Robotic Tactile Sensor

Von Simmons, Electrical Engineering Major

Thesis

Humans and robots differ in many ways: in our physical abilities, in our reaction times, and in our capacity to decide which course of action is the most appropriate at any time. But one of the most important differences between the two groups is the ability to perceive the world through various sensors. While robots may use hi-tech cameras, ultrasonic sensors, and barometric pressure gauges, they still lack the ability to touch the world as we do.

Skin is our largest organ which covers our entire body. It simultaneously senses humidity, temperature, and pressure; It allows us to understand the texture and relative shape of objects without seeing them; And it also gives us the ability to know when our body is in contact with itself - at joints, for example. This sensory organ is so central to our lives, yet it is easy to forget about.

Purpose

Motivation

Based on the research by Katherine Kuchenbecker and Oliver Brock I would like to create my own version of a tactile sensor using my knowledge of Computer Vision (CV) and Machine Learning (ML). By conducting this research I hope to learn more about the differences between humans and robots. Through this I can gain insight on how to augment human ability with the assistance of intelligent systems.

Usage

Robots which are able to feel are able to cooperate with humans more safely. More perceptive machines can prevent injury in warehouses, on the road, or in other situations requiring both parties. Furthermore, extra senses allow for more robust interactions with the surrounding world. In situations where one sensor is limited or obscured, the sense of touch has potential to make robot navigation and manipulation more reliable.

Objective

Here, I aim to detect deformations on an artificial “finger” using a camera then visualize an approximation of the depression virtually. The concept relies on a ML stack where information is passed from one layer to another, generating predictions of the state of the finger along the way. At the end, a geometric transformation of predicted force vectors on the finger will create level curves of the approximated deformation shape in R^3. Each ML layer will be trained using various experimental techniques. Due to transfer learning I will not be required to collect a large amount of data, instead I will be able to "tune" existing datasets to predict the required values.

Computer Vision (YOLO)

In the top layer, video data (frame-by-frame) will be fed into the YOLO (You Only Look Once) ML algorithm. This is one of the fastest existing methods for object recognition and will allow for detection of depressions on the inside of the finger. I will build a custom program for labeling training data (photos) with a bounding box, center of the depression, and depth of the depression. Supervised training should then enable automatic tagging of depressions with the bounding boxes and centers, but not the depth.

Tabular ML (Pandas, PyTorch)

From there, a CSV file will be created by parsing the data from the training data then fed into a second ML layer. Given supervised depth values in the training data, the tabular algorithm will give predictions on the depth.

Physical Experimentation

The TPU finger has been printed using an Ender3 Pro and Autodesk Inventor. Video from the iPhone 14 will be used to collect high definition footage. All depression depths will be measured with calipers.

Figure 1a: TPU Artificial Finger

Figure 1b: Sample Depression (Top)

Intended Outcomes/Expected Results

I intend to visualize a single point-depression (1) as a 2D heatmap and (2) as a 3D digital version of the cone and depression. The visualization should include the location and depth of the depression. For the sake of simplicity, I will constrain depressions to be toward the center of mass of the cone.

