

「未来2009」 画像処理MCR講習会

林 雄一

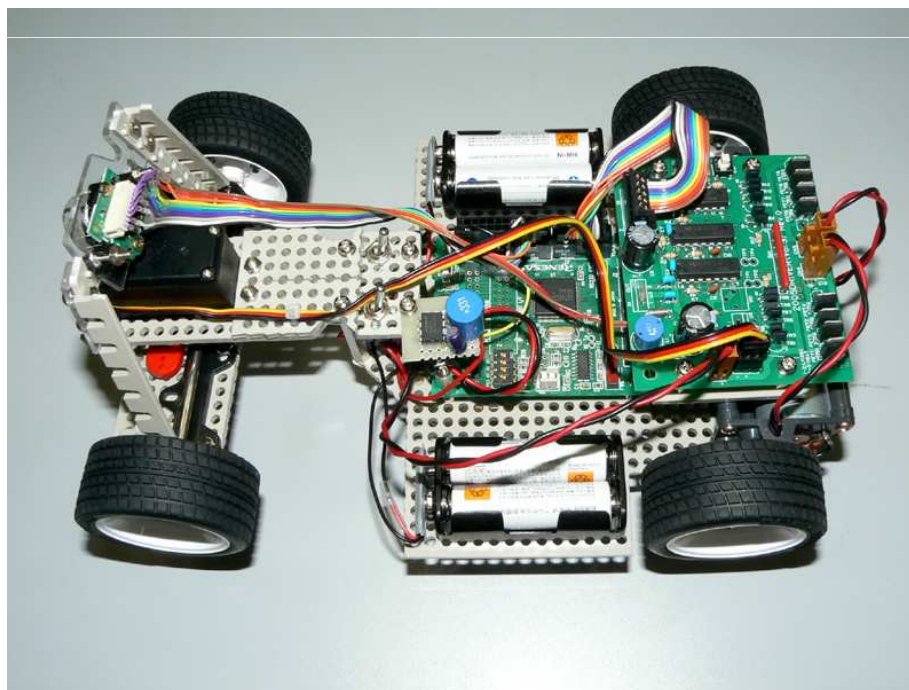
大阪電気通信大学 自由工房

- ・はじめに
 - ・ロボット経験を生かせるか
 - ・MCRの経験は少なめ
 - ・別分野での情報の共有や応用が出来る
 - ・画像処理(新しい分野)
 - ・以前にも画像処理系の製作経験
 - ・ソフトウェア(頭脳の勝負)
- ・今回、画像処理+MCRに取り組んだ
2008年11月頃～2009年2月7日(大会出場まで)
結果:優勝

優勝ロボット外観

3

- 名前: 未来2009
- 特徴: 安定性
- 実績: 第2回電通大杯 MCR大会
 - ・画像処理の部 優勝



開発スタンス・テーマ

4

短期間で大会出場できるようにする

※第2回電通大杯の大会ルール(走行距離で審議)

