UNIVERSITY OF MINNESOTA
Driven to Discover℠

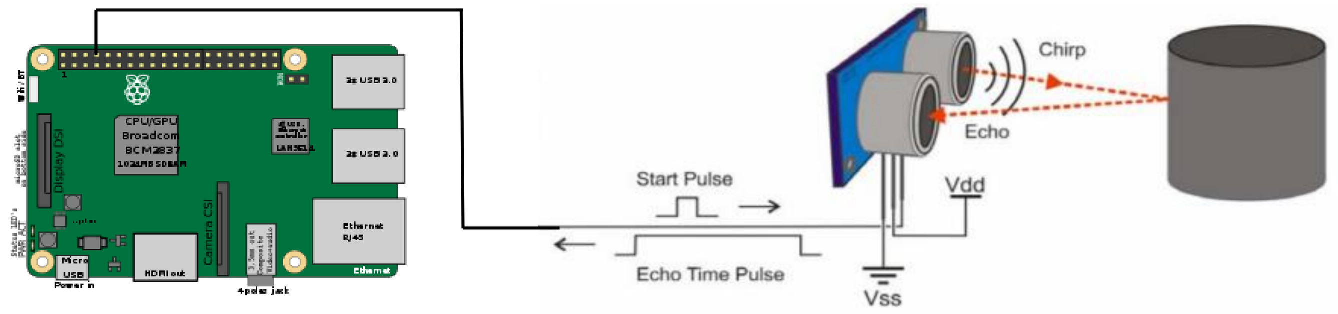
Facial-Recognition, Obstacle-Avoiding, Omni-Directional Robot Platform

Group Members: Caden DeRoche, James Kolar, Noble Koshy, Bryan Nestingen, and Yao Yao
Advisors: Dr. Chris Kim and Luke Everson

PURPOSE: This project implements obstacle-avoidance hardware and tracking software on an existing omni-directional robot platform. This project aims to create a fully-autonomous robot to serve as a testing platform for facial-recognition software. Our solution brings added-value to the robot platform by adding obstacle-avoidance, absolute-orientation, and facial-recognition capabilities. Distance and measurement data are extracted from the sensors to plot obstacle and terminal locations. The robot demonstrates its facial-recognition capabilities by being able to locate and plot a safe course to a targeted individual.

