Simon Fraser University
School of Mechatronic Systems Engineering
MSE110 Mechatronic Design I

Project – Barcode Reader

Overview

Bar codes were originally invented to keep track of railway cars. Since the invention of bar codes, they have been extensively used in many industries such as asset and inventory tracking. In this project, you will be building a robot that can scan a bar code (standard code 39) of a single ASCII alphabet character (A, B, …, X, Y, Z).
The bar code will be printed by each team on a letter size page 8.5”x11”.

Demonstration marking Criteria

1. Scanning Time: faster scanning speed is better **(20%)**
2. Barcode size: smaller bar code size (thinner bars) is better **(20%)**
3. Mean Time Between Failures (MTBF): the longer MTBF the better. The team is required to print different characters and different sizes to show and report MTBF during the demo. **(20%)**
4. Mechanical design: portable, rugged, accurate and reliable is better **(20%)**
5. Software design: well commented, readable and organized code is better **(20%)**

Project Details:

Phase I MATLAB Simulation

Design a MATLAB script to:

1. Generate a noisy barcode code39 capitalized ASCII character image file (jpg). Sample noisy barcode files and scripts is uploaded to Canvas.
2. Read a single row of pixels spreading across the circumference of a circle across the barcode to simulate the light sensor input. This will later be replaced with actual light sensor values from the EV3 robot.
3. Calculate the ASCII character corresponding to the simulated barcode image. This can be achieved through the steps taught in class namely:
    a. Filtering using the moving average algorithm. The algorithm should be implemented from scratch by the team.
    b. Calculating the bar edges using discrete difference
    c. Calculating widths and colors of each bar in the barcode
    d. Building a lookup table (or equation) to find the corresponding ASCII character
4. Students are ***required*** to fully understand, organize and comment MATLAB scripts provided on Canvas.

Phase II Physical implementation of the barcode reader

Hardware requirements:

a. As discussed in class, the exact design idea is left to the team's judgment (e.g. moving robot, rolling the paper into the robot, scanning robot arm, scanning sensor head, etc.)
b. Mechanical design should target the marking criteria stated at the beginning of this document

Software

a. Robot-C
   a. The light sensor attached to robot, scans the barcode saves the light sensor values to the data-log (see canvas datalog sample code).
   b. The datalog file holding the light sensor values and encoder values acquired in the previous step is uploaded to the PC using the robotC Robot->LegoBrick->file Management Utility.
   c. On the PC, rename the data-log file extension to be a comma separated values (csv) file.
   d. Import the csv file into MATLAB. Read and plot the light sensor values versus encoder values. If the data is inconsistent or insufficient, a faster sampling rate or slower speed of the robot arm maybe required. The use of gear ratios (geared for torque to increase accuracy) is required.

b. **MATLAB**
   a. Moving average algorithm (from scratch not using MATLAB built-in functions) to filter the noisy light sensor values
   b. Derivative (from scratch not using built-in functions) to detect edge at each black or white region in the bar code.
   c. Findpeaks (a built-in function can be used) to find the index of the start and end (i.e. width) of each region in the bar code.
   d. Once the width and color of each region in the barcode is found, design a simple lookup table to find the corresponding ASCII character.
   e. Display the ASCII character in the command window.