CPE416
Fall 2023 - Seng
Lab 3

Due Date

Friday, 10/27/23

Objectives

- To implement a neural network-based line following algorithm

Back Propagation References:

- Back Propagation (Wikipedia)

Part 1

First implement a line following function using only your proportional controller from the previous lab. This function should be called `compute_proportional()` and should have the following prototype:

```
struct motor_command
compute_proportional(uint8_t
left, uint8_t right);
```

The function should take 2 uint8_t values as parameters (these will represent 2 sensor readings). The function should return a `struct motor_command` containing the speeds to assign to left and right motors. Verify that your proportional controller can run the track smoothly and without any jittery motion.

Part 2

(or reset), it should display the word ‘**Proportional**‘ and immediately begin running the proportional controller

- When the on-board button is pressed:
    - The proportional controller should stop
    - The program should enter a data capture mode and the first line of the display should show the word ‘**Data**’ followed by a counter value (the counter value will show the number of readings captured)
    - The second line of the screen should print the left and right sensor readings
    - At this time, the robot should be capturing the left/right sensor pair readings and storing them for neural network training
    - During this data capture mode, you should move your robot side to side over the line
    - When the button is pressed a second time, the program will enter Training mode
- Training mode
    - During this mode, you will use the captured training data along with the proportional controller to train the neural network
    - The first line should display **‘Training’**
    - **In this mode, you should have a mechanism to allow the user to control how many training**

        network that cannot follow a line
    - After training is complete, when the button is pressed, the robot should enter a neural network line following mode.
- Neural network line following
    - In this mode, the robot should use the neural network to do line following
    - If the button is pressed, the robot should go back to **`Training`** mode

---

Implementation Notes

Your code should contain the following functions:

- `compute_proportional()` - this is from part 1
- `compute_neural_network()` - takes 2 `uint8_t` sensor values and returns a struct containing 2 motor values
- `train_neural_network()` - trains the neural network using actual values returned from the sensors (stored in an array from the data capture mode)

For your neural network, use the following parameters:

- Use 3 hidden layer neurons and 2 output layer neurons.
- Experiment with the learning rate .001 - .05. A higher learning rate makes the network train with fewer iterations, but increases the

obtain a usable motor speed.
- Initialize the weights and bias to random numbers between 0 and 1 using `rand() / RAND_MAX` (include `<stdlib.h>`)
- Inside `compute_neural_network`(), scale the sensor readings to values between 0 and 1
- Do not forget to adjust the bias in the same way the weights are adjusted

Demo

You will demo the following:
- Part 2