-in Wi-Fi capability) with an 8GB SD card, 8MP camera, and a touch screen would cost approximately $162 [6, 7, 8, 10]. Using an Android phone in the design would be cheaper and less complex, but it would be difficult to interface a force transducer with the phone. The Raspberry Pi option allows for more flexibility in the design which is why this option was ultimately chosen.

6. Schedule, Tasks, and Milestones

Appendix A shows the Gantt Chart which includes the major tasks that were completed, the time allowed for each step, and the order in which they were completed. The CPM chart found in Appendix B and the PERT chart found in Appendix C show more detailed information about the flow of milestones. The PERT chart shows estimates for each task that include the best-case time, the most likely time, and the worst-case time. These estimates give a probable work time of 65.68 days and a 75.8% chance of finishing the work by the Design Expo. Between ordering parts and finishing our prototype, 83 days elapsed, and the final prototype was successfully finished before the Expo.

7. Project Demonstration

The prototype demonstration involved live image acquisition and display of results as well as a general characterization of the processing algorithm. The real-time demo verified that the user interface, camera module, data transfer, image processing, and classification all worked together. Additionally, manual tests on the image processing and classification were performed prior to the live demo to determine the accuracy of the device under ideal conditions. Given that the device is sufficiently modularized, each subsystem was iteratively tested and improved throughout the design process.

7.1 Classification Effectiveness

The design specifications dictate that the prototype classify screws and bolts as quickly and accurately as possible. To be able to demonstrate this has been achieved, three quantifiable attributes of the classifier have been determined - a robust system must successfully perform in each category: consistency, precision, and scope.

- Consistency refers to the ability to repetitively identify the same object regardless of perturbations such as orientation.
- Precision is determined by the capability to distinguish very similar but distinct objects.
- Scope is defined by the ability of the system classify substantially different screws and bolts.

To test on these categories, an assortment of 25 screws was selected to cover a wide range variation. Additionally, this collection had a few screws that were extremely similar. For example, the panhead #12 1-¼” and the oval head #12 1-¼” only differed in head type. This dataset, therefore, tested very different and very similar screws. Success was measured against a goal rate of 85%. During all testing, timing measurements were taken to quantify the processing speed. The previous team’s approach required at most 10 seconds of processing [2]. To demonstrate an increase in processing speed, Sensor Team Six’s goal is to achieve total processing time under a maximum of 5 seconds.

7.2 Manual Testing

Prior to the live demonstration, the accuracy of the overall system was determined by multiple methods. Once ten images of each screw were captured and their parameters extracted, the classifier was trained on this set of data. Using the holdout method with a train:test ratio of 80:20, the random forest classifier achieved 98% test accuracy.

The classifier was then tested on new data, which was done by placing each screw into the prototype in different orientations. The accuracy of the results as well as the total latency was recorded, and the tests were 93% accurate in classifying the screw correctly. The total time for execution varied from 1.8 to 8.4 seconds with an average of 3.9 seconds. The length of the execution was positively correlated with the length of the screw. When breaking down accuracy to the specific identifiers Home Depot uses, the performance metrics from the classifier are as follows:

- Total Length 96% accuracy
- Screw Width 100% accuracy
- Head Type 94% accuracy

These statistics are from the output of the classifier. The classifier has the possibility of producing the correct output even if one of parameters extracted is not correct as other parameters could identify the screw. The accuracy of the image analysis for the total length and the width were calculated using the database of 250 parameters. The error in length had a standard deviation of 0.081 inches and mean of 0.071 inches. The error in width had a standard deviation of 0.015 inches and a mean of 0.011 inches. The smallest difference between lengths in screws is 0.125 inches, and the smallest difference between widths in screws is 0.026 inches. With these base values, the image analysis would incorrectly identify the length and width of the screws more than 50% of the time with a standard distribution.

This is because there is a bias in the image analysis that augments the size of the screw (the average error of length and width are both positive). While the image analysis is not extremely accurate, the classifier performed well because it was able to take the bias into account when performing its classification. That means that the standard deviation of the image analysis becomes more important because the consistency of the results from image to image is more important than the accuracy of the results. It is important to note that the total length failures were exclusively the result of a single screw (#10 2.5" flathead zinc) where the head consistently was lost by the image segmentation module since the material closely resembled the background.