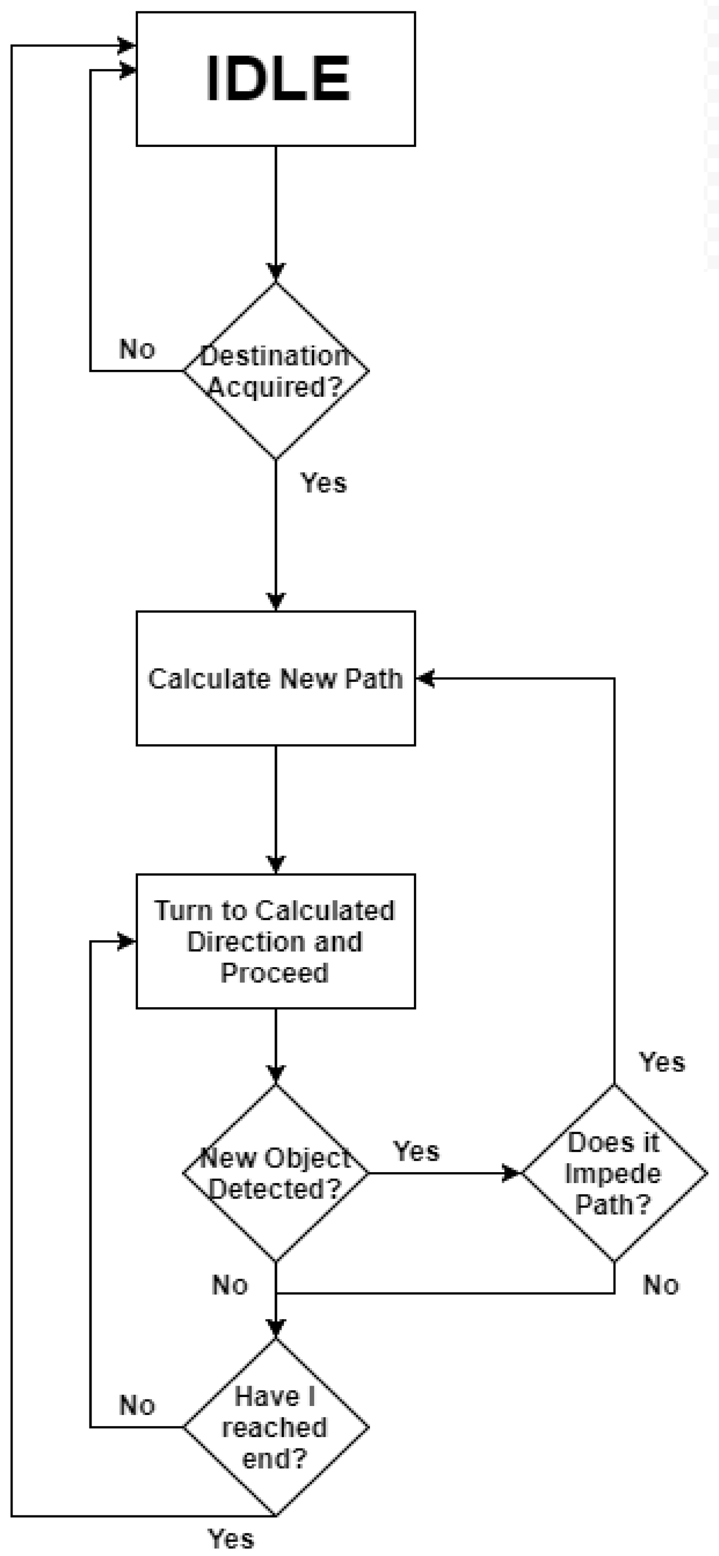
ULTRASONIC DISTANCE SENSORS

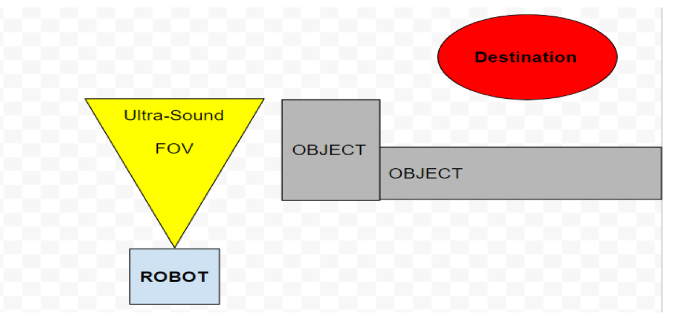
These sensors work by "pinging" the environment with an ultra-sound wave (approximately 40kHz). When the response is detected, we can determine how far the nearest object in a frontal arc is by measuring the time latency and dividing by the speed of sound. We have implemented three ultrasound sensors and get accurate measurements for objects directly in front of the sensor up to a range of 1 meter.

