More Than Just A Chopping Board

By
Suzy Ahn **(sba2)**
Rishabh Anand (**rishabh3**)
Vatsala Verma (**vatsala2**)

Final Report for ECE 445, Spring 2021
Professor: Dr. Rakesh Kumar
TA: Anand Sunderrajan

Date: May 5th, 2021
Project Number: 14

Abstract

Our project “More Than Just A Chopping Board” is an innovative solution to chopping vegetables. It includes features such as a vegetable detection system, chopping style suggestion for the detected vegetables, and four different chopping styles for the user to choose from.

Our project’s physical design resembles that of a 3D printer. The motions of the chopping board are controlled by the x and y rails using motors on each end. The z-axis motion is controlled by the linear actuator, which has a removable blade. The chopping board in the system is also detachable for easy cleaning. The acrylic lid protects the user from accidentally hurting themselves while the chopping is in progress.

For further future implementation of this product, many small changes would help run our assembly a lot more smoother. The recognition of more vegetables, and even fruits, and the ability to chop with more chopping styles would open this product for greater use. To fully market this product to its greatest potential, we would want to add multiple safety features such as an emergency stop button and waterproofing the system.

Table of Contents

1. Introduction

A study done in 2012 [1] found that there were about 1190 cases of knife-related injuries per day, with about 280 cases occurring in the kitchen. These injuries included lacerations on fingers/hands. Some of these injuries are sustained by individuals in the kitchen when they miss the vegetable while chopping it.

The COVID-19 pandemic caused a massive shutdown of restaurants and increased health concerns. This resulted in an increase in the number of people who started cooking at home. Seventy-one percent of consumers [2] stated that they would continue cooking at home often even after the pandemic is over. We decided to make this product to create an enjoyable cooking experience with fewer injuries.

Our product, “More than a chopping board”, makes cooking easier by giving consumers a tool to automate cutting vegetables. With our project we plan to minimize the injuries and preparation time that comes with chopping, so that cutting vegetables is not what gets in the way of cooking a meal.

Following are the high-level requirements we had set for our project. Our system was able to meet all of these requirements.

- Our assembly must be able to recognize when a vegetable has been inserted. It must also be able to distinguish between **six** common vegetables: potatoes, tomatoes, onions, lettuce, cucumbers, and cabbage. Our system must offer among **four** different chopping styles along with specific recommendations for the detected vegetable: large dice, medium dice, small dice, and Batonnet [3]. Lettuce and cabbage will only be chopped horizontally.
- Our assembly should achieve **75%** accuracy in the dimension of the chopped vegetable when compared to the **dimensions** defined in French cooking [3]. This 75% accuracy in dimension does not apply to chopping lettuce and cabbage.
- Our system should be able to finish the entire process, from when the vegetable has been placed on the chopping board to when the chopping is done, in **6 minutes**. In calculating this value, we have assumed a runtime of 0.5 minutes for our vegetable detection program, 0.5 minutes for the user to confirm the style of chopping, and 4 minutes for chopping the vegetable under ideal conditions, and an additional minute for any delays on the user’s end.

Our project comprises four main units: Power Supply, I/O Unit, Chopping Assembly, and Control Unit. These units are all shown in the block diagram in Figure 1. The I/O Unit and control unit work together to detect and distinguish the vegetables placed in the chopping region from our list of six vegetables mentioned previously. The I/O unit, chopping assembly, and control unit together allow the user to select the chopping style, from the four chopping styles offered by the system, and then chop the vegetable accordingly. The chopping assembly and control unit ensure that the vegetables are chopped with the highest level of accuracy in dimension as mentioned previously in the high-level requirements. The power unit powers up all the other units by providing the specific voltage and current required for each of the subsystems to work efficiently. All of these units together achieve the high-level requirements set for the system.