There was never a failure when determining the screw width. This was not the case in an earlier version of the prototype as a single shadow pixel could dramatically increase the estimated width. The change that improved this performance was to sample the screw width once per thread and take the median value of this vector. Finally, the screws that failed most consistently were those with a very similar screw in our database. For example, the #6 1" flathead and the #6 1" oval head only differ in their head type. In manual testing, these specific screws were classified correctly only 66.7% of the time. For full results from the manual testing, refer to Appendix D for the confusion matrix.

7.3 Live Demonstration

During the live end-to-end prototype demonstration, the following features on the physical device were demonstrated:

- The camera module capturing the image
- The LCD touch screen acting as the user interface and final results display
- The wired (ethernet) communication between the prototype capturing the image and the server processing the data and returning results

During the live demo, the prototype behaved as expected based on the manual tests performed. The screws were classified correctly at a rate similar to our manual testing. Just like our manual testing, the screws with similar counterparts failed the most consistently. However, most users independently chose to classify large, unique screws which performed best in classification.

Additionally, the interface was interacted with by dozens of users who were able to read the screen and use the device to retrieve results. A common complaint from this demo was the size of the text being too small. Twice during the demo, the interface crashed when the user pressed the "Begin" button. The cause of this failure is currently unknown but could be related to the power limitations of the Raspberry Pi, which was constantly warning about low voltage during the demo.

8. Marketing and Cost Analysis

8.1 Marketing Analysis

Assuming a Home Depot associate who is working in the connectors section spends 15% of their daily working hours assisting customers with identifying and locating a particular screw or bolt, then installing the Automated Connector Type Identification device in the connectors section of the store would enable those customers to quickly locate the product they require while also allowing the Home Depot associate to perform other job functions with that time. If such an associate is paid $20,800 per year, then over the course of a year, the associate would be able to perform $3,120 of additional work that they could not have before. However, this assumes that the device functions properly and entirely eliminates customers' need for employee assistance in this particular area.

8.2 Cost Analysis

Table 1 shows a breakdown of the cost of the components required to construct the prototype, which totals $198.69. The total pricing cost for power supplies, cables/wiring, miscellaneous parts (such as a single resistor or LED bulb), and packaging/housing is an estimated value. Precise values are not given because these components were procured without cost from the Senior Design Lab and the Invention Studio.

Item	**Cost**
Raspberry Pi Model B+	$35.00 [11]
Raspberry Pi 8 MP Camera	$29.95 [7]
7” LCD Touchscreen	$89.95 [12]
100g Micro Load Cell	$7.00 [9]
8GB Micro SD Card	$6.79 [10]
Power Supplies / Cables / Wiring / Housing	$20.00
Packaging / Housing	$10.00
Total	**$198.69**

Table 1. Cost breakdown for prototype components.

Five engineers completed the design and development of the prototype. The total labor hours for the project are included in Table 2. Testing and debugging filled the largest amount of the production phase due to the complexity of the programming needed for the user interface and communication with the server.

Task	**Hours**
Group Meetings	200
Reports / Presentation	25
Hardware Assembly	5
Device Housing Fabrication	5
Software Design	50
Testing / Debugging	100
Total	**$198.69**

Table 2. Hours spent per task.

Labor costs were calculated using the total labor hours from Table 2 and an assumed cost of $40 per hour, which totals $15,400. Adding that figure to the estimated component cost of $198.69 from Table 1 equals a total prototype development cost of $15,598.69.

Because the device was developed specifically for Home Depot and is not being sold to any other party, there is no intention of producing it for mass-market sale. Therefore, there is no consideration being given to sales expenses, profits, or any other factors involved with selling the device in that a manner.

The estimated component and labor cost of Home Depot creating one unit of the device is approximately $298.69. This is based on the component cost of $198.69 plus an estimated 10 hours of assembly and testing labor at an assumed rate of $10 per hour.

9. Conclusion

9.1 Results

The final prototype created by Sensor Team Six exceeded our goal of 85% accuracy for the range of screws selected. The most frequent failures come from the head type being incorrectly identified. Additionally, the average processing time was under five seconds, but the max time was greater than our goal. The interface provided detailed results that include the image taken by the device, the final outline created by the image analysis, a stock image from The Home Depot website, and the exact name of the screw.