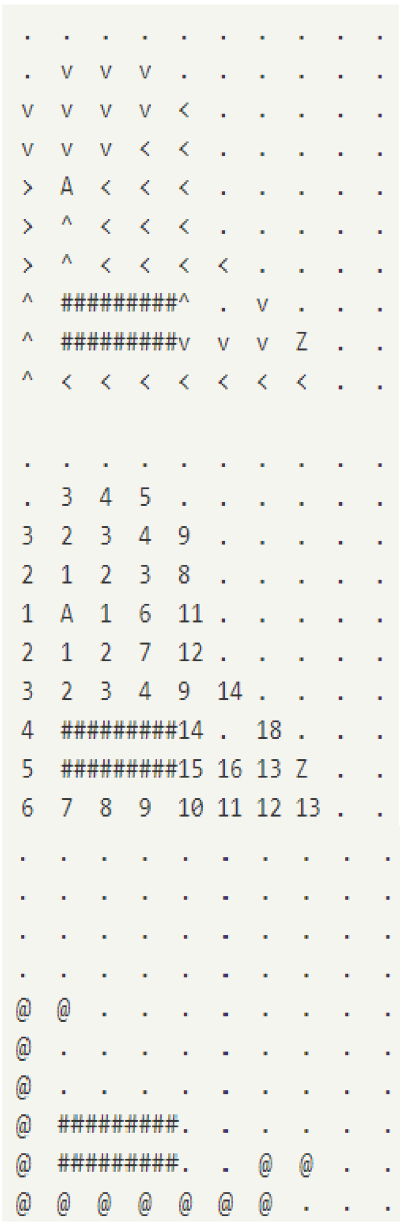


A* PATHING ALGORITHM

The current navigation protocol starts by searching through a simple graph using an A* search algorithm. The goal of this is to discover an optimal route to the destination in as little time as possible. Once this route is created, the robot executes the route movement by movement. It checks for obstacles at every point using the front mounted ultrasonic sensors and adds these objects into memory. If an obstacle interferes with the current route, it recalculates the path and starts again. This protocol allows for the robot to always find a path to the destination if one exists.



