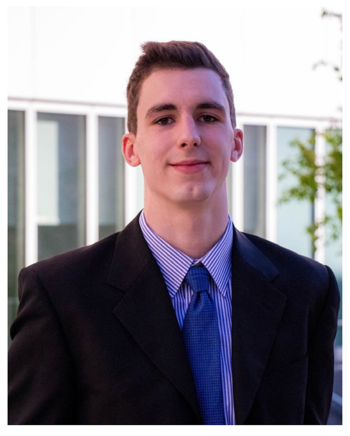
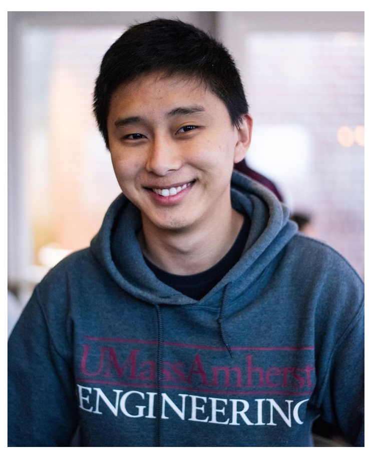



Intelligent Screw Organizer

ISO

Jordon Balskus, Jordan Gyaltsen, Andrew McGrath, Rajesh Shahi

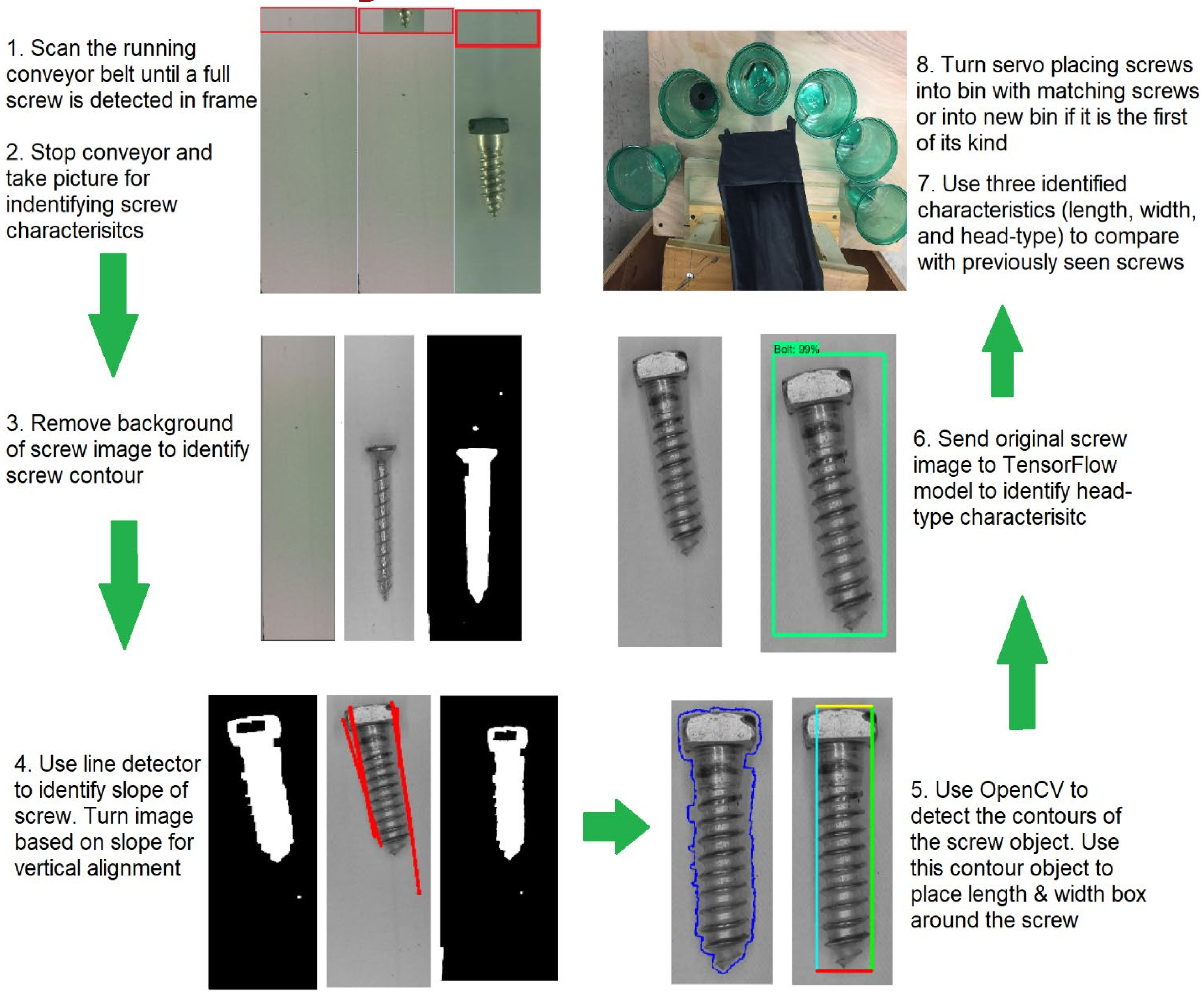
Faculty Advisor: Prof. Yadi Eslami


Abstract

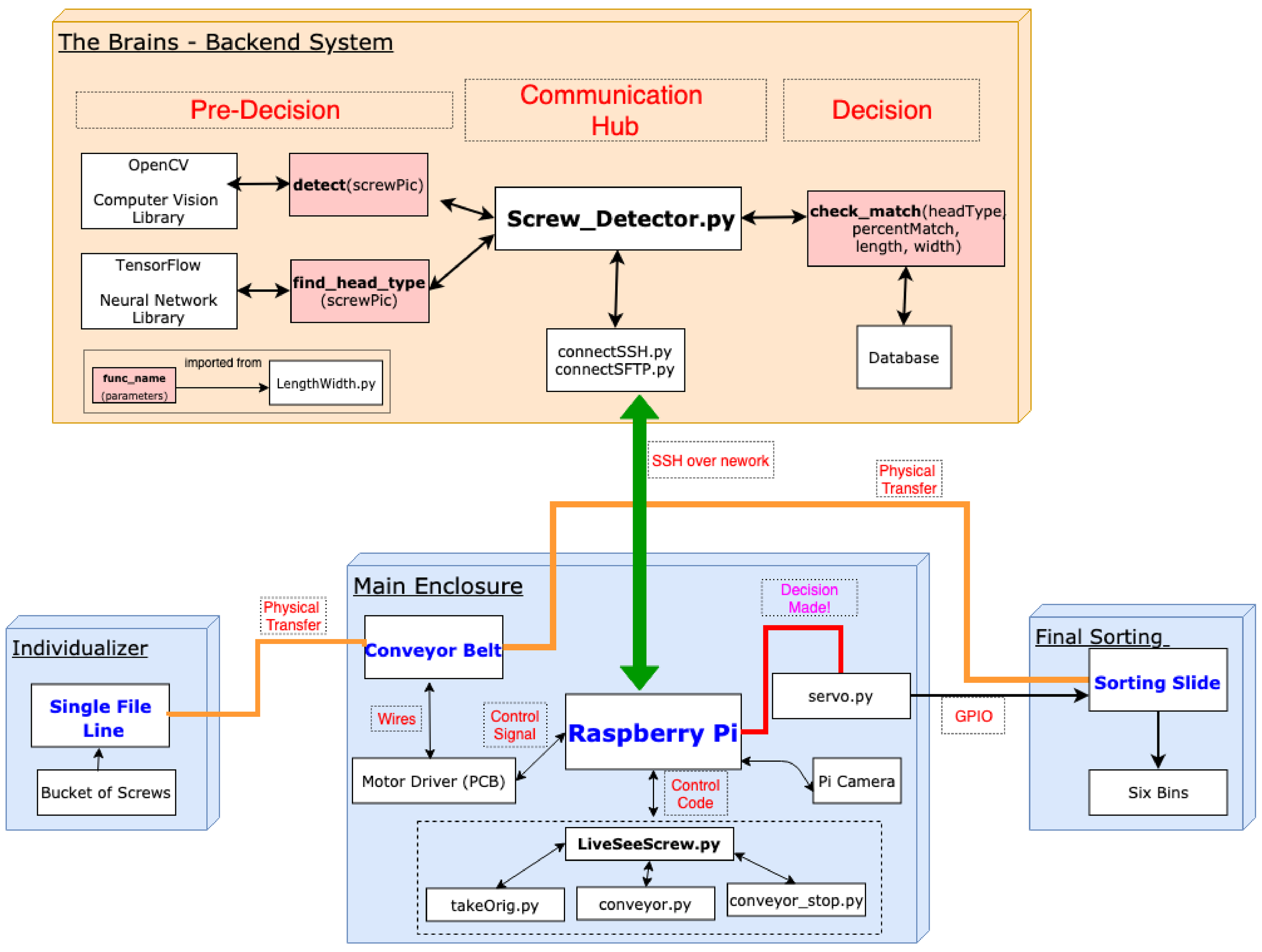
In every handyman's workroom, makerspace junk drawer, and machine shop corner, there is almost always a collection of miscellaneous, unsorted screws. To effectively repurpose this hardware, the pieces must be sorted. This can be laboriously completed by hand or sorted by a complex machine at an industrial budget. We introduce ISO - Intelligent Screw Organizer - to fill this consumer application gap by autonomously solving this problem at a price everyone can afford.