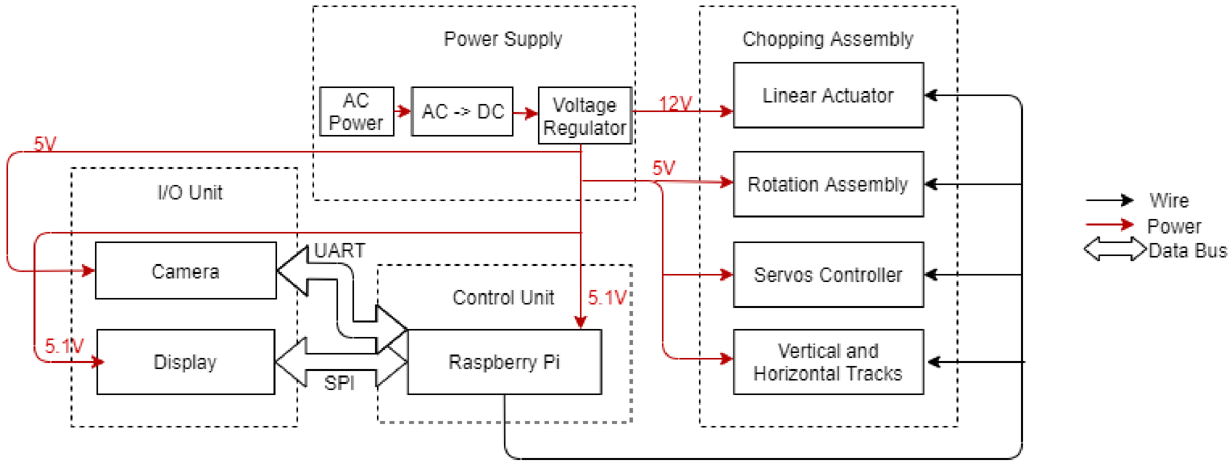

Figure 1. Block Diagram for More Than A Chopping Board

2. Design

The links to all the code used in the different subsystems of the project can be found in Appendix A. It also includes the pseudo code for the edge detection and vegetable detection codes.

2.1 Design Alternatives

While scoping our project, we ran into multiple problems with the mechanical portion of our design. Among them, the most notable ones are the following:

- The original assembly was meant to include a bigger linear actuator. This way, the force applied to cut the vegetable was higher which would make chopping hard vegetables easier. Unfortunately, due to the size of the assembly, motion along the z-axis would not be possible for the linear actuator if we were to use one that was longer than 2 inches.

- The problem of holding down the vegetable, in case the position of the vegetable changed while chopping, came up during the initial planning process. After considering different mechanical aspects to hold down the vegetable, we finally decided to use a software approach to find the offset to the next cut's location in case the vegetable had moved.

2.2 Power Supply

Our power supply was initially designed to take an AC input, convert it to DC, subsequently regulating the voltage to 12V and 5V in order to supply the correct power to our different components.

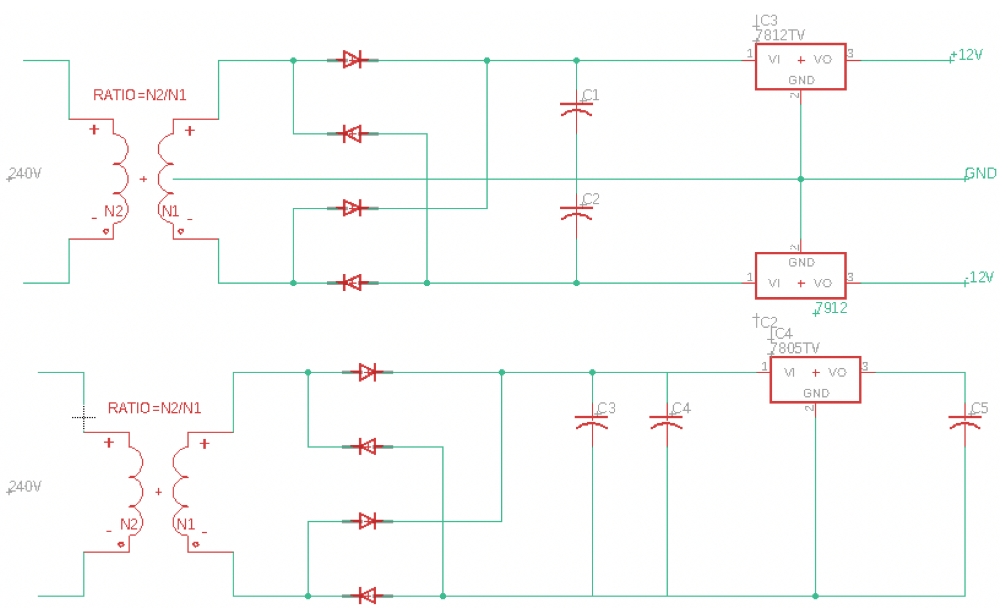

Figure 2. AC to DC Multi Power Supply (Output 5V and 12V)

The initial power supply design shown in Figure 2, had to be changed due to size constraints imposed by the transformer. With the step down required to convert the AC to DC for our voltage regulators, the required transformers were too large to go on the PCB. The added heat management and complexity from this design also lead us to use off-the-shelf AC to DC power supply instead. We were able to subsequently step down the voltage using that and were able to boost the current to power our various components.

