

視覚センサを用いた自律移動ロボットの開発

Development of Autonomous Mobile Robot Using the Vision Sensor

○藏前智映*, 及川一美*, 大久保重範*, 高橋達也*

○Tomoaki Kuramae*, Kazumi Oikawa*,
Shigenori Okubo*, Tatuya Takahashi*

*山形大学工学部

*Yamagata University

キーワード: 視覚センサ(Vision Sensor), HSV, ラベリング処理(Labeling), ヒストグラム(Histogram),
ラインブロック(Lineblock), 重心(Gravity)

連絡先: 〒992-8510 山形県米沢市城南4-3-16 山形大学工学部 機械システム工学科 大久保研究室藏前智映,
Tel.: (0238)26-3245, Fax.: (0238)26-3245, E-mail: dmr07320@dip.yz.yamagata-u.ac.jp

1. はじめに

何世紀も前から、人は機械が自分で考え、人間に尽くしてくれる事を夢見てきた。今ではその夢は新世代ロボットの登場とともに、現実のものとなってきている。将来、ロボットは感覚や感情を持つことを除けば、私たち人間ができることは何でもできるようになるだろう。しかし、現時点のロボットはコンピューターにプログラムされた命令に従って、いくつかの仕事をこなすしかできない。最新鋭ロボットは簡単な意思決定ができるようになっているが、これも実はコンピューターの命令を基にしている点では同じと言える。

自律移動ロボットは、近年多くの大学や研究機関で研究が行われ、様々なロボットが研究されている。私達の研究室で開発している自律移動ロボットはビヘイビアベースド・ロボットが比較的性能の低いセンサやマイコンで柔軟な行動を示すこと

が可能である点に着目している。低性能のセンサやコンピュータではロボットのできることは制限されてしまうが、ローテクなロボットでも特化したある仕事をこなせることを示すことで、安価でセットアップコストの低いロボットを実現できるのではないかと考える。

我々の研究室ではサブサンクション・アーキテクチャ(SA)をベースとした車輪型自律移動ロボットがイベント駆動型階層構造切替手法を用いてのナビゲーションを実現している。そこで、このイベント駆動型階層構造切替手法がどのぐらいの高度なタスクに対応できるかを調べることで、つまり、今までのナビゲーションに目標物を見つけるタスクや物を捕捉して移動させるタスクを追加したとき対応できるかについて、視覚センサであるカメラを用いて検証を行う。

2. システム構成

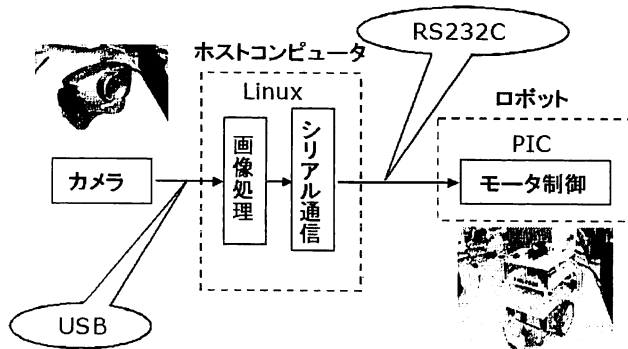


Fig. 1 System Configuration

カメラにはクリエイティブメディア社製Video BLASTER WEBCAM Plusを使用する。解像度は 350×240 である。ホストコンピュータには日本データシステム株式会社製PCM-5335を用いCPUはNS Geode GX1(Pentium MMX 300MHz相当)、メモリは最大128MBである。また、Video4Linuxを用い画像処理プログラムの作成を行うためにOSにはLinux red hat 2.4を使用する。ロボットにはナムコ製マイクロマウスを用いる。

3. 画像処理

3.1 アプリケーション

Video BLASTER WEBCAM Plusからの取得画像を画像処理するために、Fig.2のようなアプリケーションを作成した。effect buttonを押し、実行したい画像処理を選択する。

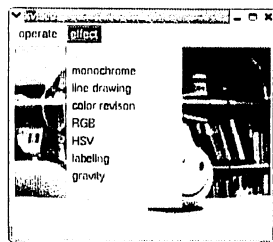


Fig. 2 Application

3.2 画像処理の流れ

本研究での画像処理は、ロボットに目標物（ボール）を認識させ向かわせることである。色情報を元にした重心により目標物の位置を求めることにした。処理の流れをFig.3に示す。

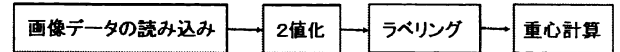


Fig. 3 The flow of image processing

3.3 2値化

3.3.1 HSVによる2値化

RGB形式がR,G,Bの各原色の濃度の組み合わせから合成色を作るのに対して、HSV形式では色の種類を表す色相(H:hue)、色の明るさを表す明度(V:value)、色の濃さを表す彩度(S:saturation)で表現している。本研究では撮影物の周りの明るさによる外乱的な反射光などの影響を軽減するためにHSV形式で進めていく。そのためRGB形式からHSV形式に変換した。