The magic behind ISO stems from its use of computer vision and machine learning. Screws are initially placed onto a conveyor belt where the system moves each piece through a camera enclosure. Upon screw detection, an image is taken and subsequently processed. The screw's length and width are optically computed, and the head-type is determined by our trained neural network. Both of these data points are factored into our custom classification algorithm, and this identity information is compared to the other screws seen in the batch. Once a decision is made, the screw falls off the conveyor belt and a sorting slide places it into a designated bin.

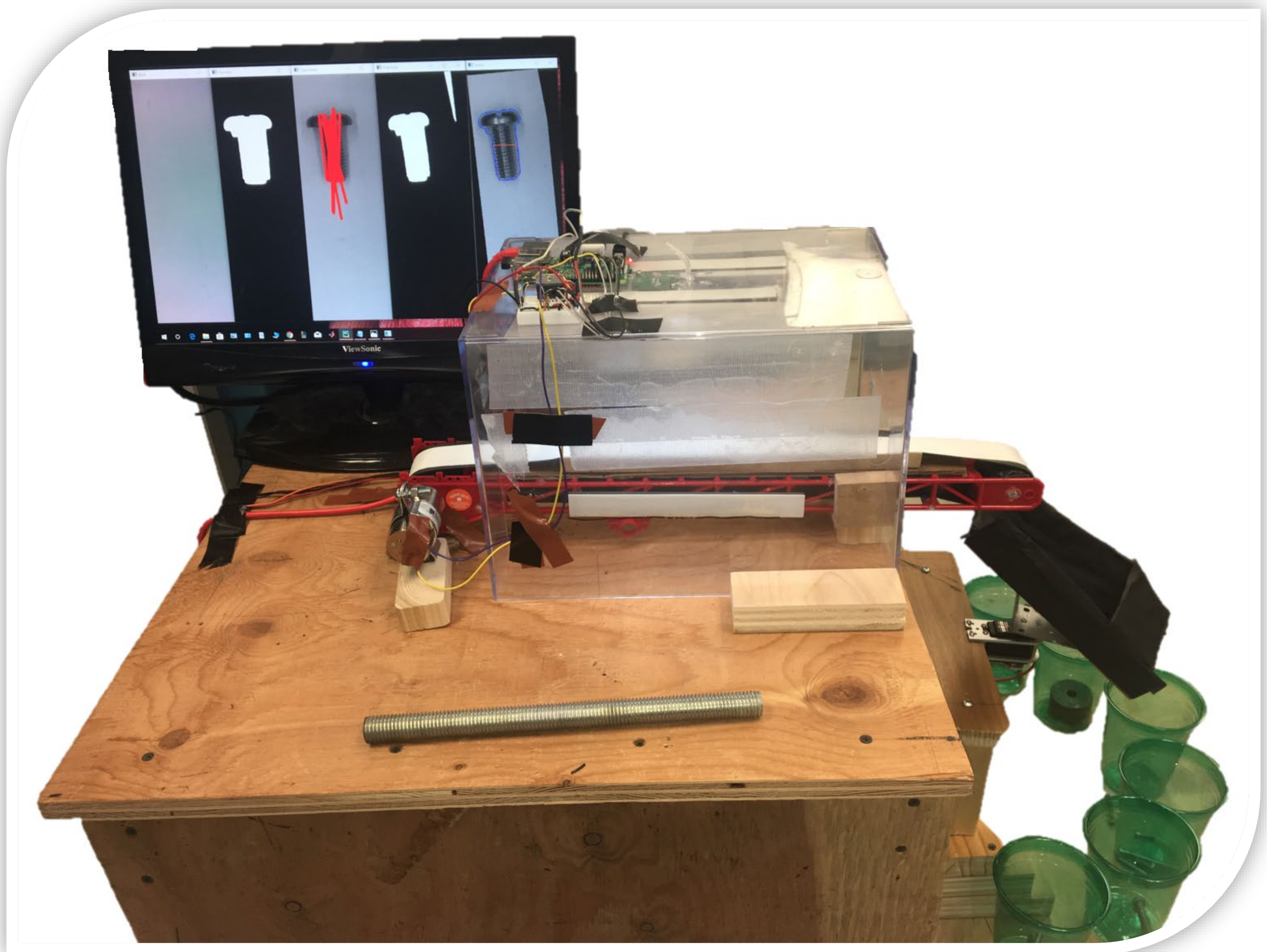
System Overview



Block Diagram



Results



•Successfully identifies screws with differences in length and width greater than 2 millimeters

•Successfully identifies head-types of screws within classification range. Some screws outside of this range are still identified.

