

地面の彩色による車両型ロボットの速度制御

2020SC666 潮田響

指導教員：大石泰章

1 はじめに

近年、自動車の自動運転化が進行している。そこで自動運転化に伴う、自動車の安全性に注目が集まっている。自動車の安全性において、速度を制御することが重要である。カーブや交差点の前の地面を彩色し、危険な箇所を明示することで、自動車の速度を落とさせることは、現在でも広く行われている。これまで人間に情報を与えるために地面の彩色は行われてきたが、これを自動走行車両でも読み取れるようにすることで、人間と機械の両方で情報が得られるため、より安全な走行を目指せるのではないかと考える。

本研究は、車両型ロボット Zumo を使い、地面の彩色によりその速度を制御することを試みる。まず、スタートとゴールを設定し、コースのラインの上をロボットに走行させる。また、コースのラインの横に色の線を塗る。そして、カメラを用いて色の線を認識し、色ごとに異なる命令を与えることでロボットの速度を制御することを考える。

2 使用機材について

本研究では、図 1 で示した車両型ロボット Zumo(青丸で囲った部分) および、小型カメラ Pixy2CUMcam5(赤丸で囲った部分) を用いて行う。Zumo は、幅 10cm × 奥行 10cm × 高さ 5cm の車両型ロボットであり、Zumo の上部には Arduino Leonardo が搭載されている。Arduino Leonardo は Arduino IDE というソフトウェアを用いることでプログラムの書き込みと実行をすることができる。

Pixy2Cumcam5 はソフトウェア Pixy Mon を用いることで最大 7 色まで色を認識させることが可能である。Pixy2 と Zumo の上部にある Arduino Leonardo を接続し使用する。

Zumo の下部には図 2 のような赤外線センサが取り付けられており、このセンサは地面に赤外線を当て、反射される赤外線の色に応じて黒と白を判別する。

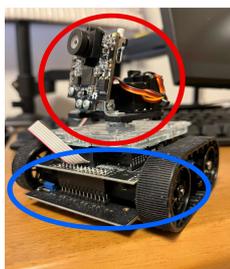


図 1 車両型ロボット Zumo(青丸) と小型カメラ Pixy2Cumcam5(赤丸)



図 2 赤外線センサ

3 問題設定

本研究では、車両型ロボット Zumo を自動車に模擬して実験を行う。図 3 のようにスタート地点からゴール地点を繋ぐ黒線を引き、その黒線の上を走行させる。

黒線の上をロボットが走行している際、黒線の横に塗られている色を Pixy2 で認識する。色を認識することができたら、その色に応じた命令が与えられ、ロボットの走行速度が変化する。具体的には、赤色(図 3(a))を認識したら、カーブがあるので減速し、次の青色(図 3(b))を認識したら、直線なので加速、最後の緑色(図 3(c))を認識したら、緩やかなカーブなので少し減速、という命令を与える。

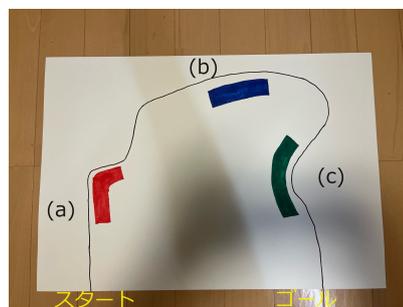


図 3 作成したコース

4 Zumo の制御について

4.1 ライントレース

ライントレースは Zumo の下部に取り付けられている赤外線センサを用いて行う。6つのセンサはそれぞれ0~5の番号が振り分けられており、黒線がセンサ2とセンサ3の中間に来ることを目標として、PD制御を用いて右車輪と左車輪の速度をそれぞれ変化させる。このプログラムを開発環境 Arduino IDE で作成し、指示を与えることでライントレースを実現する。

プログラムを実行した結果、ロボットはライン上を乱れることなく走行し、ゴールまで到達した。

4.2 速度の指示について

本研究ではカーブで減速，直線で加速という指示をロボットに与える．速度の指示はプログラムの中の「`motors.setSpeeds(,)`」というプログラムを用いて行い，カッコ内の数値を変更することで指示を行う．

プログラムを実行した結果，(400,400)の時は0.45m/s，(250,250)の時は25m/sとなり，ロボットは変更した数値に応じて速度を変えた．

5 Pixy2による色の認識

色の認識を行うために，まずPixy2のカメラで地面に塗ってある色を映す．この時にPixy2に内蔵されている「Color Connected Components」いうプログラムを用いることで，映した色を識別させる．そしてPixy Monで色を認識させることができる．認識させると色ごとのカラーコードを得ることができる．これをArduino IDEに送り，プログラム中で条件分岐などに使うことができる．

プログラムを実行した結果，Pixy Monの画面において図4のように赤いボールに「red」というラベルが付き，Pixy2に色を認識させることができたことが確認できる．

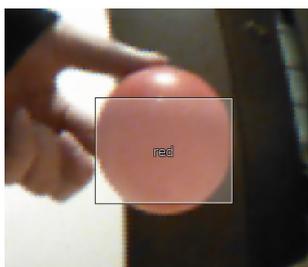


図4 Pixy2に認識させたカラーボール

しかし，今回の実験ではPixy2で得ることができたカラーコードをプログラムに反映させ，色に基づいてロボットの速度を制御することはできなかった．

6 制御できなかった原因の考察

制御できなかった原因は，Pixy2で読み取った色の情報をArduino IDEにうまく送信できなかったことであると考えられる．読み取った色の情報は「`blocks=pixy.ccc.getBlocks`」というコードを使用し，`blocks`の中にそれぞれの色に対応する数値 (`red=1`, `blue=2`, `green=3`)として代入する．そしてif文を使い，1のときは減速などのように，色によって異なる指示を与える．しかし，`blocks`の中に意図した数値が入らず，色の情報送信できていないことが分かった．

7 おわりに

今回の実験では，ロボットを黒線の上を走行させ，速度を指示することができた．また，Pixy2を用いて色を認識させることもできた．しかし，Pixy2で認識した色の情報を，Arduino IDEに送信することができず，色に基づいた

ロボットの制御を行うことができなかった．

今後の改善点として，`blocks`に意図した数値を代入し，Pixy2からArduino IDE間の情報送信を成功させることが挙げられる．

参考文献

- [1] Pixy Documentation,
<https://docs.pixycam.com/wiki/doku.php?id=wiki:v2:start>