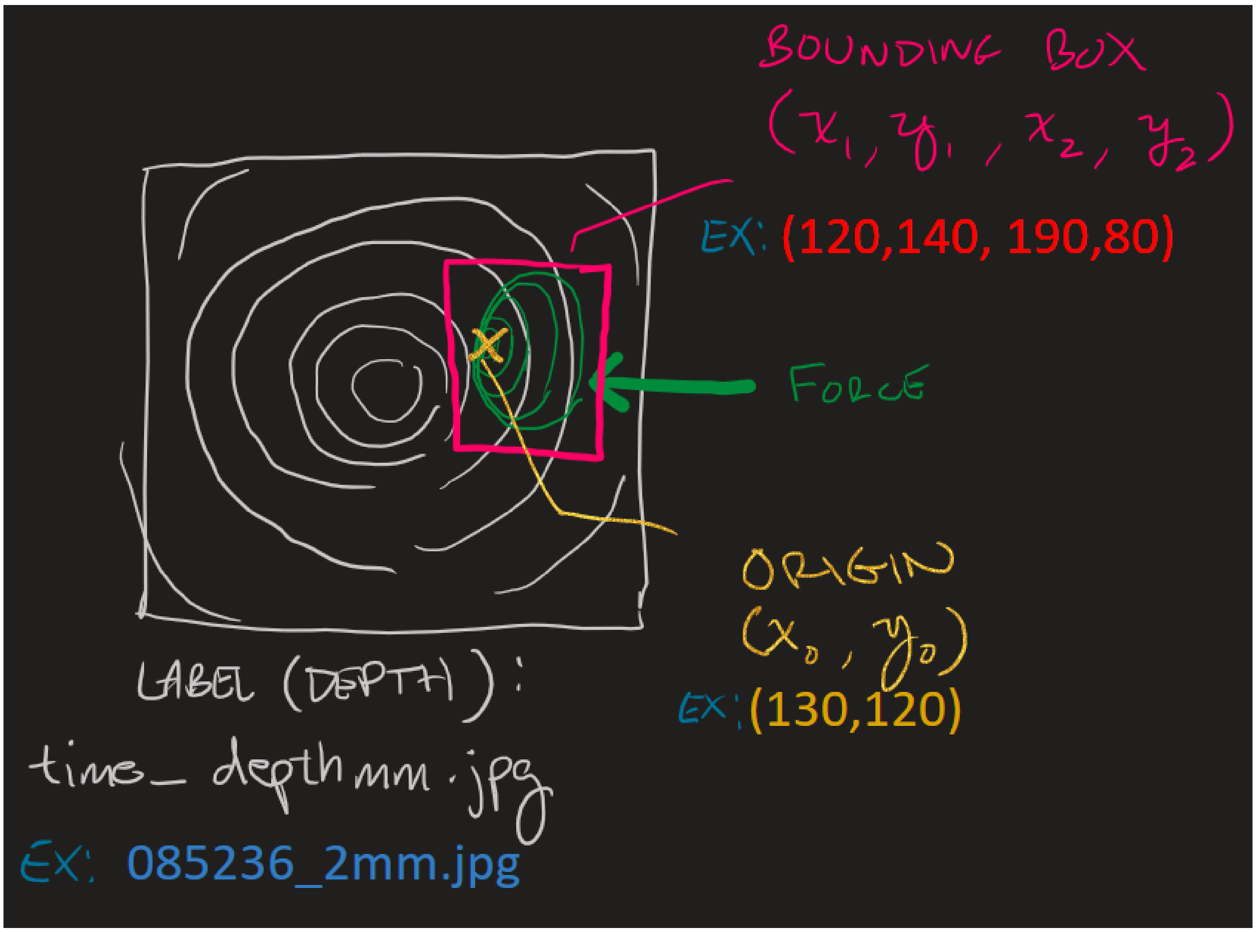


Figure 2: Anticipated supervised training data sample.

Timeline

Development of this project is divided into three phases: Learning and Training, Data Visualization, and End-to-End. Phases have been identified by major breakpoints in the project. At each breakpoint, the project will reach a steady state where everything before it works and can demonstrate some functionality.