•Successful identification rate of 84% with the majority of bad identifications separated from sorting batch.

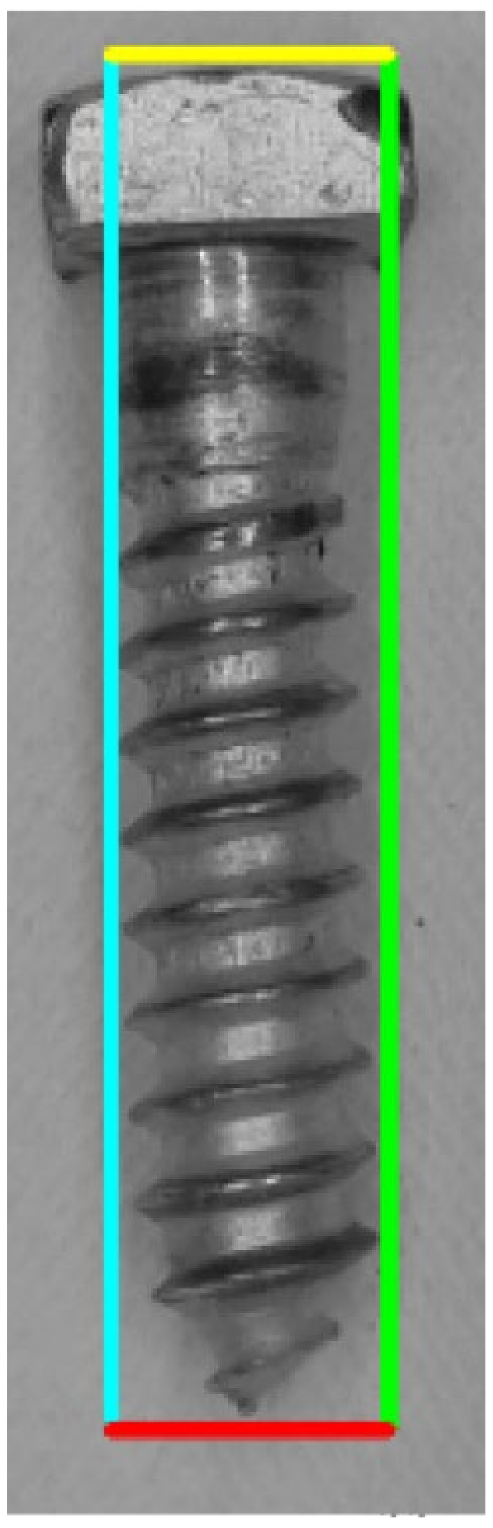
Specifications

Specification	Goal	Actual
Accuracy	90%	84%
Length Range (inches)	0.5" - 3"	0.5" - 2.25"
Width Range (inches)	0.1" - 0.25"	0.1" - 0.25"
Detection Time (sec)	< 6s	6.5s
# of Screw Types	5	5
Cost ($)	< 500	216
Automation Level	Fully Automated	Hand-Fed

Acknowledgements

The ISO team would like to thank our advisor, Dr. Yadi Eslami, for his continuous inspiration and thoughtful guidance throughout this entire project. We would like to thank Shira Epstein for kindly gifting ISO with the idea behind this project and for her insightful problem-solving advice. Finally, we would like to thank our evaluator Professor Xia, the ECE Department, and the University of Massachusetts Amherst.