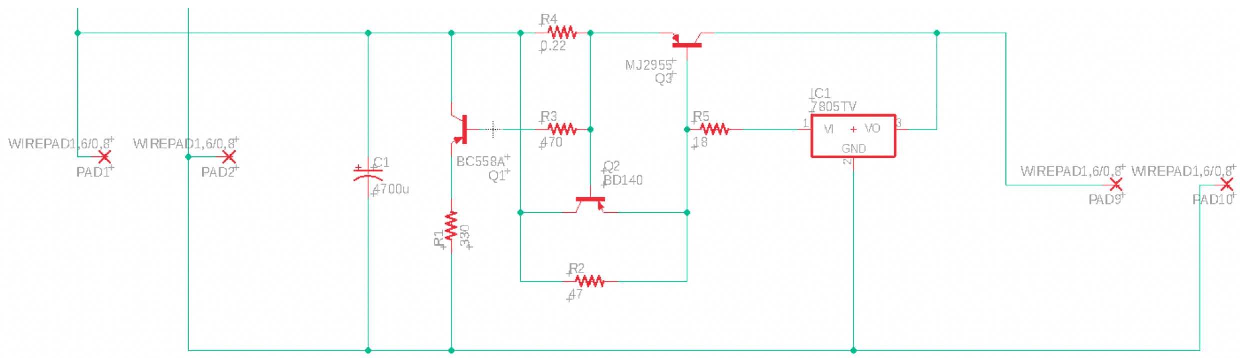

Figure 3. 12V DC to 5V at 3A Power Supply

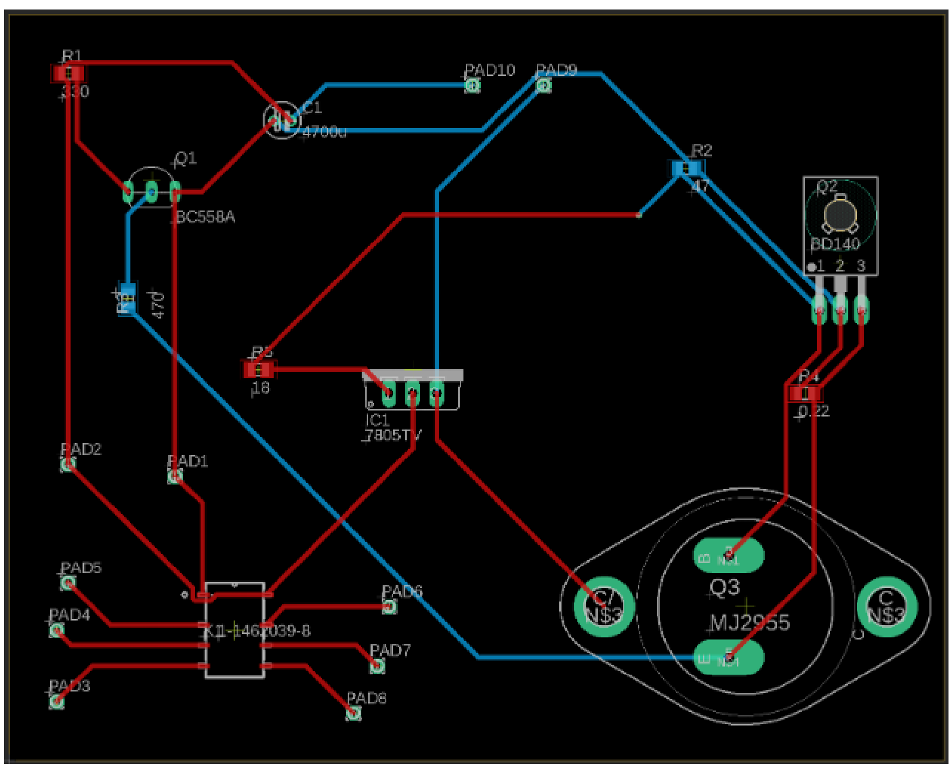

Figure 4. 12V DC to 5V at 3A Power Supply PCB

This power supply design used a combination of bipolar junction transistors (BJTs) including MJ2955, BD140, and BC558, and capacitors as shown in Figures 3 and 4 to boost the current through the 7805 IC from 5V at 1.5A to 5V at 3A. The input to the power supply was then taken in parallel to the rest of the components which required 12V.

2.3 Control Unit

The physical component used for the control unit was the Raspberry Pi computer. The Pi was loaded with a program to enable interaction with the I/O unit and the Chopping Assembly.

The first program to interact with the I/O unit is the Vegetable Detection program. Upon inserting the vegetable, the display asks the user to confirm that a vegetable has been placed in the chopping area. Once a confirmation is received from the user, the vegetable detection takes place and relays back to the display unit to inform the user of the detected vegetable and its recommended cutting style. The pseudocode for this algorithm can be found in Appendix A: Algorithm 1.

This module was implemented with machine learning within the Tensorflow library. The generated graphs in Figure 5 show the model accuracy and model loss, and it was observed that we were able to reach an accuracy of 1.0. We had trained 23 classes of vegetables initially, but only allowed certain vegetables to be labeled due to the limited style of cuts we were going to offer according to our high-level requirements.