During the Georgia Tech Capstone Expo, the prototype was extremely successful both in screw identification as well as customer satisfaction. Expo attendees were encouraged to approach the prototype and use it as if they were a Home Depot customer by choosing a random screw and using the prototype to classify it. Once classified, the result was verified by comparing the name of the classified screw to the array of products the team had brought to the Expo. The volume of attendees who were interested in trying the prototype as well as the positive feedback after using it suggests that this is a device that could improve customer experience in a Home Depot store.

9.2 Future Work

Future work could be done by The Home Depot to reduce some of the failures of our prototype and expand the device to identify more connectors. The expansion of this work is facilitated by the existence of the current prototype which can capture consistent images of screws. Accruing large amounts of data in the form of images is essential to some of the future work.

9.2.1 Head Type

To solve our head type issue, a deep learning technique could be employed for identifying the head type of the screw. By acquiring many images of different head types and then manually identifying them, a deep learning algorithm (such as a neural network) could be constructed. Since this was the largest failing of the current prototype, focusing on this parameter could result in improved accuracy, especially as more and more screws are added to the database.

9.2.2 Material

The material of the screw is unidentifiable by the current prototype. An outline of the screw, which is what the image analysis produces, is not sufficient. New software or hardware could be added to identify the material of the screw. A load cell would allow for a seventh parameter, weight, to be included in the list of parameters. The prototype could then discern screws that are identical except for material type. Additionally, having the weight as a parameter in the training of the classifier would mean that it could more accurately discern between screws of different sizes and therefore weights. A software solution for determining the material is to use texture analysis. If the device knows the color of the screw, it could use that data to identify the material type, which generally varies with color. However, if a user places a painted screw into the device, texture analysis would be ineffective.

9.2.3 Driver Type

The screw driver type is another parameter of the screw that is not identified with the final prototype. Because Sensor Team Six determined that the customer should not be required to place the screw in a specific orientation to make the interaction more convenient, it cannot be guaranteed that that the driver type of the head is always visible. To tell the difference between a Philips or slot head, either this quality of life feature needs to be removed or multiple perspectives are required. In regard to the multiple perspective approach, independent cameras can be mounted to the device. Instead of one camera in the center, a camera can be attached in each corner. One camera would serve as the primary source that would determine the orientation of the screw and parameters from the outline according to the process implemented in the current design. Once the orientation is known, the camera that can view the driver type on the top of the head with take a picture to be processed. Additionally, a single moving video perspective can be applied to simulate a device with multiple cameras. This approach would not be recommended for a physical apparatus in Home Depot stores, however, this could be useful for the phone application that that Home Depot is considering in conjunction with this physical device.

9.2.4 Bolts, Nuts, Washers

In the future, The Home Depot could expand the prototype to identify multiple types of connectors (screws, bolts, washers, etc.), with the current implementation being a submodule for screws. A deep net could first identify what type of connector has been placed on the stage. Once the type of connector is determined, then tailored parameters could be extracted from the image (such as thread count, width, etc. for screws) and then run through a custom classifier.

9.2.5 Mobile App Tradeoffs

The current prototype is a hardware and software solution that requires external power and would only be found in a Home Depot store. This means that customers would have to come into a store to identify and buy replacement parts. A mobile app could be developed that would use the phone's built-in camera to identify screws in any location and offer an opportunity to buy the screw right away. The lack of friction in identifying and purchasing within an app makes it extremely attractive. However, there are significant obstacles on the way to achieving this. All hardware aspects involved in the current prototype, including lighting, fixed camera distance, and plain background, facilitate the extraction of parameters from the screw. By removing the standard setup with a mobile phone app, the accuracy of the parameter extraction would suffer.

Overcoming these obstacles could be achieved by developing a deep learning network. With enough data, a deep learning network could be robust enough to handle the variable distance to the screw and changing background. The major issue with this approach is the acquisition of enough data in the form of images. Acquiring images could be started by using a variation of the current prototype in production setting. By having customers use this device, high quality images could be compiled quickly, and the validity of the deep learning approach could be probed.

10. Leadership Roles

Each team member was delegated a leadership role and technical role(s) for the project.

- Joseph Doughty: Group Leader and Hardware
- Christopher Fox: Communications Liaison, DSP Algorithms, and Image Processing
- Nicholas Korzik: Treasurer, Webmaster, Object Illumination, and Data Acquisition
- Matt McBrien: Documentation Coordinator and Software Design
- Lindsey Robirds: Wireless Communications, and User Interface

11. References

[1] C. Hoy, S. Hurst, N. K. Jung, B. Kim, J. Kim, and E. Norby. ECE 4011. Final Class Presentation, Topic: “Automated Identification of Screws.” School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, Apr. 14, 2017.

