

# 視覚センサを用いた自律移動ロボットの開発

○藏前智映 大久保重範 (山形大学) 及川一美 (山形大学)

## Development of the Autonomous Mobile Robot Using the Vision Sensor

\*T. Kuramae, S. Ohkubo(Yamagata University) and K. Oikawa (Yamagata University)

**Abstract**— We pay attention to the point that Behavior-based autonomous can show a flexible action with the sensors and the microcomputer with comparatively low performance in our laboratory. If you can show that even such a robot is good at the navigations or the transports. A cheap, useful robot can be achieved. The purpose of this research is for the robot to move based on the navigation, to catch the object, and to move it to the destination. This paper discusses about the robot with a camera approaches a object and catches.

**Key Words:** Vision Sensor, RGB, HSV, Labeling, Gravity

### 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。

最終的な目標はロボットがナビゲーションを基に移動し、対象物を捕捉し目的地まで移動させることである。本研究では、視覚センサであるカメラを用いて画像処理を行い、対象物(黄色のボール)を追いかけ捕捉することができるロボットについて論ずる。

### 2 システム構成

Fig.1に本研究でのシステム構成と流れを示す。ホストコンピュータにUSBで接続されたカメラから画像を取得し、ホストコンピュータで画像処理を行う。そして、RS232Cを介してロボットのPICとシリアル通信をしてモータを制御している。

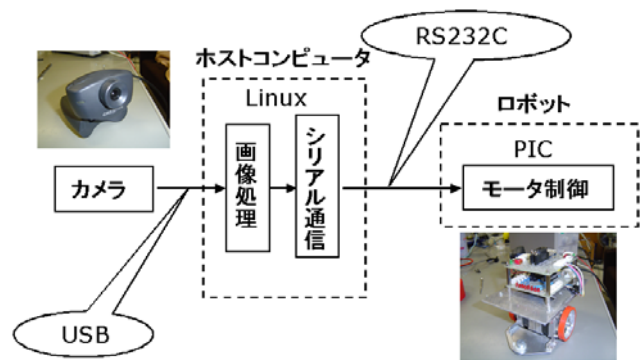


Fig. 1: System configuration

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、PICにはPIC16F873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。

ボタンを押し、実行したい画像処理を選択する。また、動画を静止画として取得でき、ppmファイルとして保存できる。

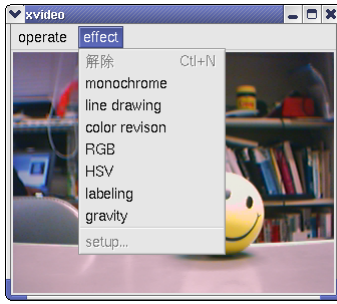


Fig. 2: Application

### 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（黄色のボール）を認識させ向かわせることである。そこで、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。

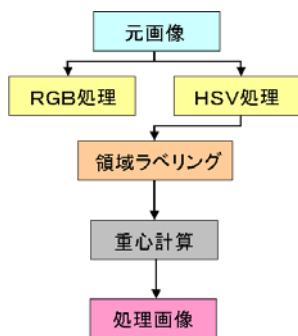


Fig. 3: The flow of image processing

処理の流れとして、初めにカメラから画像を取得した後、RGB形式からHSV形式に変換し、色相・明度の範囲で黄色部分だけを抜き出す。次にその抜き出した黄色部分に領域ラベリング処理を行い、最大面積ラベル（一番大きな画素の塊）だけを抜き出し表示させる。最後に最大面積ラベルの重心を求める。

### 3.3 色認識処理

#### 3.3.1 RGBによる画像処理

RGB形式では1つの色を表現するのに赤・青・緑の3つのパラメータで表現している。そこでPhotoshop Elementsで目標物である黄色のボールのR,G,B値をそれぞれ調べた。しかし、光の当り具合により値が変化するため、R,G,B値それぞれの最大値、最小値を調べた。その範囲を以下に示す。

$$\begin{aligned}
 145 < R < 250 \\
 145 < G < 255 \\
 0 < B < 69
 \end{aligned}$$