Length / Width Computation



Left / Right

•Find the upper and lower y-axis bounds of the identified screw object

•Split the screw object into fifths and remove the first and last delineation. This contains the head portion of the screw.

•Move down the contour points averaging the x-positions until reaching the lower y-bound (yields the left side)

•Continue to run through contour points until looped above lower bound (now on right side)

•Right side x-average is now calculated with the remaining contour points

Bottom / Top

•Contour points start in the middle of the head

•Average the first half of the top until reaching left x-bound

•Loop back above left x-bound (now at the tail)

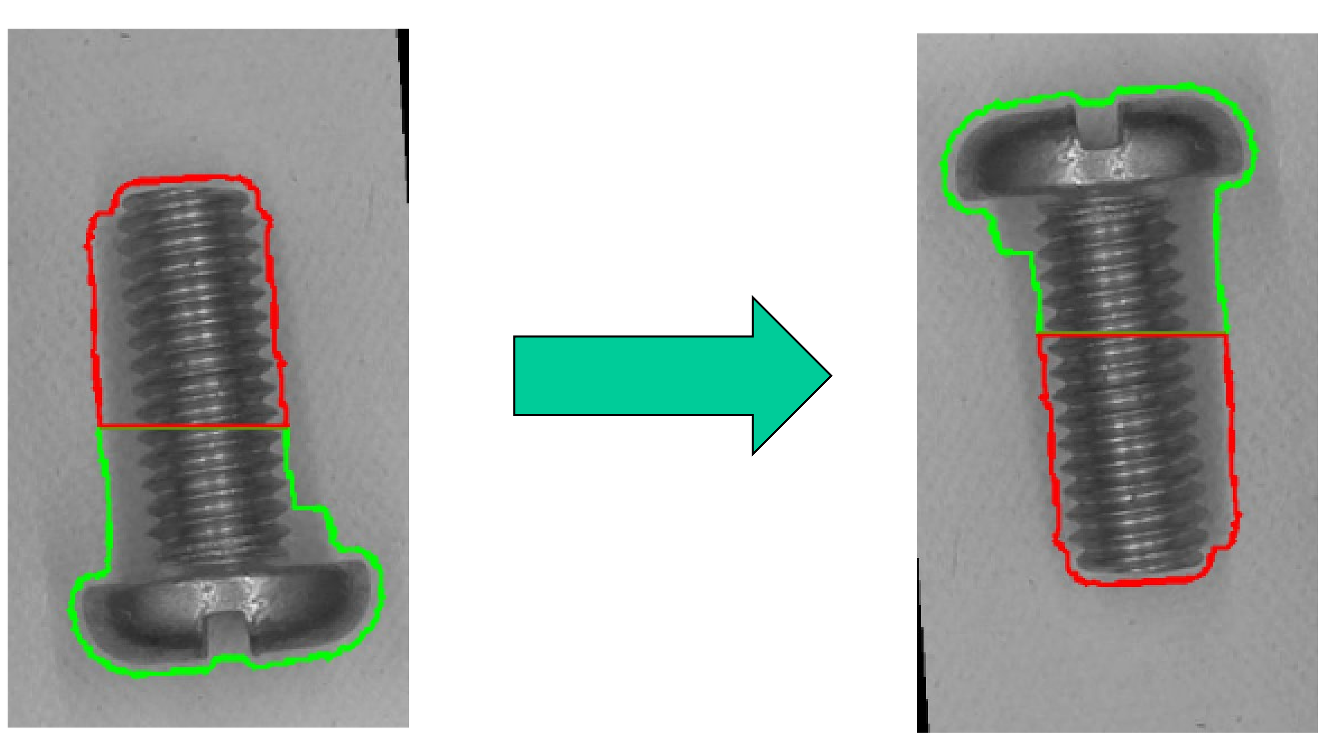
•Calculate bottom average until right bound is reached

•Wait until we are back below the right bound (now at second half of top)

•Calculate the top average by combining the two halves from the beginning and end

Upside-Down Detection

- **Split the screw image into top and bottom half based off upper and lower length average**
- **Using these new split contours, calculate the area of both ends of the screws.**
- **In our scope, the head of the screw will always have an area greater than the tail. Therefore, if the bottom half of the image has a greater area, the screw is upside-down.**
- **Re-run the identification algorithm with the flipped screw**



Conveyor Belt

•Structural component made from a Bruder conveyor belt toy.