[2] C. Hoy, S. Hurst, N. K. Jung, B. Kim, J. Kim, and E. Norby, “Automated Identification of Parts,” Georgia Institute of Technology, Atlanta, GA, Tech. Report. 14 Apr. 2017.

[3] A. Krizhevsky, I. Sutskever, G. E. Hinton, “Imagenet classification with deep convolutional neural networks.” In Proc. Advances in neural information processing systems, 2012, pp. 1097-1105.

[4] Amazon, “Cloud Compute with AWS” *Amazon.com*, 2018. [Online]. Available: https://aws.amazon.com/products/compute/?nc2=h_l3_db. [Accessed Apr. 15, 2018].

[5] Google, “Compute Engine” *Google.com*, No Date. [Online]. Available: https://cloud.google.com/compute/. [Accessed Apr. 15, 2018].

[6] Raspberry Pi, “Raspberry Pi 3 Model B+” *Raspberrypi.org*, No Date. [Online]. Available: https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/. [Accessed Dec. 9, 2018].

[7] Adafruit, “Raspberry Pi Camera Board v2 - 8 Megapixels” *Adafruit.com*, No Date. [Online]. Available: https://www.adafruit.com/product/3099. [Accessed Apr. 11, 2018].

[8] Adafruit, “HDMI 4 Pi: 7" Display w/Touchscreen 1024x600” *Adafruit.com*, No Date. [Online]. Available: https://www.adafruit.com/product/2395. [Accessed Apr. 11, 2018].

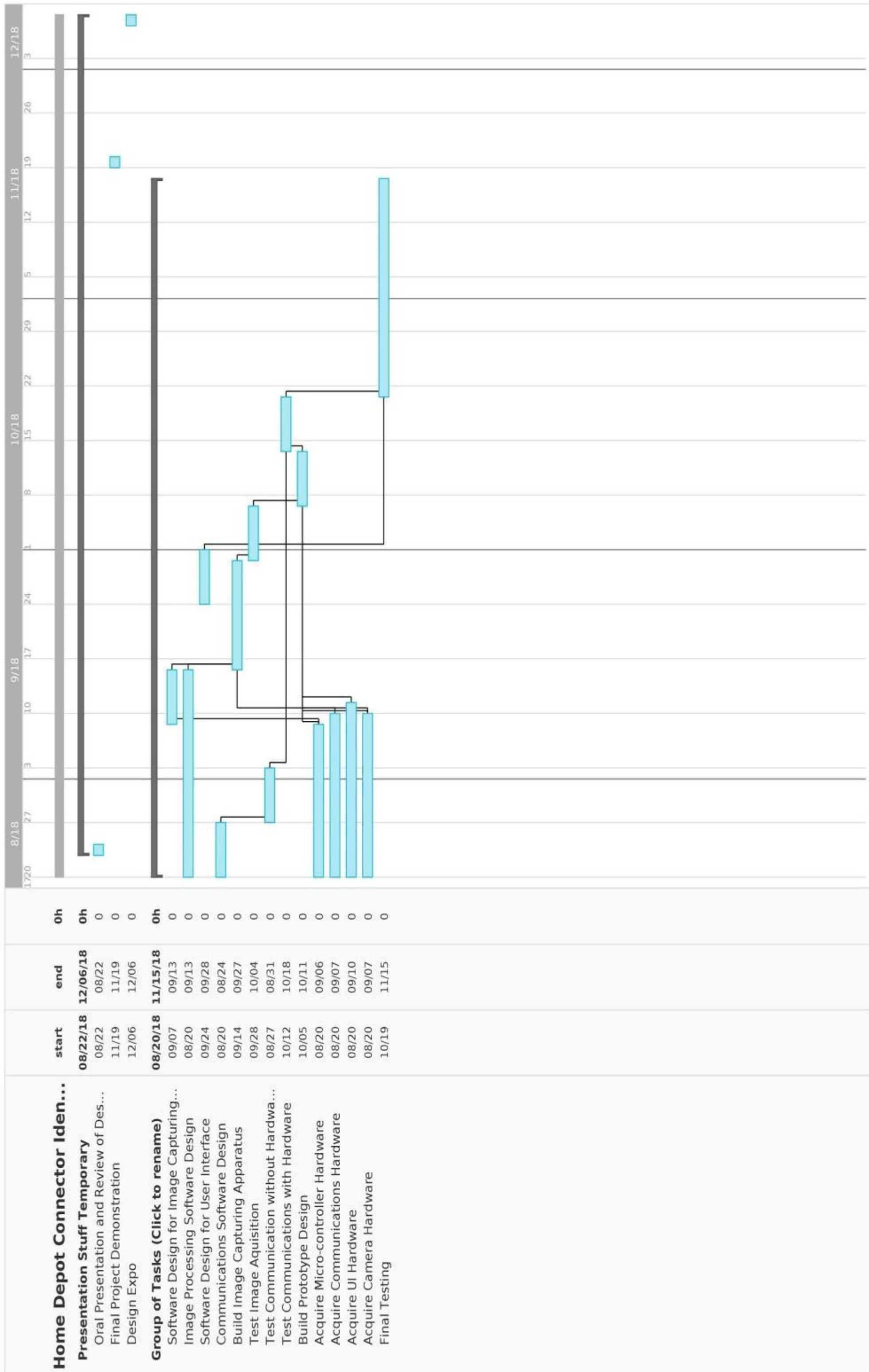
[9] RobotShop, “100g Micro Load Cell” *RobotShop.com*, No Date. [Online]. Available: https://www.robotshop.com/en/100g-micro-load-cell.html. [Accessed Apr. 15, 2018].