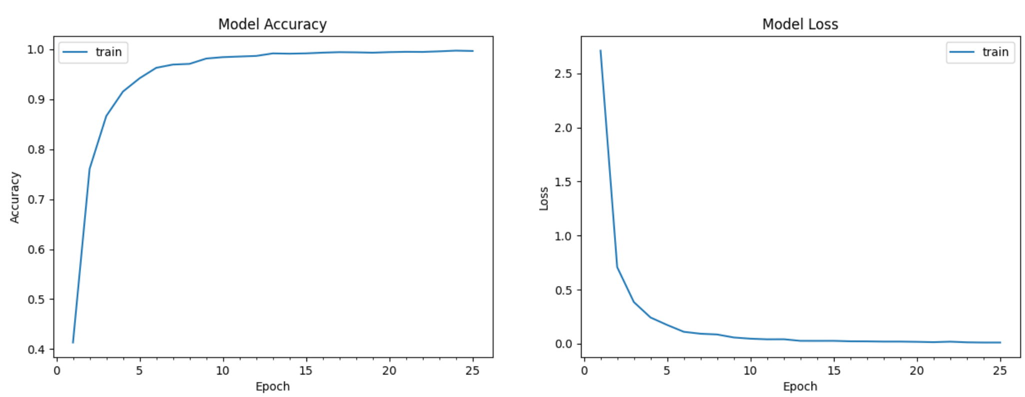

Figure 5. *Model Accuracy and Model Loss of the Vegetable detection program*

The control unit also sends the starting location of the vegetable and the user selected cutting style to the chopping assembly. The starting location of the vegetable is calculated by the edge detection algorithm. With a reference such as a sticker of a fixed size, the algorithm, located in Appendix A: Algorithm 2, is able to calculate the size and the starting location once the vegetable has been placed in the system.

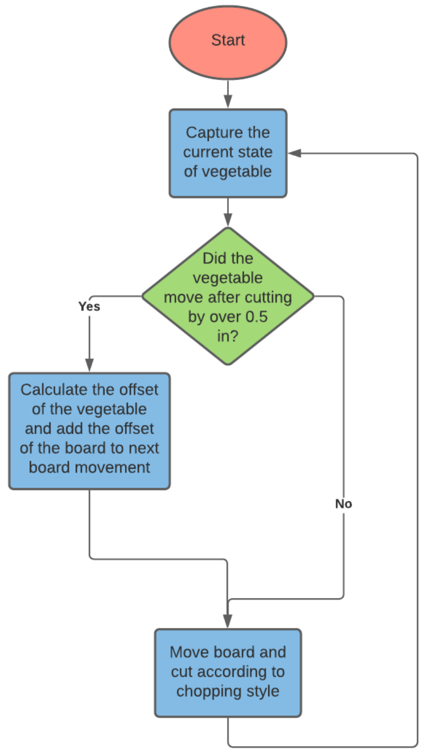

Figure 6. Flowchart for edge offset calculation

With every chopping motion, the camera will capture an image to relay back to the control unit to be analyzed as shown in Figure 6. If the offset of the vegetable, before the cut is made and after the cut is made, is greater than 0.5 inches, the control unit will guide the chopping assembly to move the chopping board by the offset. With an image being captured with every upward motion of the blade, we can ideally capture the vegetable with each individual cut, and move the board accordingly to achieve 75% dimensional accuracy even if the vegetable slides around in the chopping area.

2.4 Input/Output (I/O) Unit

The I/O unit comprises a camera and a display screen with buttons attached to it. Figure7 shows the entire flow chart of the functionality of the display unit.

The camera is positioned such that the system can get a clear visual of the chopping area. The camera was crucial for object detection and edge detection which were a part of the high-level requirements.

The display unit used for our assembly was a 1.8 inch LCD module [4]. There are two push buttons in the display unit; one is used to toggle between different options and the other one is used to confirm the user's choice.

A regular LCD module with push buttons was chosen over a touchscreen interface. This design choice was made because water proofing the touchscreen display would be necessary yet difficult. Since this product is made to be used in a kitchen, ensuring that it is away from water is impossible. The heavy motions of the chopping assembly could easily damage the touchscreen abilities of the system and render the user interface part of the system useless.

Push buttons on the other hand can be waterproofed more easily and remain unaffected by the motions of the chopping assembly. Considering all of these factors, using push buttons with a regular LCD display seemed to be the smarter choice.

