Chase-Bot

Emmanuel and Shuto
Project Overview

未成年人

Motivation

Method

Centralize the object

Desired position, size of the object
Project Overview

❖ **Features**

➢ **Chasing**

➢ **Goalkeeping**

Same Algorithm!!
Hardware

2WD Smart Robot Car Chassis Kit

L298N Motor Driver

OV7670 Camera
Hardware

Upper view of the car:
- Camera
- Battery Box for Motor Driver
- FPGA

Back view of the car:
- Camera
- Motor Driver
- Power for FPGA
L298N Motor Driver

Pulse Width Modulation(pwm)

https://www.electronics-tutorials.ws/blog/pulse-width-modulation.html

http://www.bristolwatch.com/L298N/index.htm
Tracker Module

- Camera get Image
- RGB2HSV
  - To easily threshold in color
- Binarize
  - To easily count inside pixels inside the ball
- Estimator
  - Radius of all 1’s in a circular ball
  - Average of positions of pixels with value
Initializer

From Tracker

cur_pos, cur_rad

User Inputs

Command buttons

Activate switch

Confirm button

Initialize module

box

Box pixel

Camera Signals

Bram

Camera inputs

Pixels

Camera pixel

Configuration

Cursor

Cursor pixel

Cursor position

goal_pixel

goal_rad

To Control

To Tracker

XVGA signals
Initializer

+ IF ACTIVATE SWITCH
  + Initializing mode
    + Cursor active
    + Select pixel
  + Confirm mode
    + Confirm initial position and radius from tracker
+ ELSE
  + Activate control module
  + Deactivate cursor
Controller

From Tracker

- Controller module
  - cur_pos
  - cur_rad

From Initializer

- goal_rad

Control

- Signed Speed
- Signed Turn

Motor Out

- enable
- inA, inB

User Inputs

- Parameter Switches
Controller

Control module

Inputs
Current position and radius, Goal radius

Outputs
speed, turn

Equations for chasing
\[
\begin{align*}
\dot{r} &= r_{des} - r \\
\dot{x} &= x_{des} - x \\
\delta\dot{r} &= r_n - r_{n-1} \\
\delta\dot{x} &= x_n - x_{n-1} \\
speed &= K_{sp} \cdot \dot{r} + K_{sd} \cdot \delta\dot{r} \\
\text{turn} &= K_{tp} \cdot \dot{x} + K_{td} \cdot \delta\dot{x}
\end{align*}
\]

Equations for Goalkeeping
\[
\begin{align*}
\dot{x} &= x_{des} - x \\
\delta\dot{x} &= x_n - x_{n-1} \\
speed &= K_{sp} \cdot \dot{x} + K_{sd} \cdot \delta\dot{x} \\
\text{turn} &= 0
\end{align*}
\]
Motor Out module

Inputs
speed, turn

Outputs
enable, inA, inB for each motors

Converts the pixel space speed and turn into the motor outputs as needed for the motor driver.

<table>
<thead>
<tr>
<th></th>
<th>InA</th>
<th>InB</th>
<th>Enable</th>
</tr>
</thead>
<tbody>
<tr>
<td>forward</td>
<td>HIGH</td>
<td>LOW</td>
<td>HIGH (pwm)</td>
</tr>
<tr>
<td>backward</td>
<td>LOW</td>
<td>HIGH</td>
<td>HIGH (pwm)</td>
</tr>
</tbody>
</table>

Conversion from car speed to motor speed
Goals

➢ Baseline & Expected goals
  ○ Initialize which object to follow through an interface
  ○ Chasing Task
  ○ Goalkeeping task

➢ Ideas for stretched goals
  ○ Generalize the object
  ○ Fetching the object
  ○ Self-estimation using the speed encoder information
  ○ Future prediction of the ball trajectory
  ○ Calculating the real distance
<table>
<thead>
<tr>
<th>Date</th>
<th>Emmanuel</th>
<th>Shuto</th>
</tr>
</thead>
<tbody>
<tr>
<td>11/04 - 11/10</td>
<td>Camera</td>
<td>Hardware</td>
</tr>
<tr>
<td>11/11 - 11/17</td>
<td>Tracking Module</td>
<td>Controller Module</td>
</tr>
<tr>
<td>11/18 - 11/24</td>
<td>Tracking Module</td>
<td>Controller &amp;Initializer Module</td>
</tr>
<tr>
<td>11/25 - 12/01</td>
<td>Initializer, Tuning</td>
<td>Initializer, Tuning</td>
</tr>
<tr>
<td>12/02 - 12/08</td>
<td>Debug, Report</td>
<td>Debug, Report</td>
</tr>
</tbody>
</table>
Thank you!