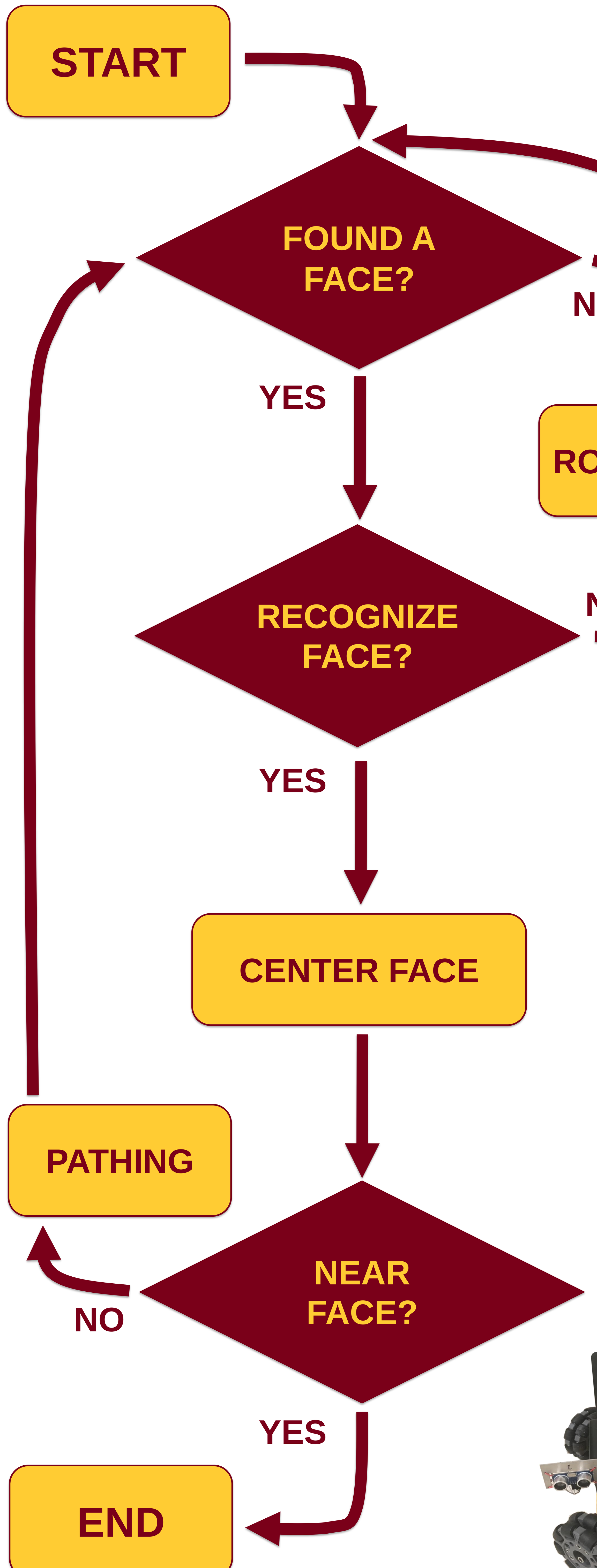




# Denotes Obstacles in Grid

Numbers Denote Evaluated Heuristic Values

Actual Path To Reach Destination
A-Destination
Z-Start

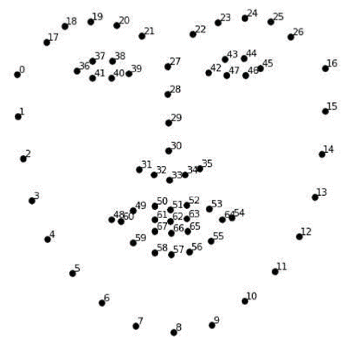


FACIAL RECOGNITION SOFTWARE

Dlib's face recognition model is accessed using a Python API. It is a pre-trained model capable of face recognition, better than facial detection software such as the Haar-Cascade method.

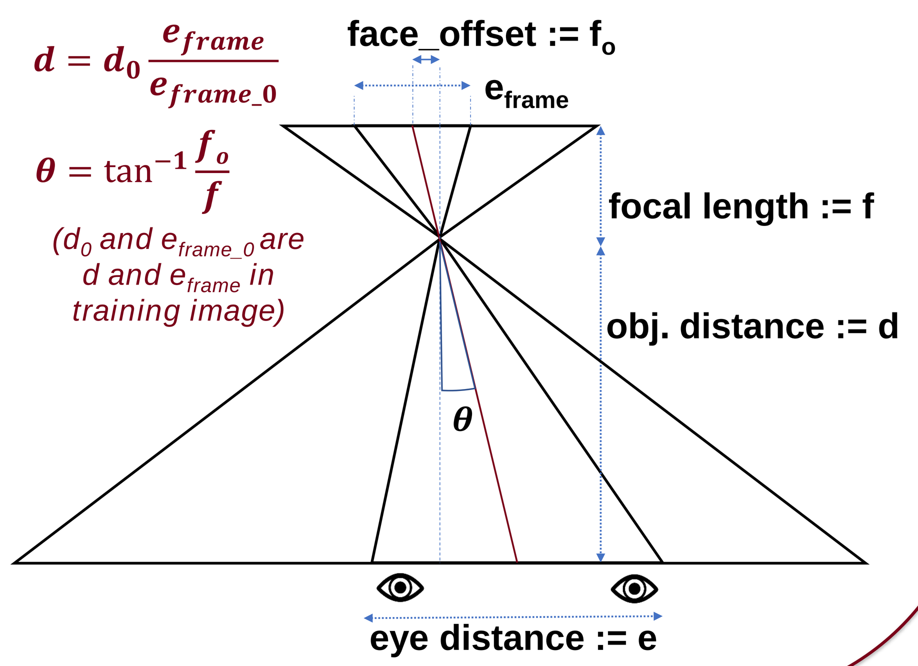
Summary of steps for facial recognition:

1. Detect/identify faces in an image
2. Predict face poses/landmarks (for the faces identified in step 1)
3. Using data from step 2 and the actual image, calculate face encodings (numbers that describe the face)
4. Compare the face encodings of known faces with those from test images to tell who is in the picture



Facial Encoding: 68 facial landmarks

Position Calculation



NEXUS ROBOT OMNI-WHEEL DRIVE TRAIN & ADAFRUIT INERTIAL MEASUREMENT UNIT (IMU)

The Raspberry Pi serially communicates movement commands to the Arduino. The Arduino's powers and controls the omni-wheel motors. Our robot uses only forward movements and 90° rotations because of sensor placements. Forward movements resulted in a tracking error of 3 cm displacement over 30 cm of distance traveled. After testing, the error is due to limited drive train capabilities by the Arduino which initiates wheel commands sequentially. Wheel 3 demonstrates steady-state speed oscillations thereby needing PID tuning or motor replacement. We added a Adafruit BNO005 IMU to correct for drive train error and calibrated the robot to have ±1° accurate rotational movements by utilizing the IMU.