

画像処理を用いた車両型ロボットの走行制御

2018SC071 岡本晃青

指導教員: 大石泰章

1 はじめに

近年では、外食チェーン店や介護施設において、人手不足対策のため配膳ロボットの導入が進められている。配膳ロボットのような無人搬送車の開発にはカラーマーカーを用いたものがある [1]。

本研究では配膳ロボットへの応用を意識し、頭上に設置したカメラセンサを使って一定の領域を観測し、この領域内で指定した目的地に走行する車両型ロボットの制御法を考える。

2 使用する実験機とシステム

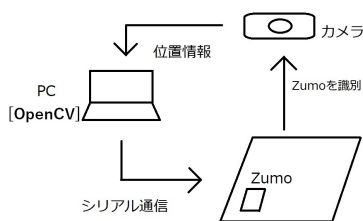


図1 構成するシステム

本研究では図1のようなシステムを構成する。カメラは Web カメラ HD1080p (図2) を使用し、上方から車両型ロボットが走行する範囲を撮影する。撮影した画像に対して PC 上で OpenCV を使った画像処理を行うことにより車両型ロボットの位置と向きを取得する。車両型ロボットは Zumo (図3) を使用し、Arduino の開発環境である ArduinoIDE を使ってプログラムを書き込むことで動作を制御する。

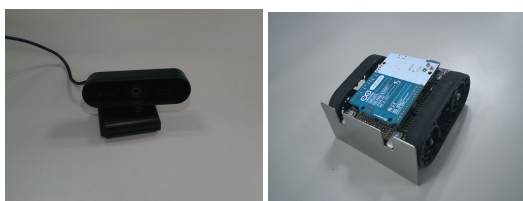


図2 Web カメラ HD1080p 図3 車両型ロボット Zumo

3 目指す動作

図4のようにカメラで撮影する範囲を縦横それぞれ4分割して、16個の領域に分ける。左上の領域を起点として、横方向に i 番目、縦方向に j 番目の領域を (i, j) と表す。これらのうちの任意の領域にロボットを置き、そこをスタート地点とする。また設置時のロボットの方向は図の上下左右い

ずれを向いていてもよいとする。そしてあらかじめ定められたゴール地点の領域までロボットを走行させる。

図4ではスタート地点が領域 (2,3)、ロボットの初期方向が左向きであり、そこからゴール地点として定められた領域 (4,1) まで走行させるとしている。



図4 撮影した画像

4 アルゴリズム

4.1 走行の手順

本研究では以下の手順を踏むことで3章で述べたような走行を実現する。はじめにカメラによる撮影を行い、取得した画像から、ロボットの位置と向きの初期設定を読み込む。これをもとに OpenCV 上でゴール地点までの走行計画を決定する。その内容をシリアル通信でロボットに指示を与えることで走行を実現する。

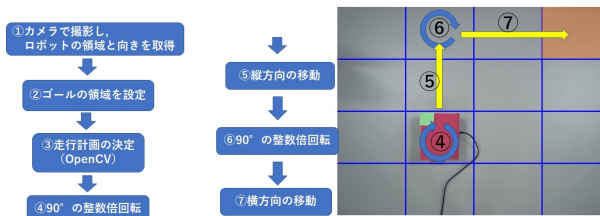


図5 走行制御の手順

なおロボットの走行は左右のモータを必要な角度だけ回転させることで行われる。本研究では、縦方向に1領域分前進するための回転角、横方向に1領域分前進するための回転角、その場で右回りに90°向きを変えるための回転角、同じく左回りに90°向きを変えるための回転角をあらかじめ求めておき、これらを整数回組み合わせることで期待されるような走行を実現する (図6)。

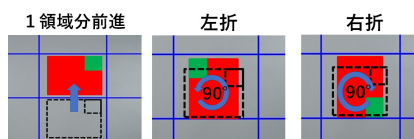


図6 ロボットの動き

4.2 位置情報と向きの取得

ロボットのスタート地点とその向きの情報はロボット設置時にカメラで撮影した画像から OpenCV の色検出機能を用いて取得する。今回は赤色と緑色の色検出を行ってロボットを認識するため、ロボットの上に赤色と緑色の画用紙を図 7 のように張り付けてある。緑色用紙は Zumo の進行方向に対して、右上隅の位置に赤色用紙の上から張り付けてある。