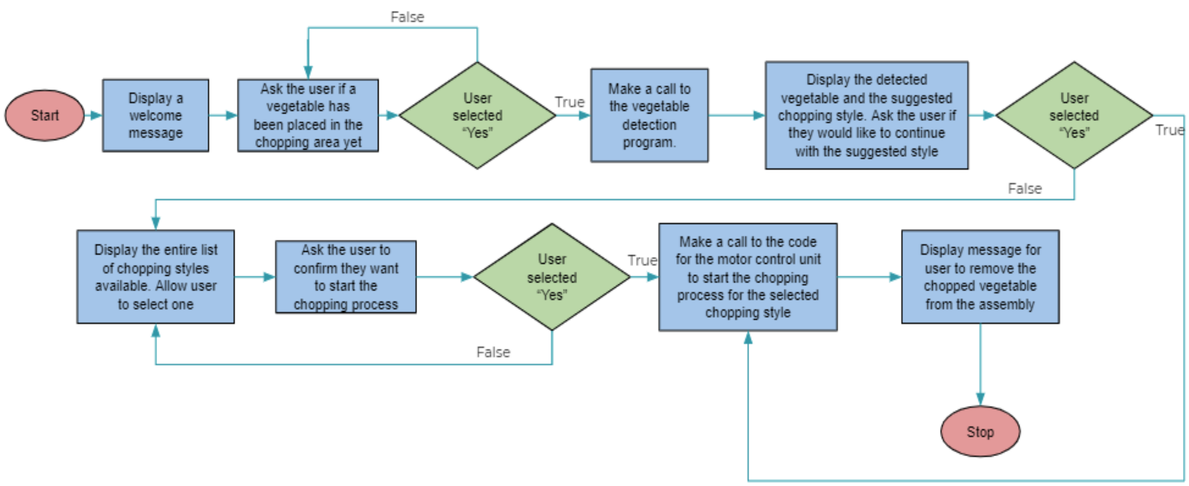

Figure 7. Flowchart for the functionality of the display unit

2.5 Chopping Assembly

The chopping assembly consisted of 3 stepper motors, 1 servo motor and a linear actuator. The motors worked together to move the chopping board along a defined track depending on the selected chopping style. The linear actuator was used to extend and retract the blade to make the desired cuts.

The 3 stepper motors were used to move the chopping board in the X and Y axes; one motor was used for motion along the X axis, and two motors worked together for motion along the Y axis. These motors were controlled with the help of three A4988 chips, the firmware for which was written in Python using the GPIO and RpiMotorLib libraries[5].

The motor control code is used to drive the step and direction pins on the chip as shown in Figure 8. The enable chip is an active low. By setting the direction pin to high or low we make the motors rotate clockwise or counterclockwise while the number of steps decide how much the motor is going to rotate giving us fine control over the dimensions of the cuts. It took the motor 1125 steps to cover 22.5 centimeters which is the entire length of the X axis rail and formed the basis for the calculation on the required number of steps for each cut.

The servo motor used to rotate the chopping assembly was a mini servo connected directly to the Raspberry Pi with no need for firmware since it can be controlled with pulse width modulation to set specific angles of rotation.

Since the linear actuator did not have any form of built-in control logic, it had to be wired to a double pole double throw (DPDT) relay to switch polarities for extension and retraction.

As shown in Figure 9, the DPDT relay was wired to a 12V power supply with the coil being controlled with the help of our Raspberry Pi in order to extend or retract the blade based on a time based loop. With our 2inch linear actuator extending at a speed of 12mm/s the Pi would switch the relay after 4.5 seconds in order to ensure that a full cut had been made.

Given that the linear actuator can draw a maximum of 3A, we chose the PB1174TR DPDT relay which was rated for 12V at a 5A contact current.

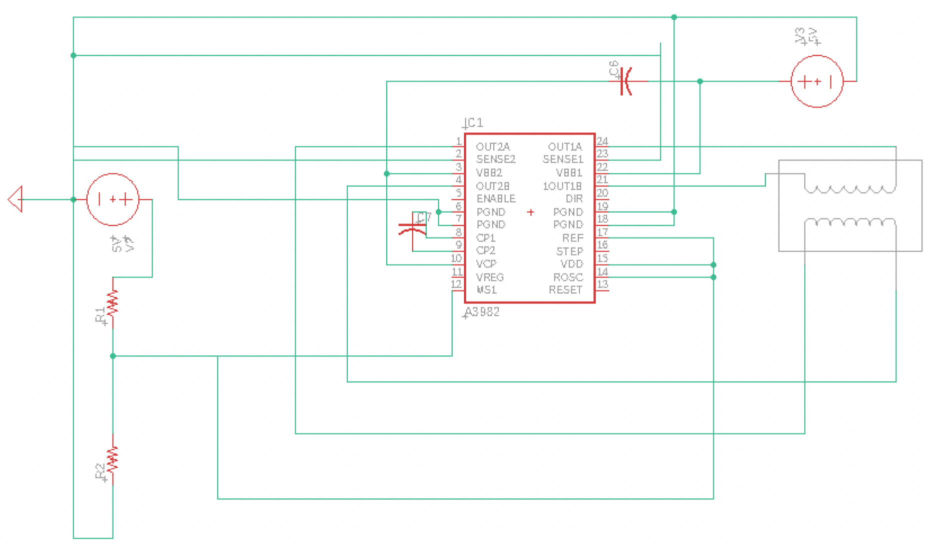

Figure 8. A4988 Driven Stepper Motor (Enable, Step, and Dir from Raspberry Pi)

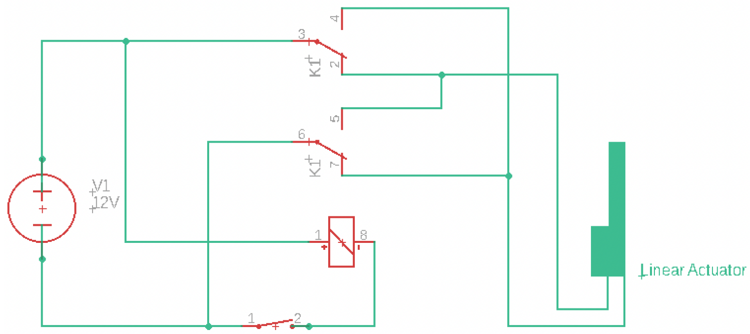

Figure 9. 12V DPDT Relay Controlled Linear Actuator

3. Design Verification

The requirements and verification table for all the subsystems can be found in Appendix B.