[10] Amazon, “SanDisk® microSDHC™ 8GB Memory Card” *Amazon.com*, No Date. [Online]. Available: https://www.amazon.com/SanDisk%C2%AE-microSDHCTM-8GB-Memory-Card. [Accessed Apr. 11, 2018].

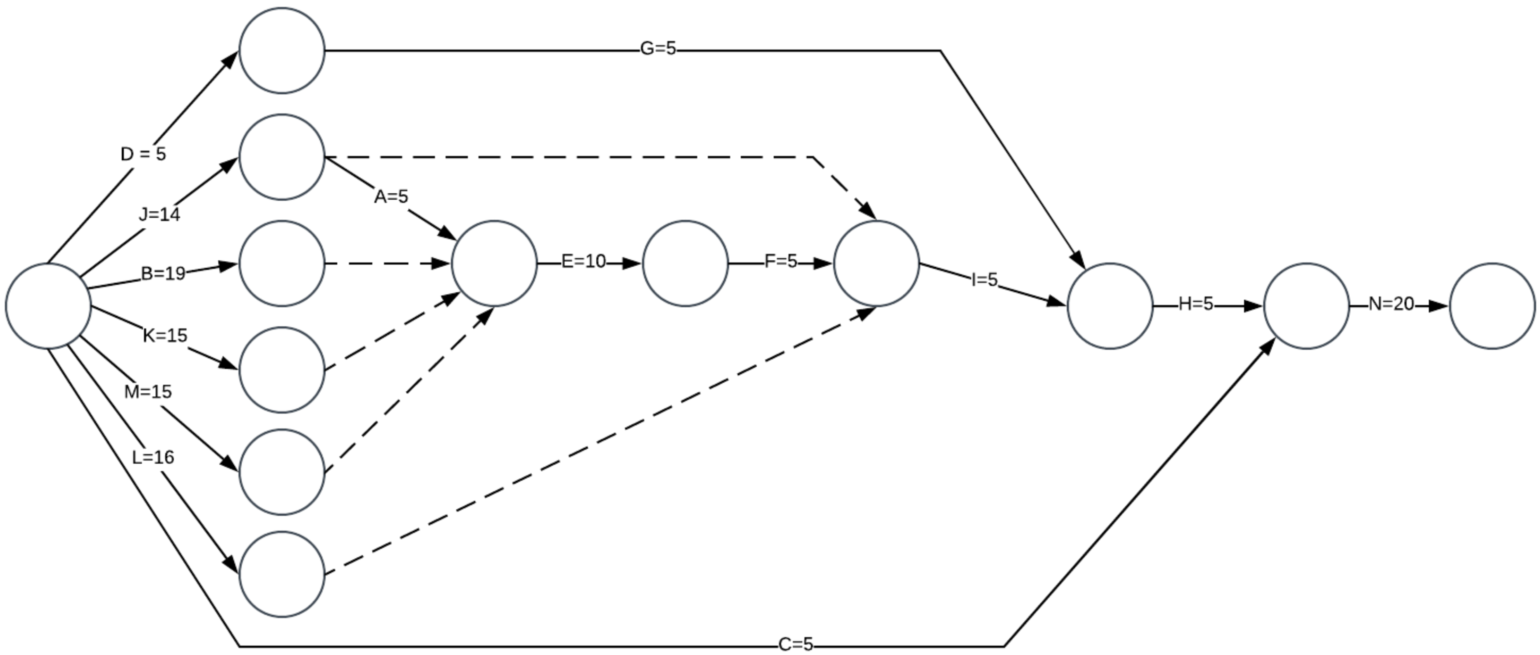
[11] Raspberry Pi, “Raspberry Pi 3 Model B+” *Raspberrypi.org*, No Date. [Online]. Available: https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/. [Accessed Dec. 9, 2018].