•Removed stock belt and utilized an inner cycle tube lined with tape as it was more elastic, flexible, and applicable for screws

•Triangular side-supports create a dip in belt to maintain degree of verticality in order to aid in the orientation of skewed skews

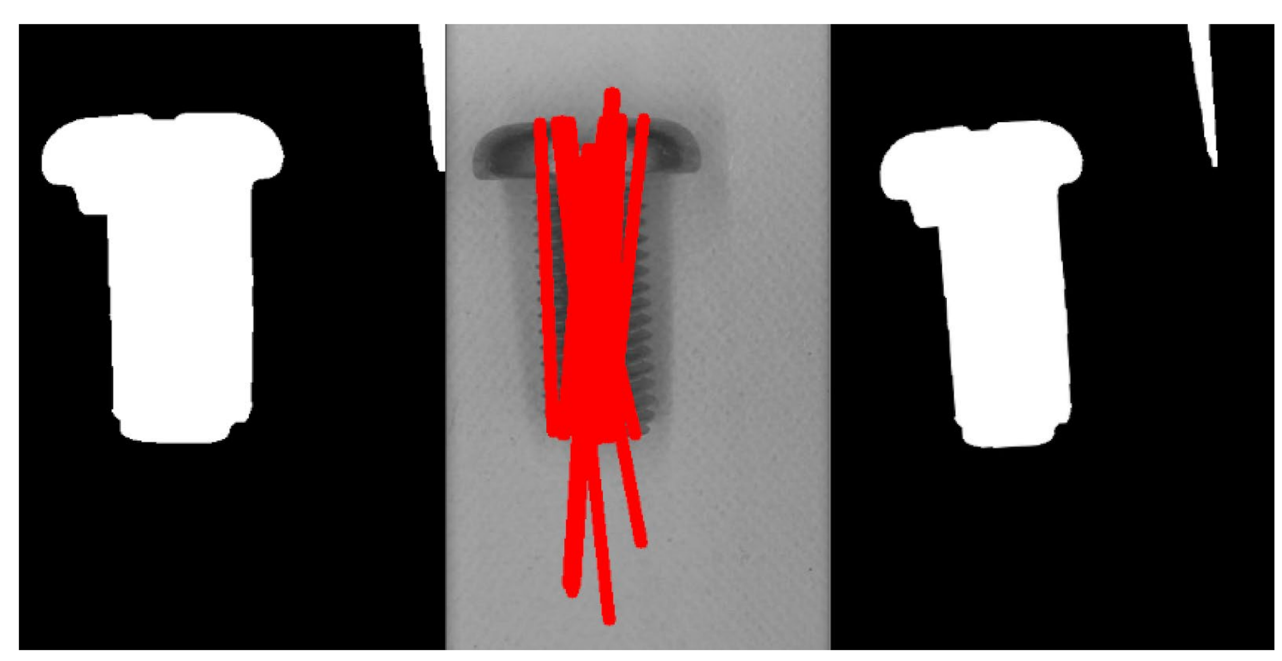
•Efficient design results in minimal power expenditure, durable usage, and a movement channel perfected for screw detection