3.1 Power Supply Unit

1. The first requirement for the Power supply was that it should be able to supply 5V at 3A to the Raspberry Pi. We verified this by connecting the power supply to a digital multimeter and probing for voltage which was at a steady 5.024V. While we weren't able to test the power supply's current output since it was load determined, we were able to ensure that the PCB was functioning properly by wiring the power supply and Pi and checking for system stability.

2. The second requirement for our power supply was to be able to supply 12V to the stepper motor circuits and the linear actuator which was also verified by probing the parallel connections for the same from the input of the PCB.

3.2 Control Unit

1. The first requirement for the control unit was that the vegetable inserted should be detected and the user should be given the option of the recommended cutting style for the detected vegetable. This unit was able to receive the image captured by the camera of the I/O unit and feed the image through the pre-trained model after some pre-processing to properly identify the vegetable inserted and give cutting style recommendations based on it.

2. The second requirement for the control unit was to send commands to the chopping assembly to guide the motion of the blades for cutting the vegetables. To verify this requirement we placed a vegetable on the board and after the detection process, the chopping board was able to move such that the starting edge of the vegetable was directly beneath the blade to start the chopping process.

3.3 Input/Output (I/O) Unit

1. The first requirement for this unit was that the camera should be able to send a visual feed of the general chopping area to the control unit. To verify this we connected the camera to the Raspberry Pi and connected the Raspberry Pi to a monitor. We then ran a test code to take an image from the camera. The image successfully captured the target region and we were able to verify this requirement.

2. The second requirement for this unit was that it must display the option to select between the chopping styles and display the recommended chopping style. To verify this requirement we

started by connecting the screen and buttons to the Raspberry Pi. We then ran the complete code for the display unit's functionality and made necessary selections. On running this code, the user was shown the recommended chopping styles and was also able to toggle through the menu with the different chopping styles. We were thus able to verify this requirement.

3. The final requirement for the display unit was that it must allow users to confirm the selected chopping style. To verify this, we continued using the set up from the previous verification and got the expected results once the user confirmed the chopping style. Therefore, we were able to verify this final requirement for the I/O unit.

3.4 Chopping Assembly

1. The first requirement for the chopping assembly was that the linear actuator must be able to exert the required amount of force to chop a vegetable (at least 300N) which was tested by placing a bell pepper in the assembly and chopping it. The linear actuator was able to achieve this quite easily with a very low current draw compared to its peak rating. This helped us establish that the linear actuator would be able to exert the required force for any vegetable.

2. The second requirement for the chopping assembly was that it should not draw more than 12V at 3A in total. We were able to test this by wiring the power supply to the assembly and running it for approximately 10 minutes, testing for system stability which would have failed had the assembly drawn more than the designed power output of our power supply.

3. The final requirement for the assembly was to be able to chop the vegetables in 4 minutes with up to 75% accuracy of the dimensions specified in French cooking [3]. We tested this by placing a bell pepper on the chopping board and selecting the small dice chopping style. Small dice is the finest cut offered by our system and therefore the system consumes the most amount of time in its execution. The dimensions for the cut were quite accurate with most cuts 0.25 inches wide. However the time taken for the total assembly to run was higher than our estimate of 4 minutes.

4. Costs

4.1 Parts

Table 1. Cost of the parts used

Part Name	Manufacturer	Retail Cost ($)	Bulk Purchase ($)
Linear Actuator	Sipye	43	35
Nema 17 Stepper Motor (3-pack)	RTELLIGENT	26	18
Raspberry Pi 4B (4GB)	Raspberry Pi	35	35
Raspberry Pi Camera Module	Arducam	18	10
1.8 inch LCD Module	Waveshare	9	8
Stepper Motor Driver	HiLetgo	5	3
ICs (7812, 7912, 7805)	STMicroelectronics	2	1.5
Total		**138**	**110.5**

4.2 Labor

By observing the salary averages for ECE graduates from UIUC [6], we concluded that an hourly wage of $48 is a safe assumption. We estimate that we worked on this project for at least 15 hours per week for 12 weeks.

Table 2. Cost of salary and time spent

Hourly Wage	Total hours spent on the project	Number of team members	Total Labor Cost
48	15*12 = 180	3	$25,920

5. Conclusion

5.1 Accomplishments

Throughout the 12 weeks, we were able to achieve most of the goals we had set for this project. With the control unit, the I/O unit and the chopping assembly working together, we were able to achieve the core functionality we had proposed for the project. Given the circumstances of limited lab hours and the fair amount of equipment burnout, we were able to achieve a high level of completion.