以上の範囲で画像処理を行った結果を Fig.4 に示す。



Fig. 4: After RGB processing

#### 3.3.2 HSVによる画像処理

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形式に進めていく。そのためRGB形式からHSV形式に変換した。変換式を以下に示す。

変換式

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

彩度

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

MAX(R,G,B) < MIN(R,G,B) のとき、彩度 S=0

色相

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

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

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

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

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

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

MAX(R,G,B) > 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。(Fig.5) ただし、指定した範囲内の黄色以外は白色で塗りつぶしている。

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



Fig. 5: After HSV processing

## 4 ラベリング処理

ラベリング処理とは、2値化された画像において、それぞれの連結成分に固有の番号をつける作業のことであり、ラベリング処理後、連結成分の最大面積を抜き出すことによって、背景の領域からボールの領域を抜き出すことができる (Fig.6)。本研究ではラベリングによって連結成分のサイズ (面積) を調べ、最大サイズの連結成分以外はノイズとして除去を行う。本研究内ではこのノイズ除去作業までを含めてラベリング処理とする。

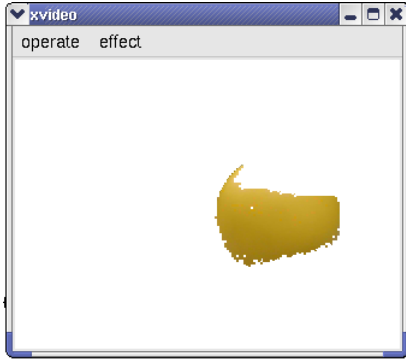


Fig. 6: After Labeling processing

## 5 重心

ラベリング処理後、最大面積の重心を求める。重心を求める式は以下の通りである。画素の位置  $(x_i, y_i) (i = 0, \dots, n - 1)$  の平均値

$$\left( \frac{1}{n} \sum_{i=0}^{n-1} x_i, \frac{1}{n} \sum_{i=0}^{n-1} y_i \right) \quad (1)$$

重心の求める処理を行った結果が Fig.7 である。

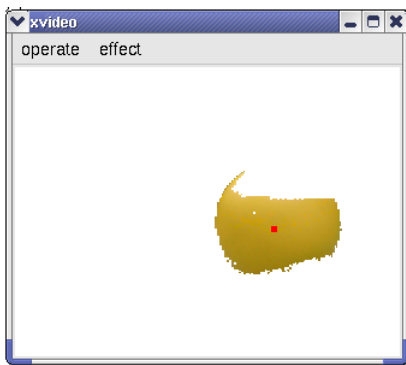


Fig. 7: After center-of-gravity processing

ラベリング処理後の最大面積は不完全な形であったがボールの動きを認識するためには十分な結果が得られた。

## 6 ロボット

### 6.1 ロボット概観と仕様

本研究で用いるロボットの概観を Fig8, Fig9 に Table 1 に主な仕様を示す。



Fig. 8: Front



Fig. 9: Back

Table 1: 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])

