

視覚移動ロボット「晴信4号機」の開発 No. 3

4C-4

—向目標移動のサインパターン抽出と追跡—

○湯口 徹 中山 信一 森 英雄
(山梨大学工学部)

1. はじめに

当研究室で開発している視覚移動ロボット晴信は、”サインパターンと定形行動”の概念の基に開発が進んでいる。現在、沿目標移動は実現している[1][2]。今回、高速カメラ回転台を用いて、柱、門などをサインパターンとし、それに向かう向目標移動について報告する。

2. 走行指令書

目標とするサインパターンについての情報(色、大きさ、距離など)は、走行指令書に記述するものとする。

走行指令書の形式

```
MOVING-TOWARD path_length_limit
SP_LOCATION (distance_from_last_SP,
             orientation_relative_to_last_SP)
SP-ATTRIBUTES-IN-REAL-WORLD
             (area,width,height,color)
.....
```

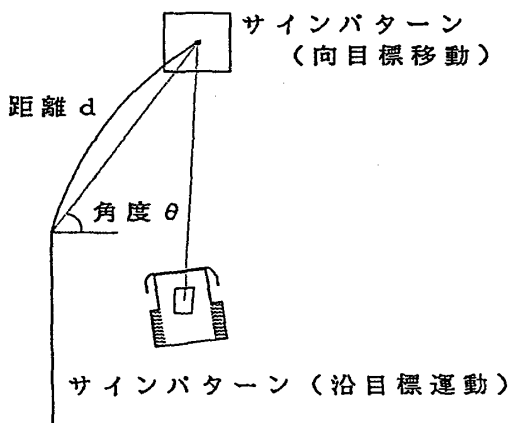


図1 概略図

画像中で求めた、サインパターンの属性値を、実空間上の値に変換し、走行指令書の値と比較する。変換は、次の式によって行う。(ここで、 X_i 、 S_i 、はそれぞれ、画像中のサインパターンの長さ、面積で、 D はサインパターンまでの実際の距離であり、 X_r 、 S_r は、実空間上の長さ、面積の値である。また、単位は X_i 、 S_i は画素数、 D 、 X_r 、 S_r はcmである。)

サインパターンの長さの変換式:

$$\text{水平方向 } X_r = X_i \times \alpha \times D$$

$$\text{垂直方向 } X_r = X_i \times \beta \times D$$

サインパターンの面積の変換式:

$$S_r = S_i \times \alpha \times \beta \times D^2$$

(α 、 β は、水平、垂直方向の変換定数である。)

3. サインパターン抽出・認識

サインパターンの各属性(色、面積、角度など)毎に判別関数を設けて、各々の属性について、三つの関数値(-1:有り得ない、0:なんともいえない、1:有り得る)によって表現する。画面上の全ての領域について、関数値を求め、-1を持たない領域を選び出す。選ばれた領域が一つしかない場合には、それを求めるサインパターンとし、二つ以上の時には、関数値の和の最大のものをサインパターンとする。もし、サインパターンが見つからないときには、ロボットを一時停止させ、サインパターンの発見に努めるようなエラー回復の定形行動をとる。

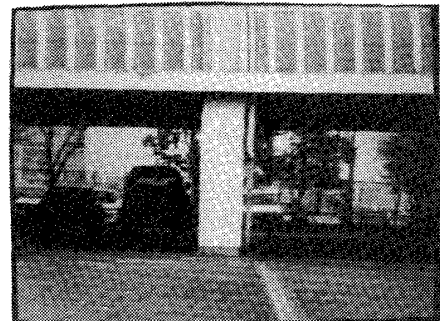


図2.1 原画像
(白い柱がサインパターン)

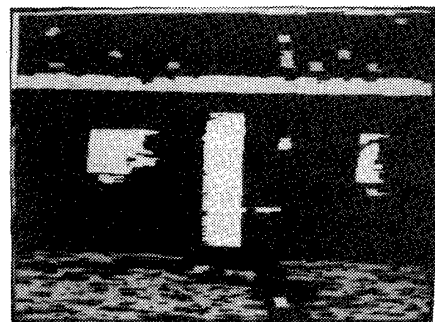


図2.2 領域分割画像

判別関数	領域番号			
	1	2	3	4
色	1	1	0	-1
面積	0	1	0	-1
角度	-1	1	-1	-1
...