5.2 Uncertainties

Due to the nature of this project being heavily mechanical, there were shortcomings that we could not mend due to the lack of proficiency in mechanical engineering. This included the software consideration for the mechanical aspect of the vegetable cutting process, which should be solved mechanically in later iterations of this project.

5.3 Ethical Considerations

Possible safety concerns of this project include injuries caused by sharp objects. In the ACM Code of Ethics 1.2 [7], we are to “Avoid Harm”. Since we plan on providing blades along with the assembly of our project, the users will have to exercise the same precautions as they would around any other sharp objects.

Consumers could hurt themselves if they insert their hand into the assembly while chopping is in progress. We have eliminated this danger by making sure the users confirm their choice to begin the chopping process. This ensures that the users are aware that chopping has begun and refrain from placing their hand in the assembly. A message is also displayed once the chopping is complete to tell the user it is safe to remove the vegetables from the assembly.

There are a few general precautions necessary when operating this machine. Children should be kept away from this machine. The product has an acrylic door which needs to be opened before the chopping area can be accessed. In addition, the high guard rails designed to prevent the slipping of the vegetables have been calibrated such that it is difficult for someone to unintentionally put their hand in the two inches of spaces between the blade and the board.

This assembly could also cause electric shock if mishandled, especially since it will be placed in the kitchen. Since the vegetable inserted would most likely be washed, our product should be able to handle contact with water between the blade and the vegetable. This problem is dealt with by the blade being waterproof.

This product should not be submerged into water at any point. The user might shock themselves if the device is near water or if the user has an excessive amount of water on their hands to operate it. The blade and chopping board are detachable for easy cleaning and must be dried before they are placed back into the system.

Unfortunately, there are ethically gray areas about developing such products. Since we are not able to detect much more than an object and hand inserted, we cannot foresee what sort of objects are going to be cut. This product is only intended to be used on vegetables. This product is only suitable for food use and to avoid ethical breaches, there will be a written warning to insert only food items into the board to urge users to only put in items they deem appropriate.

The user will additionally be provided with a safety manual which will include warnings and precautions that need to be taken while using the product.

5.4 Future Work

Given the time and design constraints that we were working with, there were various aspects of the project where we see opportunities for future improvement. One aspect of the project we would like to improve vastly upon is safety. Currently, it is up to the users to keep themselves safe when the machine is in operation. The addition of a circuit breaker to the acrylic lid would provide additional safety. When users lift the lid to reach for the vegetable, the machine would pause its operation, thereby preventing any potential injuries the user could have sustained if chopping was in progress. Additionally, we would like to add an emergency stop button to the side of the assembly to add an extra layer of safety to the circuit breaker lid.

Aside from the safety issues, we want to reduce the time taken by the system to complete chopping a vegetable. In order to make the chopping process more efficient we would replace the single blade design with a multi blade design thereby achieving more cuts on a single extension of the linear actuator.

Given the modularity of the code we would like to expand the option of being able to detect even more vegetables and offer more chopping styles to better suit the user's needs.

Finally, having demonstrated a proof of concept outlining the ability of a machine to chop vegetables, we envision this project iterating to finally getting to consumers as part of a refrigerator. A multi blade design would eliminate the need for the Y-axis motion of the vegetables. Using a conveyor belt to carry vegetables across the blades would save space along the X-axis too. The vegetables would be stored in a vertical array inside the door of the refrigerator, being chopped in half vertically before dropping down to the conveyor belt, where the multi blade design would then be able to dispense the users' desired cuts with a single extension of the actuators.

References

[1] G. A. Smith, “Knife-Related Injuries Treated in United States Emergency Departments, 1990–2008,” The Journal of Emergency Medicine, vol. 45, no. 3, pp. 315–323, Jul. 2013.

[2] J. Devenyns, “Survey: 7 in 10 consumers say they will keep cooking at home after the pandemic,” Food Dive, 19-Jan-2021. [Online]. Available: https://www.fooddive.com/news/survey-7-in-10-consumers-say-they-will-keep-cooking-at-home-after-the-pand/593532/. [Accessed: 03-Mar-2021].

[3] R. Tan, “Kitchen Language: What Are The Basic Knife Cuts?,” MICHELIN Guide, 14-Jan-2019. [Online]. Available: https://guide.michelin.com/en/article/dining-in/kitchen-language-what-are-the-different-knife-cuts. [Accessed: 03-Mar-2021].

[4] 1.8inch LCD Module, Waveshare. [Online]. Available: https://www.waveshare.com/wiki/1.8inch_LCD_Module. [Accessed: 03-May-2021].

[5] RpiMotorLib. [Online]. Available: https://github.com/gavinlyonsrepo/RpiMotorLib. [Accessed: 06-May-2021].