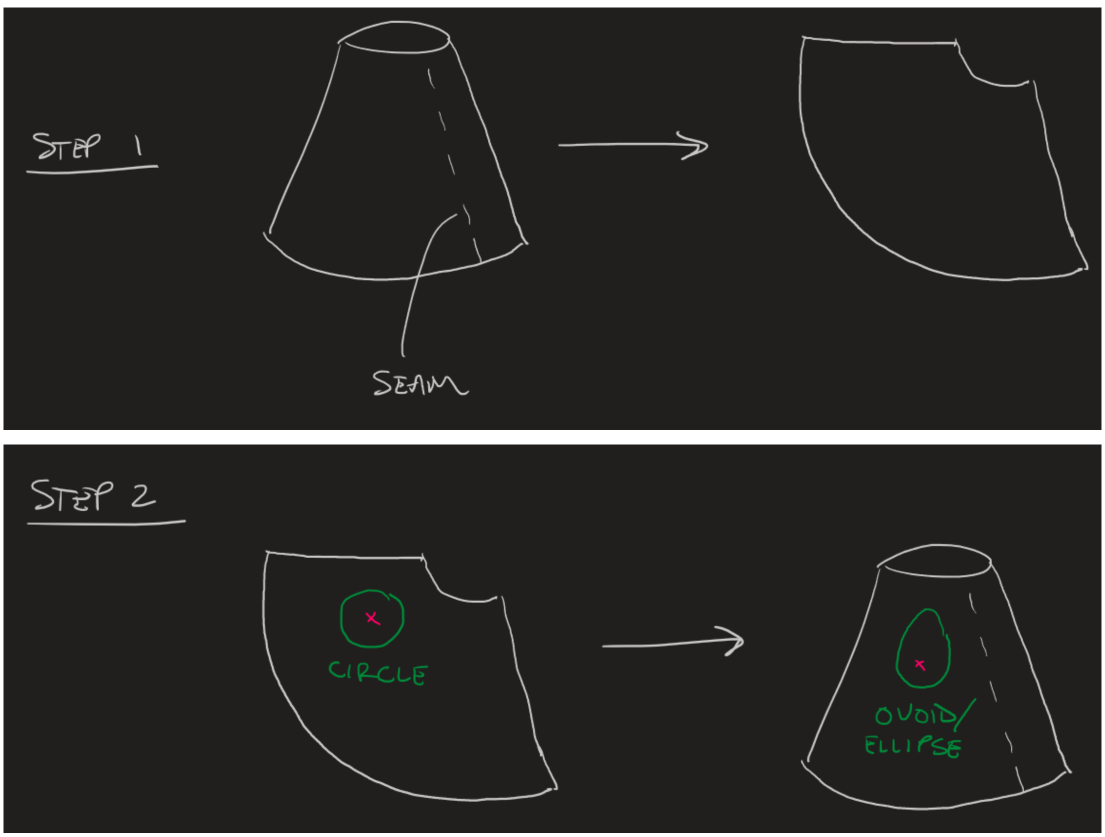
Learning and Training

The Learning and Training phase is dedicated to, firstly, learning how to use the selected ML algorithms. Deep knowledge of these systems is not required for effective use, which is the main advantage. The goal of this project is not to develop new technology, but to utilize existing techniques to create a mechanism for future work. After learning how to use ML algorithms in an example case, I can then modify these tutorials to be trained on relevant data. I will collect, clean, and label data with a custom tool I build. At the end of this phase, there should be two separate, yet working ML algorithms which output the desired kind of prediction.

Data Visualization

The Data Visualization phase is for developing the mathematics algorithm and code which converts the anticipated predictions from the preceding portion of the stack into digital representations. The heat map shows depressions as hot spots and unperturbed zones as cold. The idea here is to mathematically unroll the finger (which is a cone), plot the disturbance on the flat manifold, use an approximation of the disturbance shape, extrude it to three dimensions, then reroll the cone. This can also be used to show level curves of the depression in 3D or track changes in the depression over time.

