Teaching an Assistive Robotic Arm to Recognize and Respond to Human-Intent

Austin Salcedo, Ian Abraham

Yale University

Howard Hughes Medical Institute

ABSTRACT

Teaching robots to understand human intent remains a challenge. The goal of this project is for a robotic arm to autonomously assist a human in completing tasks by interpreting and responding to the human's intent. Here, we have made the first step toward this goal by **developing an interface for human-robot intent prediction** by programming the robotic arm to track a human's face via OpenCV's face-detection algorithm and visual servoing techniques. This is important for future human-robot interaction (HRI) research as it will allow the robot to keep the human in sight as it continuously collects data. **Future steps for this project involve training the robot to interpret human's intent** (e.g., via emotion, gaze, or body-pose recognition) and to respond with appropriate, assistive responses.

MOTIVATION

Robotic arms have been assisting humans for decades, yet only when pre-programmed for a specific task. Today, we implement machine/deep learning methods to teach robots how to autonomously help humans on a more personalized level. Seen below, researchers taught a robotic arm to interpret a paralyzed woman's intent via neural signals by using a brain-computer interface implant.

Similarly, this project aims to teach a robotic arm to non-invasively interpret a human's intent and assist them.

Figure 1: Brain-Computer Interface.

Paralyzed from a stroke for over 15 years, a woman controls a robotic arm via a brain-implant.

Source: https://news.brown.edu/articles/2012/05/braingate2

MATERIALS

- Panda Robotic Arm by Franka Emika
- Cybertrack H5 1080P Webcam by Adesso
- Robotic Operating System (ROS)
- OpenCV Face-Detection Algorithm
- Visual Studio (VS) Code (using Python)

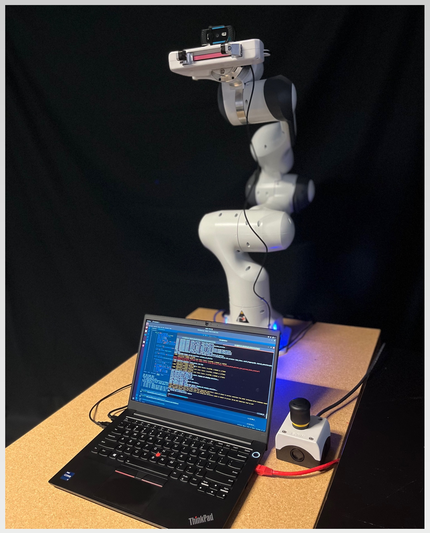

Figure 2: Project Set-up. A Franka Emika robotic arm with webcam mounted at the end-effector (gripper). The webcam is connected via USB to the laptop, and the laptop is connected back to the arm via an ethernet cable.

APPROACH

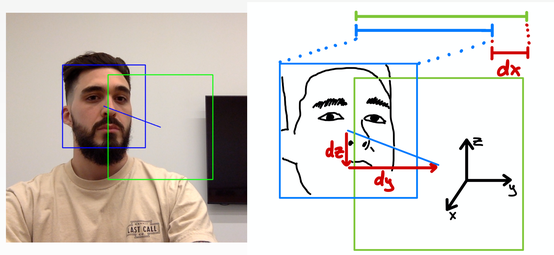

Figure 3: From Pixels to Instructions. OpenCV's algorithm draws a rectangle around detected faces. The line drawn from the face's center to the image frame's center is converted from distance (in pixels) to velocity controls in the y-/z-direction. Similarly, the size difference between the face's width (blue rectangle) and the desired width (green rectangle) is converted to velocity controls in the x-direction.

APPROACH (cont.)

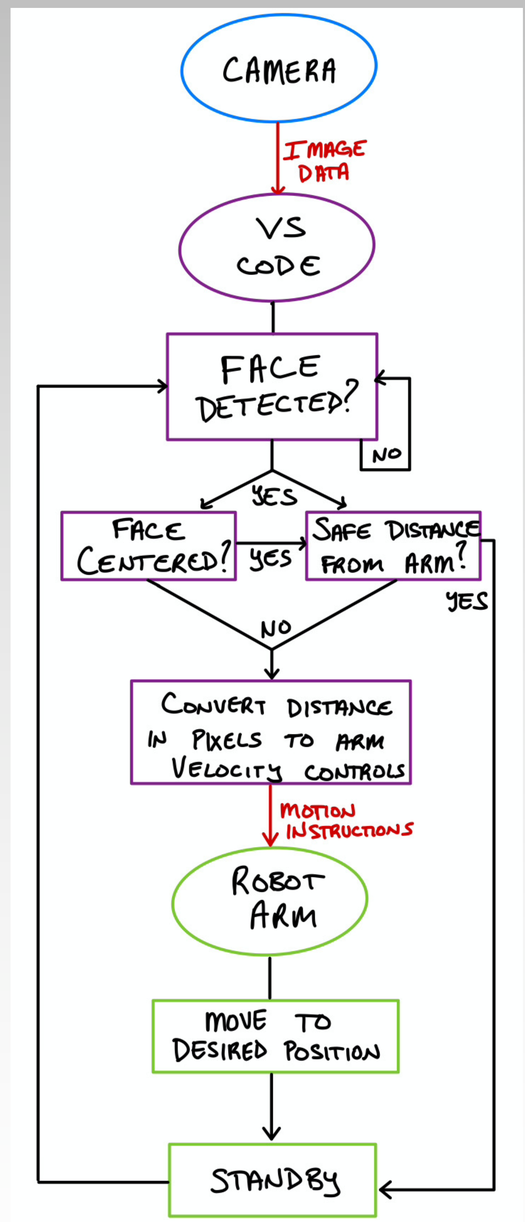

Figure 4: Pseudocode Flow Chart. A behind-the-scenes look at the data flow process between the camera, code, and robot arm nodes. The camera continuously passes image data to VS Code for image processing. Then, only if needed, VS code sends motion instructions to the arm. Data passed between nodes are represented by red arrows.

FUTURE DIRECTIONS

- Currently, the arm only moves the camera's position-in-space, but its orientation is fixed. Next, the arm must **learn to keep faces oriented directly at the camera lens**.
- Variations of human-intent recognition may involve **teaching the robot to respond to emotions or body-pose**.
- A future collaboration with the Yale Interactive Machines Group may use this interface for a **human-robot interaction study looking at human gaze patterns.**

TAKEAWAYS

- Some of **the simplest tasks for humans** (e.g., recognizing another face) **are incredibly difficult for robots** to learn.
- Given the intrinsic complexity of humans, **teaching the robot to interpret human intent** will be **the greatest challenge**.
- Assistive robots such as these may offer a **quicker, easier, non-invasive alternative** to brain-computer interface implants (as seen in the *Motivation* section).

REFERENCES

- Microsoft. VS Code. *https://code.visualstudio.com/*. Last accessed 29 July 2022.
- OpenCV. (2015). *Open Source Computer Vision Library*.
- Stanford Artificial Intelligence Laboratory et al. (2018). *Robotic Operating System*. Retrieved from https://www.ros.org/

ACKNOWLEDGEMENTS

I thank my PI Ian Abraham and my labmates for their support throughout this project, as well as my funders: the REVU Program and HHMI. I would especially like to thank Marla Geha, Jeremy Bradford, Teena Griggs, and my incredible cohort for their unwavering support both in and outside of my participation in REVU.