[12] Amazon, “Raspberry Pi 7’’ Touchscreen” *Amazon.com*, No Date. [Online]. Available: https://www.amazon.com/Raspberry-Pi-7-Touchscreen-Display/dp/B0153R2A9I. [Accessed Dec. 9, 2018].

Appendix A - Gantt Chart

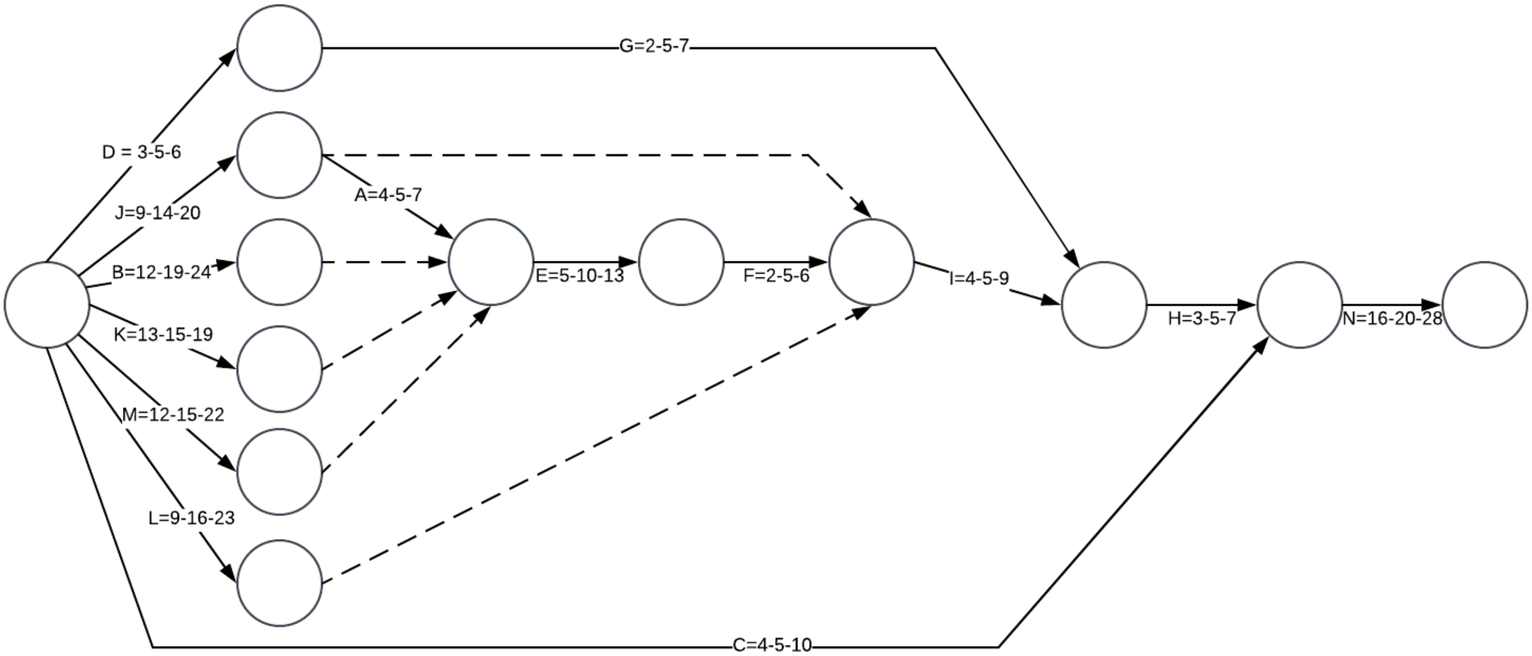


Appendix B - CPM Chart



Node Letter	Corresponding Task
A	Software Design for Image Capturing via Hardware
B	Image Processing Software Design
C	Software Design for User Interface
D	Communications Software Design
E	Build Image Capturing Apparatus
F	Test Image Acquisition
G	Test Communication without Hardware
H	Test Communications with Hardware
I	Build Prototype Design
J	Acquire Microcontroller Hardware
K	Acquire Communications Hardware
L	Acquire UI Hardware
M	Acquire Camera Hardware
N	Final Testing