Mathematics: Cone Unwrapping Concept



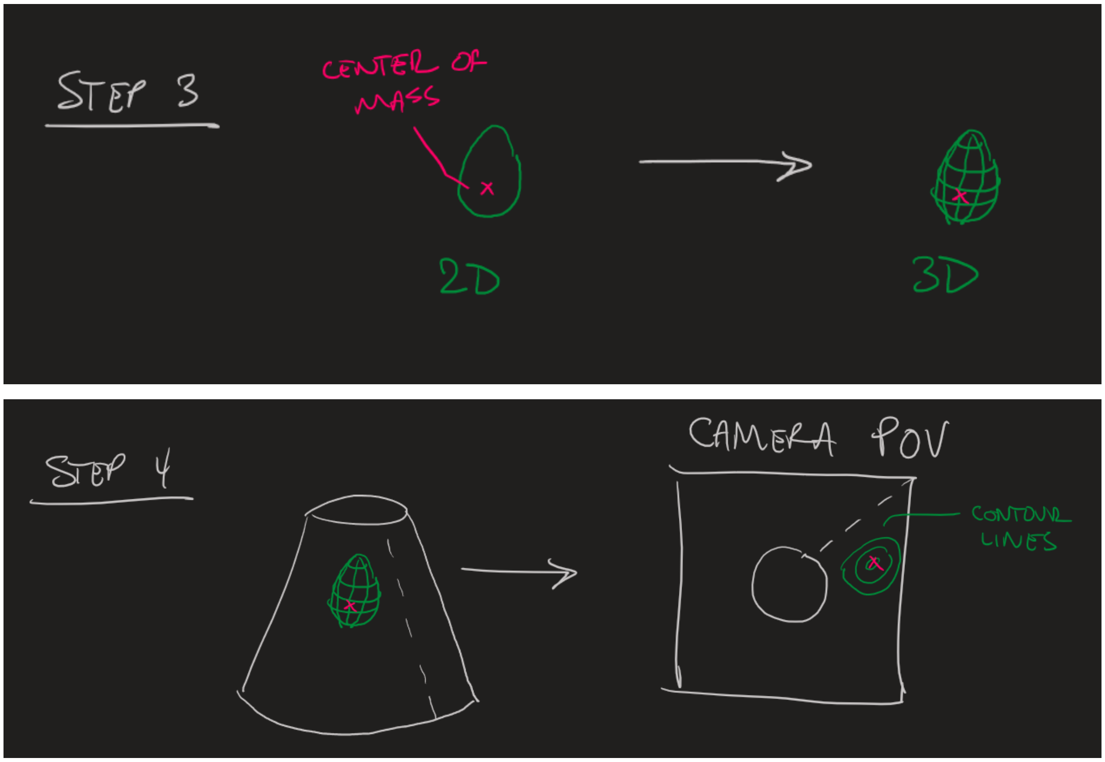


Figure 3: Rough conceptual plan for placing approximate depression on abstract cone.

End-to-End

With two ML algorithms and Data Visualization complete, the End-to-End phase is meant to connect all components of the stack together. Here, I will programmatically align each module, then run a portion of my training data through the full stack. If the system works as expected, then an unlabeled picture of a depression should be processed and visualized by the Data Visualizer. From here, I can begin testing on real data - first as images, then as full video. Ultimately, I hope to track a depression moving across the cone in real-time and see it visualized by the stack.

Itemized Budget

All needed items are already in possession. This project is for recognition only and does not require funding.

References

Example UROP proposal:

- https://cpb-us-w2.wpmucdn.com/wp.ovptl.uci.edu/dist/e/3/files/2022/10/Engr-1.pdf

Those two conference talks and their related papers

- MIT Robotics - Katherine Kuchenbecker - Tactile Sensing for Robots with Haptic Intelligence
    - https://youtu.be/0Vg0jkzVaFw?t=1283
    - https://arxiv.org/abs/1511.06065
- MIT Robotics - Oliver Brock - Why I Believe That AI-Robotics is Stuck
    - https://youtu.be/tr6aatJL84A?t=1440

FastAI fastbook and any related papers

- https://github.com/fastai/fastbook
- https://docs.fast.ai/tutorial.tabular.html
- https://arxiv.org/pdf/1604.06737.pdf
- https://arxiv.org/pdf/1606.07792.pdf

YOLO description, tutorial and the original paper

- https://medium.com/analytics-vidhya/yolo-explained-5b6f4564f31
- https://www.v7labs.com/blog/yolo-object-detection
- https://towardsdatascience.com/yolo-v5-object-detection-tutorial-2e607b9013ef
- https://arxiv.org/pdf/2209.02976.pdf