図2.3 判別関数表

図2 サインパターン抽出

4. サインパターン追跡

一度サインパターンを発見したら、以下のようにして、追跡を行う。

画面を水平ライン走査して、次の条件を満たす画素を検出する。(但し、 R_{i-1} 、 G_{i-1} 、 B_{i-1} は、その前の水平ライン走査で得たサインパターンのRGB値であり、 DR 、 DG 、 DB は、RGB値の許容誤差範囲とする。)

$$ABS(R - R_{i-1}) < DR$$

$$ABS(G - G_{i-1}) < DG$$

$$ABS(B - B_{i-1}) < DB$$

この画素の個数 N と重心 X が

$$ABS(N - N_{i-1}) < DN$$

$$ABS(X - X_{i-1}) < DX$$

であれば、それをサインパターンとし、重心 X の方向に走行する。この条件に当てはまらないときには、サインパターンを見失ったものとして、サインパターン認識のプロセスに戻る。

このように、微分や構造化の処理を省くことにより、高速化を図ることができる。

5. 走行制御

サインパターンを常に画像中で認識することにより、画面の中心とサインパターンの重心とのずれが生じる。車体の制御にこのずれを利用する。サインパターン認識は高速で行えるため、画像処理と制御の遅れは少ない。画素のずれが正の時、左にカーブを切り、負の時には、右にカーブを切る。画素のずれが少ないときには直進する。カーブの割合は、図3により与える。

(1)の場合、少しの画素のずれでも制御するので、目標到達点が、ロボットの近いところに設定され、かなりの蛇行走行をする。(2)の場合には蛇行は少なく、スムーズな制御ができる。(3)の場合には、目標到達点が遠いところに設定されるため、収束しにくい。制御としては、(2)の場合が望ましいが、屋外環境の状況やサインパターンまでの距離などにより、いずれか

に決定する。画素のずれは、画面中央より右側を正、左側を負とする。

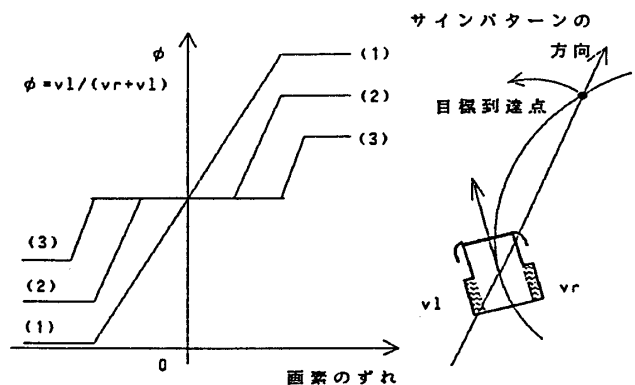


図3 制御角度の決定法

6. TVカメラ回転台の利用

上記のようにして認識、追跡したサインパターンがほぼ画面の中央にくるようにカメラ回転台を制御する。カメラと車体のずれを測るために、車体前方中央に基準棒をつける。この基準棒が、画像中のどこに映っているかを調べることにより、カメラと車体のずれ(角度)がわかる。

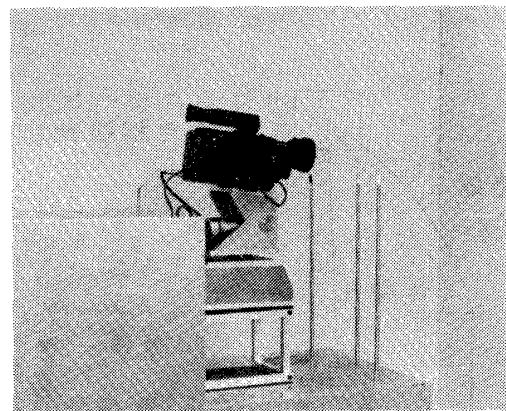


図4 基準棒

7. おわりに

このような方法により、サインパターンの抽出、認識、追跡は実現できた。処理速度は、抽出、認識は約3秒、追跡は約100msecで行うことができる。今後は、走行実験を繰り返して、安定した走行が行えることを目指す。

参考文献

[1] 視覚移動ロボット”晴信4号機”の開発
石黒、小谷、安富、片山、森(山梨大学)
第36回情報処理学会全国大会
[2] 視覚移動ロボット”晴信”の開発 No. 2
中井、齊藤、森(山梨大学)
第38回情報処理学会全国大会