### 6.2 物体追跡方法

ロボットが目標物に向かうことができるように、1画面を分割し、さらに場合分けして物体を追跡する。

#### 6.2.1 目標物が画面中心にある場合

ボールが中心の領域にある場合、その領域をさらに3分割して、目標物の重心がどの領域に存在しているかに応じて、モータの回転を変化させる。

- ボールが画面上部にある場合：前進
- 画面中央にある場合：静止
- 画面下部にある場合：後進

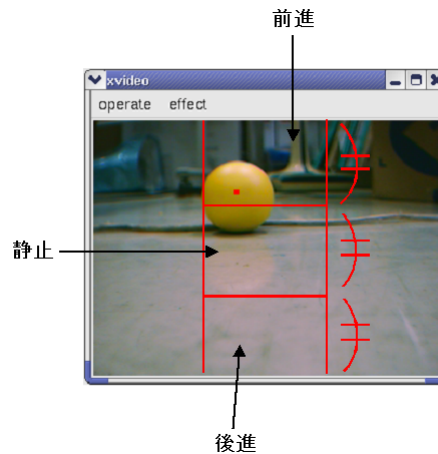


Fig. 10: The division method of center

## 6.2.2 目標物が画面左または右にある場合

ボールが左または右の領域に存在する場合、その領域をさらに2分割して6.2.1節の場合と同様に、目標物の重心の位置に応じて左右のモータのスピードを変化させ左右旋回させる。

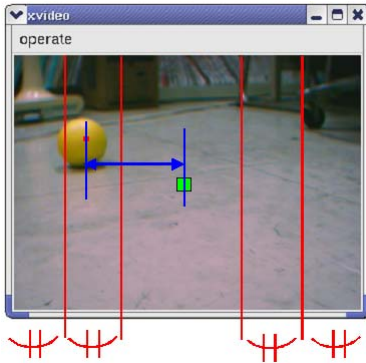


Fig. 11: The division method of left and right

## 7 実験

ロボットが目標物（黄色のボール）向かっていく実験を行った。詳細は当日紹介する。

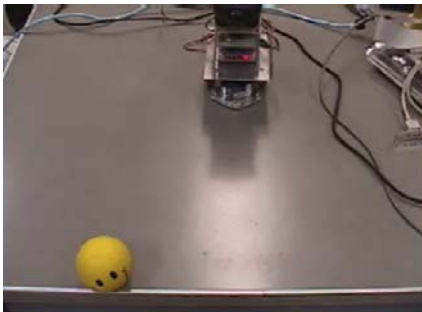


Fig. 12: One scene of experiment

## 8 考察

本手法では、ロボットに目標物（黄色のボール）を認識させ向かわせるために、色情報を元にした重心により目標物の位置を求め、追従させることができた。しかし、カメラから画像を取得してから重心を求め表示するまでに時間がかかり、実環境にすばやく対応しているとは言えない。その原因としてラベリング処理に問題があると考えられる。本研究のラベリング処理では、すべての画素1つ1つに対して8近傍を調べ連結成分を探すという手法を取り入れている。この手法では、計算量が膨大で無駄な処理が多い。例えば、ラベリング処理途中で最大面積ラベルを見つけた後も、すべての画素について調べてしまう場合や、8近傍を調べるあまり同じ画素を重複して調べてしまう場合などがある。無駄のないアルゴリズムの作成が考えられるが、この部分の改善は今後の課題とする。

## 9 今後の課題

今後の課題として、第一に考察で論じた画像処理プログラムの処理時間を短縮する。中でもラベリング処理時間の短縮し、よりの確にボールの位置を解析できるようにする。現段階ではロボットは目標物の重心が分割された画面のどの領域に存在するかを認識してモータ駆動時間を変化させ前後左右方向を変えているため、正確に目標物に向かっていないわけではない。そこで、画面の中心と目標物の重心との距離に応じてロボットの旋回角速度を変化させる方法などを検討し、改良していく。また本研究では取得画像から黄色の物体を見つけ抜き出し、最大面積の物体だけを表示するようにしている。しかし、黄色のボールとよく似た色をした物体が近くにある場合区別ができていないという問題点がある。その問題点の改善方法としては物体の特徴パラメータである円形度、または周囲長を調べ抜き出す手法を取り入れることで区別することが可能であると考えられる。

第二に目標物を拘束して掴み運ぶ機構を考える。具体的にはタッチセンサを用いた手を作製し目標物を捕捉できるようにしていきたい。

## 10 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、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ができた。また、考察で本手法の問題点を挙げ今後の課題と展望を示した。

## 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)
- [5] 服部昌博: C言語とデータ通信、工学図書株式会社 (1994)