Cost

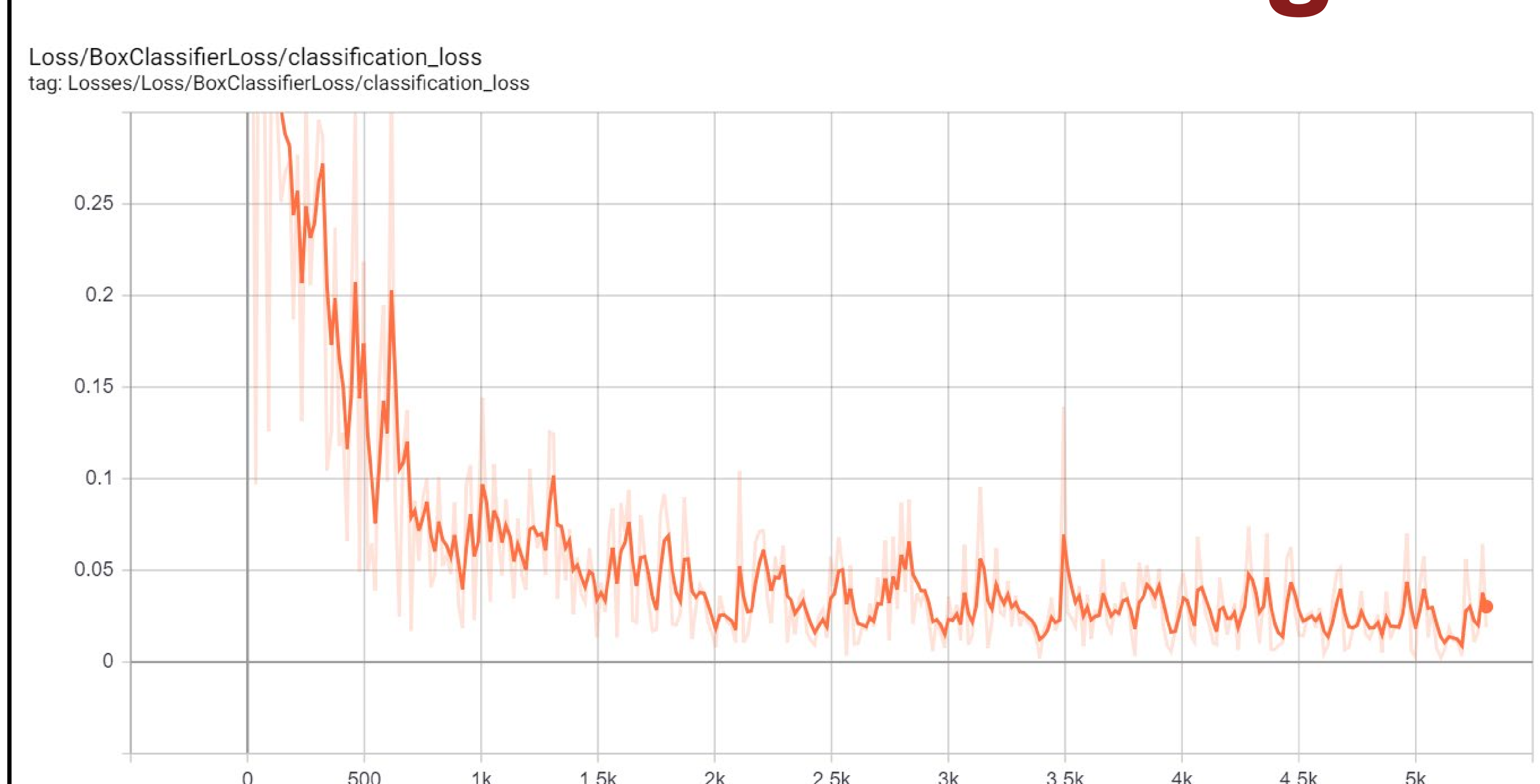
Part	Development	Production
Conveyor Belt	$45	$25
Raspberry Pi	$25	$25
ArduCam 5	$30	$30
PCB	$4	$1
PCB Components	$5.84	$4.07
Diodes	$0.20	$.05
12V DC Gear Motor	$20	$15
LED Lights	$26.98	$5
Enclosure	$24.99	$13
Wood Stand	$14	$14
Batteries for LEDs	~$20	$7
Total	$216	$139

Line Detector

- **Utilizing the Hough line detection function in OpenCV, 25 lines are drawn on the screw and they are oriented in the direction it is currently facing**
- **Run through each of these lines and find the slope**
- **Calculate the combined slope of these lines to find the avg_slope of the screw**
- **Use this slope to calculate the angle of degree away from full vertical using the arctan function**
- **Turn the image to vertically align the screw based on this calculated angle**



Machine Learning



Above graph depicts the classification loss over time of our neural network while training.

As an additional classification factor, we utilized the open-source machine learning library, TensorFlow, to create and train a neural network model that can detect screw head-types.

The network can classify four different types of screws: wood, machine, cap, and bolts. After an image is taken, this data is fed into the network and a percentage of similarity is computed. This percentage is then factored into our comprehensive decision algorithm with a custom weight.