[6] Grainger Engineering Office of Marketing and Communications, Salary Averages. [Online]. Available: https://ece.illinois.edu/admissions/why-ece/salary-averages. [Accessed: 06-May-2021].

[7] ACM Code 2018 Task Force, “ACM Code of Ethics and Professional Conduct,” Code of Ethics, 22-Jun-2018. [Online]. Available: https://www.acm.org/code-of-ethics. [Accessed: 03-Mar-2021].

Appendix A Links and Algorithms

The code for edge detection and vegetable detection can be accessed using the following link: https://gitlab.com/vatsala2/ece445_display/-/tree/master/display_code/VegeDetection.

The code used to control the display unit can be accessed through this link: https://gitlab.com/vatsala2/ece445_display/-/blob/master/display_code/examples/main.py.

The code used for controlling the chopping assembly can be accessed here: https://gitlab.com/vatsala2/ece445_display/-/blob/master/display_code/examples/motor_control.py.

The code used for controlling the chopping assembly for the tests can be found here:
https://gitlab.com/vatsala2/ece445_display/-/blob/master/display_code/examples/motor_test.py

Algorithm 1: Vegetable Detection Program

scan_image():

```
    Wait for confirm vegetable inserted
    Open webcam
    Capture image
    Write image in imgpath
    Return imgpath
```

model_test():

```
    Set image path (scan_image()):
    If saved model dir does not exist:
        model not found, exit
    Model = model_dir + model name
    Read test image
    Resize (100, 100)
    Image array = np.array (image)
    Pred = model.predict(image array)
    Get matching label of idx of pred.argmax(axis = -1)
    From label name, get recc. Cutting style
```

Algorithm 2: Edge Detection Program

Display_measurement:

```
    Preprocess image (cv2.canny)
    Find contours in edge map
    For each contour:
```

```
        If on the edge:
            Remove from contour list
        Find euclidean distance
        If first object:
            Calculate area
            If area is the similar to ref obj area:
                Move to the front of contour list
    Convert to tuple from list

    Compute rotated bounding box
    Draw the obj # on image
    Compute midpoints of each box
    Compute distance between midpoints
    If first contour:
        Set ref obj
        Calculate pixels per metric
    Find size of object
    Draw size on image
```

rect_edges(tl, tr, br, bl):

```
    Top = compare y or tl tr, whichever higher
    Bottom = compare y of bl br, whichever lower
    Left = compare x of tl bl, whichever one smaller
    Right = compare x or tr br, whichever larger
    Return left (start coordinate of vegetable cutting)
```

Appendix B Requirements and Verification Tables

Table 3. Requirements and verification table for the power supply unit

Requirements	Verification	Verification Status
1. Must supply at least 5.1V at 3A to the Raspberry Pi	Connect the power supply to an AC outlet and subsequently test the line of power supply to the Raspberry Pi using a digital multimeter	Yes
2. Must supply 12V at up to 3A to the linear actuator	Connect the power supply to an AC outlet and subsequently test the line of power supply to the linear actuator using a digital multimeter	Yes
3. Must supply 12V at 5.1A to the rest of the chopping assembly	Connect the power supply to an AC outlet and subsequently test the line of power supply to the chopping assembly using a digital multimeter	Yes
4. Must supply 5V at 2A to camera and 5.1V at 2.5A to display in the I/O unit	Connect the power supply to an AC outlet and subsequently test the line of power supply to the camera and display unit using a digital multimeter	Yes

Table 4. Requirements and verification table for the control unit

Requirements	Verification	Verification Status
1. Must be able to recognize the vegetable inserted	Insert a vegetable into the assembly and read a signal indicating which vegetable has been inserted	Yes
2. Send commands to control the motion of the blade for the selected chopping style	Select a chopping style and confirm selection. The finished chopping should match with specified chopping style	Yes

Table 5. Requirements and verification table for the I/O unit

Requirements	**Verification**	**Verification Status**
1. Must be able to send a visual feed, of the general region where the object to be chopped will be placed, to the Raspberry Pi	Connect the Raspberry Pi to a display and show visual feed from the camera	Yes
2. Must display the option to select cutting style and view recommended cutting style	Be able to view recommended cutting style and toggle between different cutting styles	Yes
3. Must allow users to confirm selected vegetable and cutting style to start	Push the confirm button after selecting a chopping style. The chopping assembly should start chopping in the selected style	Yes

Table 6. Requirements and verification table for the chopping assembly

Requirements	**Verification**	**Verification Status**
1. Must be able to exert a maximum of 300N of force on the vegetables [5]	Placing a weighing scale under the assembly without the blade to measure the force exerted	Yes
2. Must operate by drawing not more than 12V at 3A in total	Use a multimeter to measure the power drawn by the assembly	Yes
3. Must chop the vegetables within 4 minutes with 75% accuracy in dimension according to the chopping dimension standards set in French Cooking [4]	Measure the chopped vegetables to see if they are the right size and measure the time taken to complete the chopping process for all the chopping styles offered by our system	Yes