撮影した画像を図 8 に示す。画像中に左上隅の点を原点とし、右向きに x 軸を、下向きに y 軸を取って、画素数を単位とした座標を導入する。取りうる座標の範囲は $0 \leq x \leq 640, 0 \leq y \leq 480$ である。検出した色の周りをそれぞれ最小の矩形で囲み、その中心の点の座標を取得する。赤色の中心座標にもとづいてロボットが位置する領域を判定する。またロボットの向いている方向は赤色の中心にもとづいて構成した図 9 の①～④の 4 つの領域に対し、緑色の中心点がどこにあるかで判断する。すなわち緑色の中心が①にあるならば左向き、②にあるならば上向き、③にあるならば右向き、④にあるならば下向きと判断する。



図 7 色検出するロボット

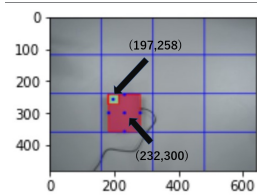
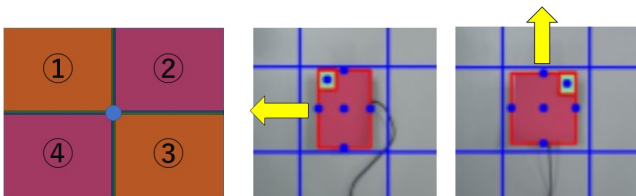


図 8 撮影した画像



赤色の中心にもとづいて構成した 4 つの領域
 緑色の中心が領域の①に属するとき Zumo は左向きと判定
 緑色の中心が領域の②に属するとき Zumo は上向きと判定

図 9 ロボットの向きの判定

4.3 走行計画の決定

ロボット設置時の位置と向きが得られたら、それをもとにしてゴール地点まで走行するのに必要な前進と回転の回数を求める。はじめに縦移動のため上下でゴールのある方向に回転し、スタート地点とゴール地点の縦の領域の差だけ前進させる。次に横移動のため左右で方向転換する、そして各領域の横の差だけ前進させて完走となる。

図 10 ではスタート領域座標を (i_s, j_s) 、ゴール領域を (i_g, j_g) とおいたとき、 j_g と j_s の大小関係と差からロボット

は上を向き、2 領域進む。次に i_g と i_s の大小関係と差からロボットは右を向き 2 領域進み完走となる。

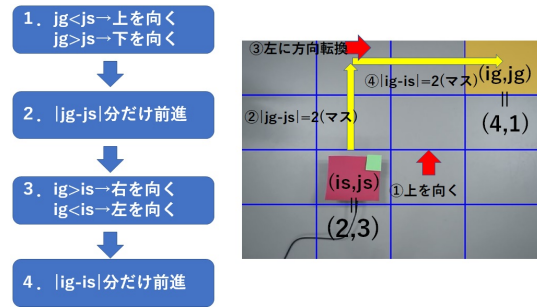


図 10 走行計画

5 実験結果

以上のアルゴリズムを用いて図 4 のようにスタート地点と初期方向およびゴール地点が与えられたとして実際にロボットを走行させた。その際に取得した座標の変化を図 11 に示す。図 11 からロボットは目標どおり領域 (2,3) から領域 (4,1) に移動していることがわかる。また本研究が施行した以外の領域や向きの設定の場合も問題なく走行することができた。

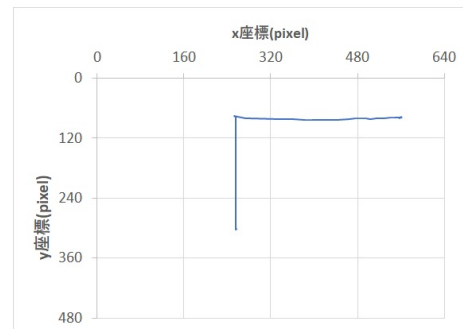


図 11 座標の変化

6 おわりに

スタート地点と初期方向を任意としたうえであらかじめ定められたゴール地点までロボットを走行させることを、画像処理にもとづいて実現した。

本研究では走行領域内に障害物がないことを仮定したが、実用的には障害物があることが多いと考えられ、障害物回避の機能が必要であると考えられる。今後はそのような制御方法も考える必要がある。

参考文献

- [1] 岩岸 恭諒, 池田 貴公, 川村 拓也, 大坪 克俊, 山田 宏尚: カラーマーカーを用いた AGV の位置合わせ技術に関する研究, 東海支部総会講演会講演論文集. 2020.69 巻 (2020).
- [2] 北山直洋: 『Python で始める OpenCV4 プログラミング』. カットシステム, 東京, 2019.