変換式

$$\text{明度 } V = \text{MAX}(R, G, B)$$

彩度

$$S = (\text{MAX}(R, G, B) - \text{MIN}(R, G, B)) / \text{MAX}(R, G, B)$$

$\text{MAX}(R, G, B) \leq \text{MIN}(R, G, B)$ のとき、彩度 $S = 0$

色相

$$\text{MAX}(R, G, B) = R \text{ なら}$$

$$H = 60 * (G - B) / (\text{MAX}(R, G, B) - \text{MIN}(R, G, B))$$

$$\text{MAX}(R, G, B) = G \text{ なら}$$

$$H = 60 * (B - R) / (\text{MAX}(R, G, B) - \text{MIN}(R, G, B)) + 120$$

$$\text{MAX}(R, G, B) = B \text{ なら}$$

$$H = 60 * (R - G) / (\text{MAX}(R, G, B) - \text{MIN}(R, G, B)) + 240$$

$\text{MAX}(R, G, B) > \text{MIN}(R, G, B)$ のときのみ

色相Hと明度Vの範囲を以下のようにして画像処理を行った。

$$40 < H < 80, Y > 150$$

4. ラベリング処理

4.1 ヒストグラムを用いた接続成分検出法

この手法はHSV処理後の、黄色の部分だけが抜き出された画像に対して、まずは一行ずつ黄色の画素がどのぐらい含まれているかを調べる。そしてFig☆☆のようながその個数のヒストグラムを作成する。次に縦の一行ずつにも同様の処理を行ってヒストグラムを作成する。最後にそれぞれの座標を目標物の推定位置と見なす手法である。この手法は2値化された画像にラベリング処理を行い最大面積ラベルを求めていないため処理時間が他の手法に比べ実用的であり、ロボカップなどで使用されていたので、本研究でも主として取り入れてきた。しかし推定位置が離散的という問題を解決だった。そこでラインブロックを利用したラベリング手法を提案する。

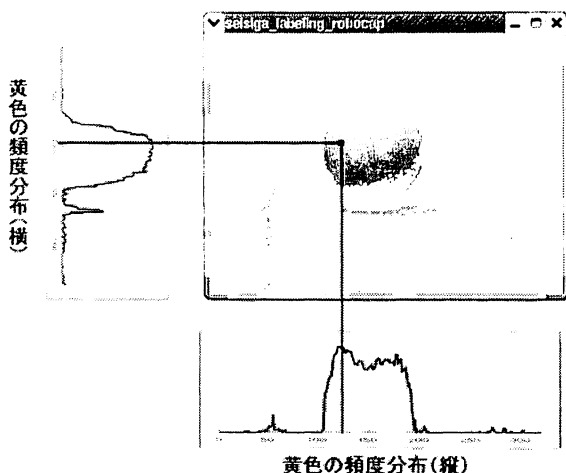


Fig. 4 histogram

4.2 ラインブロックを用いたラベリング

この手法は従来のラベリング手法が既走査の4つの近接画素で繋がりを調べるのに対し、ラインブロックを利用したラベリング手法は上下のラインブロックの最大値・最小値の位置関係で繋がりを

を調べる点で異なる。これによって処理をする画素が軽減しより実用的な正確かつ処理速度を実現できると考える。

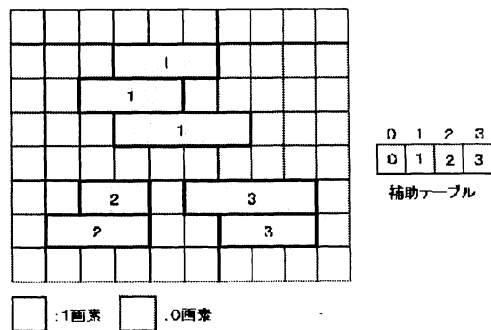


Fig. 5 lineblock

またラベルの衝突の処理の際には既走査ブロックのラベルを付けなおすのではなく、予め用意した補助テーブルの中の情報を変更し更新していく。(Fig.6) そうすることによって連結成分のラベルは異なる値を持つことになるが、重心位置を求めるなどの解析は、補助テーブルの値を利用することで解決できる。