Appendix C - PERT Chart



Node Letter	Corresponding Task
A	Software Design for Image Capturing via Hardware
B	Image Processing Software Design
C	Software Design for User Interface
D	Communications Software Design
E	Build Image Capturing Apparatus
F	Test Image Acquisition
G	Test Communication without Hardware
H	Test Communications with Hardware
I	Build Prototype Design
J	Acquire Microcontroller Hardware
K	Acquire Communications Hardware
L	Acquire UI Hardware
M	Acquire Camera Hardware
N	Final Testing

Appendix D - Confusion Matrix

Predicted	flathead #10 1-1/2	flathead #10 1-3/4	flathead #10 2-1/2	flathead #10 3	flathead #6 1	flathead #6 1-1/2	flathead #6 1/2	flathead #8 1/2
True								
flathead #10 1-1/2	100	0	0	0	0	0	0	0
flathead #10 1-3/4	0	66.66666667	0	0	0	0	0	0
flathead #10 2-1/2	0	33.33333333	100	0	0	0	0	0
flathead #10 3	0	0	0	100	0	0	0	0
flathead #6 1	0	0	0	0	50	0	0	0
flathead #6 1-1/2	0	0	0	0	0	100	0	0
flathead #6 1/2	0	0	0	0	0	0	100	0
flathead #8 1/2	0	0	0	0	0	0	0	66.66666667
hexhead #10 1/2	0	0	0	0	0	0	0	0
hexhead #14 1-1/2	0	0	0	0	0	0	0	0
ovalhead #10 1	0	0	0	0	0	0	0	0
ovalhead #10 1-1/4	0	0	0	0	0	0	0	0
ovalhead #12 1-1/4	0	0	0	0	0	0	0	0
ovalhead #6 1	0	0	0	0	50	0	0	0
ovalhead #8 2	0	0	0	0	0	0	0	0
panhead #12 1-1/2	0	0	0	0	0	0	0	0
panhead #12 1-1/4	0	0	0	0	0	0	0	0
panhead #12 3	0	0	0	0	0	0	0	0
panhead #12 3/4	0	0	0	0	0	0	0	0
panhead #4 3/8	0	0	0	0	0	0	0	0
panhead #6 1-1/2	0	0	0	0	0	0	0	0
panhead #6 1/2	0	0	0	0	0	0	0	0
panhead #6 5/8	0	0	0	0	0	0	0	0
panhead #8 1-1/4	0	0	0	0	0	0	0	0
panhead #8 3/8	0	0	0	0	0	0	0	33.33333333

hexhead #10 1/2	hexhead #14 1-1/2	ovalhead #10 1	ovalhead #10 1-1/4	ovalhead #12 1-1/4	ovalhead #6 1	ovalhead #8 2	panhead #12 1-1/2	panhead #12 1-1/4
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
100	0	0	0	0	0	0	0	0
0	100	0	0	0	0	0	0	0
0	0	100	0	0	0	0	0	0
0	0	0	100	0	0	0	0	0
0	0	0	0	100	0	0	0	0
0	0	0	0	0	100	0	0	0
0	0	0	0	0	0	100	0	0
0	0	0	0	0	0	0	100	0
0	0	0	0	0	0	0	0	100
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

panhead #12 3	panhead #12 3/4	panhead #4 3/8	panhead #6 1-1/2	panhead #6 1/2	panhead #6 5/8	panhead #8 1-1/4	panhead #8 3/8
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
100	0	0	0	0	0	0	0
0	100	0	0	0	0	0	0
0	0	100	0	0	0	0	0
0	0	0	100	0	0	0	0
0	0	0	0	100	0	0	0
0	0	0	0	0	100	0	0
0	0	0	0	0	0	100	0
0	0	0	0	0	0	0	100