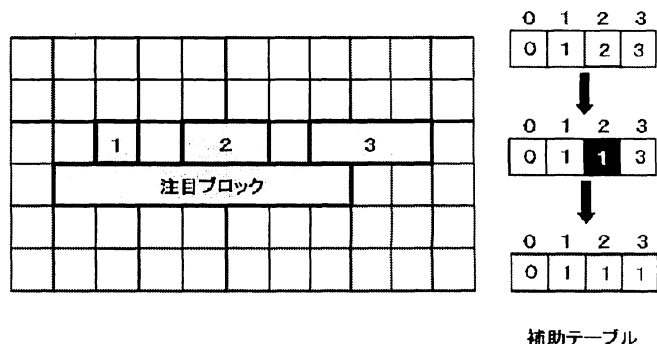


Fig. 6 collision

4.3 実験

各ラベリング手法の平均処理時間を調べる実験を行った。カメラと目標物（黄色のボール）の距離を一定に固定した。

4.4 実験結果

Table1に各ラベリング手法の平均処理時間[msec]を示し、Fig.7~10に推定位置の推移を示した。

平均処理時間を比較するとヒストグラムを用いた連結成分検出法より遅かったが、本稿で提案するラインブロックを利用したラベリング手法が有効であることが解る。これはヒストグラムを用いた連結成分検出法はラベリングをするのではなく、X軸・Y軸それぞれの最大画素数の交点を推定位置と見なしているため、他の手法よりも処理が少なく平均処理時間が早くなったと考えられた。

しかし、推定位置の推移ではヒストグラムはラベリングをして最大面積ラベルを調べていないためばらつきが見られ、精度の面ではその他の手法より劣っていることが解る。

我々が提案したラインブロックを利用したラベリング手法は平均処理時間においてヒストグラムとほぼ同等の処理時間であり、推定位置の推移ではヒストグラムよりも良好な結果が得られた。よってラインブロックを利用したラベリング手法を取り入れる。

Table 1 実験結果

	平均処理時間[msec]
standard	187
assist table	93
histgram	57
lineblock	59

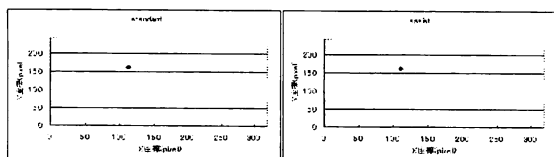


Fig. 7 standard

Fig. 8 assist

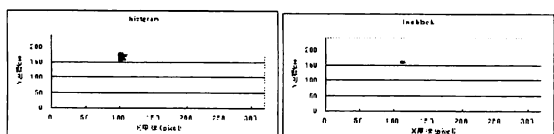


Fig. 9 histgram

Fig. 10 lineblock

5. ロボット

5.1 ロボット概観と仕様

本研究で用いるロボットの概観をFig.11, Fig.12にTable2に主な仕様を示す。



Fig. 11 front

Fig. 12 back

Table 2 Specification

ロボット	マイクロマウス (ナムコ)
モータ	ステッピングモータ×2 103-540-36(山洋)
サイズ	130×150×130(W×D×H[mm])

5.2 ロボットのビジュアルトラッキング

ロボットのビジュアルトラッキングを実験した。詳細は当日紹介する。

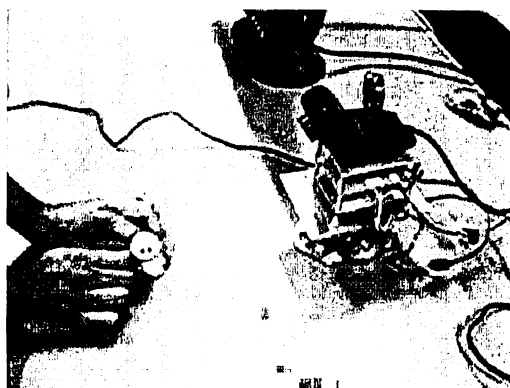


Fig. 13 One scene of experiment

6. 今後の課題

本稿で提案したラインブロックを利用したラベリング手法で目標物を見つけるタスクは解決でき

た。今後は物を捕捉して移動させるタスクを追加し、ナビゲーションを実現させたい。

7. おわりに

本研究ではロボットが目標物を認識し掴み運ぶシステムを構築するためにカメラによる物体認識について論じた。その結果、我々が提案する目標物を抜き出す手法について検証して、ロボットが目標物の重心の位置により特定物体の位置を認識し行動することを実証した。

参考文献

- 1) 井上誠喜、他：C言語で学ぶ実践画像処理、オーム社（1999）
- 2) 飯尾淳：Linuxによる画像処理プログラミング、オーム社（2000）
- 3) 及川一美：“切替可能な階層構造を持つ行動規範型自律移動ロボットの行動決定”、日本ロボット学会誌（2003）
- 4) 及川一実：“自律移動ロボットにおける「考える」”、平成16年山形大学公開講座資料（2004）