□ ソフトで工夫

- ・画像取得に特殊なハードは使わない
- ・処理アルゴリズムの吟味
- ・ラインの識別
- ・姿勢制御

ハードウェア構成1

5

- 車体

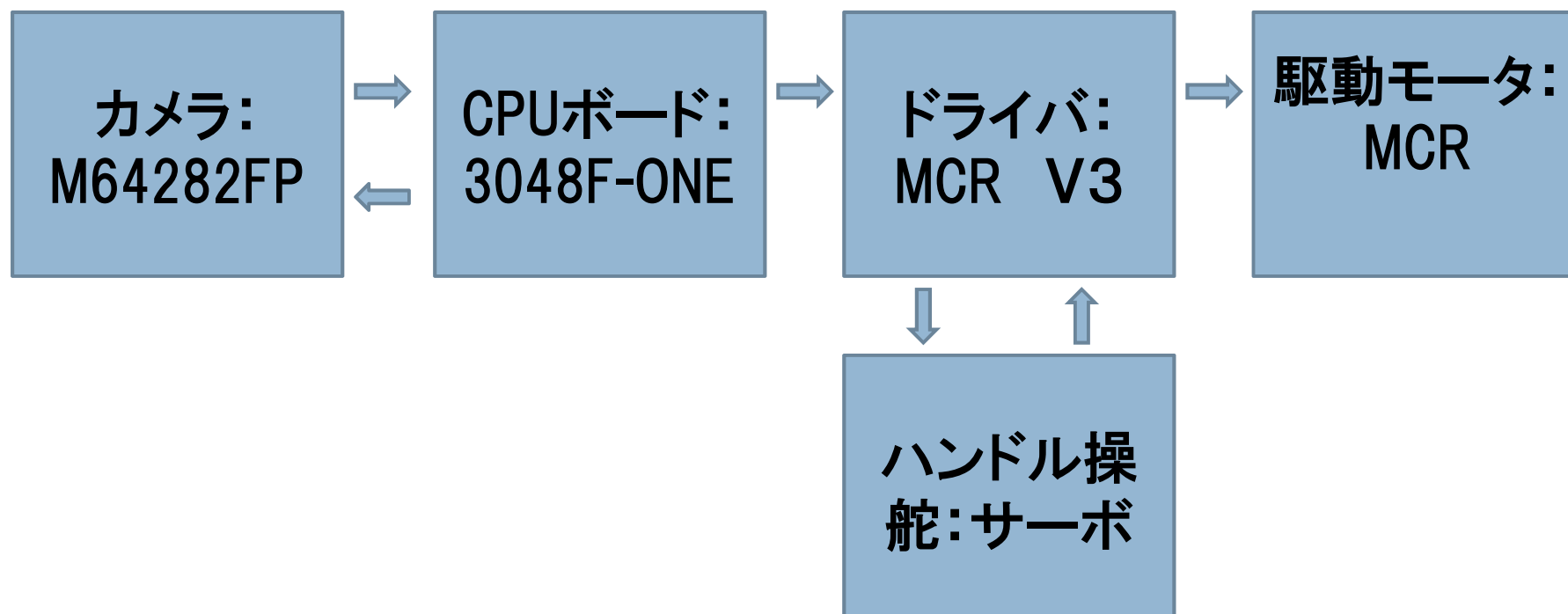
ベース:マイコンカーキットVer.3

カメラ:GBポケットカメラ-三菱人工網膜チップM64282FP

CPU:RY3048F-ONEtypeH

開発環境:C言語(HEW),Visual C++ 2008

- システムブロック



ハードウェア構成2

6

□ カメラについて

- 画像演算とアナログ調整を内蔵している
 - 128×128画素のモノクロCMOSカメラ
 - 人工網膜の、情報圧縮を保有
 - 5V単一電源で動作
 - 低消費電力
 - 高フレームレート(Max 16.4ms/フレーム)
 - ポジネガ画像出力切り替え
 - 輪郭強調
 - 輪郭抽出画像出力
 - ゲイン・レベル調整
 - 光学サイズ 1/4インチフォーマット
 - 画像サイズ 3.07×3.07mm 画素サイズ 24μm×24μm
-
- 現時点で安価に入手できる(300円程度)



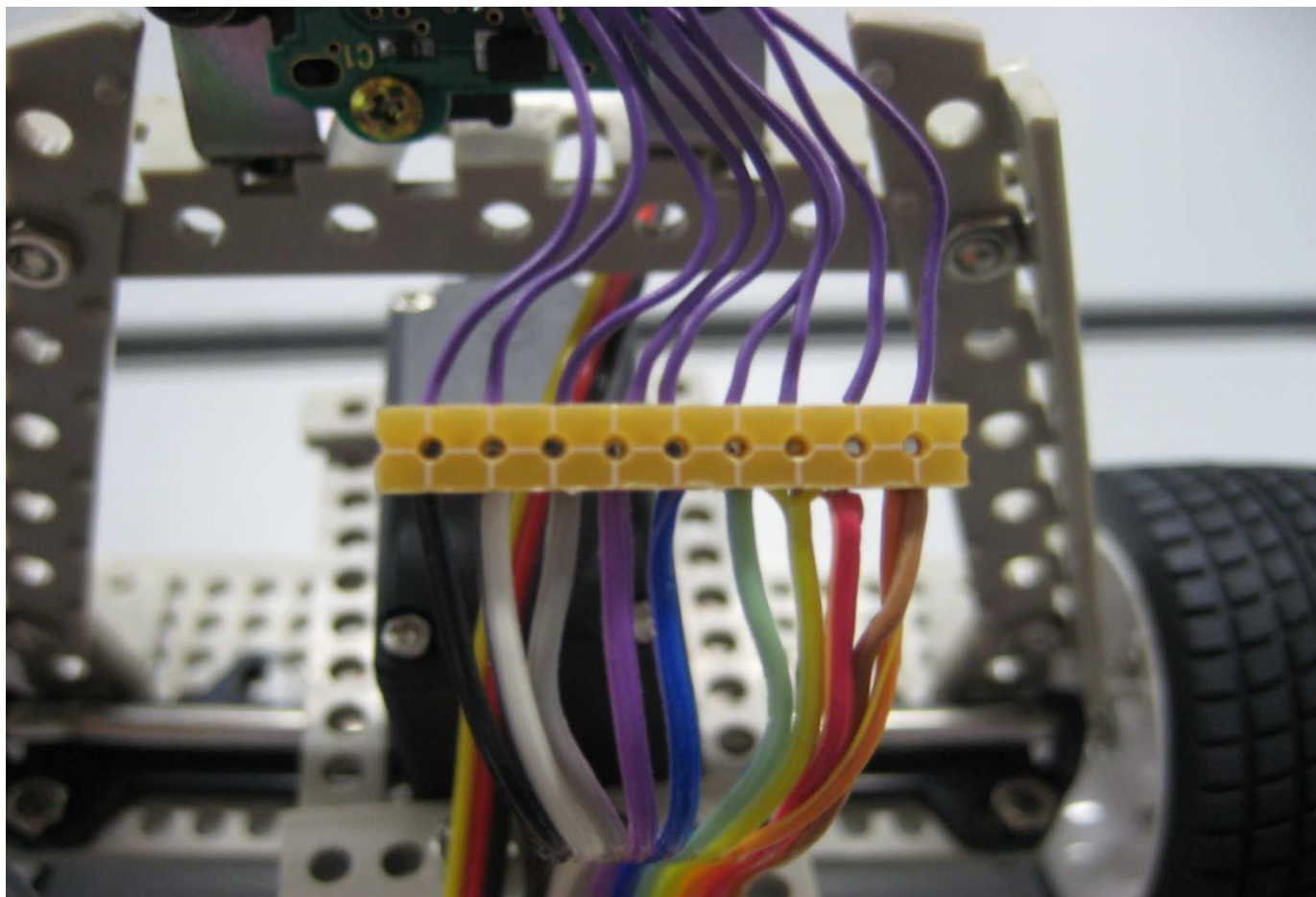
カメラ-CPUピンアサイン

7

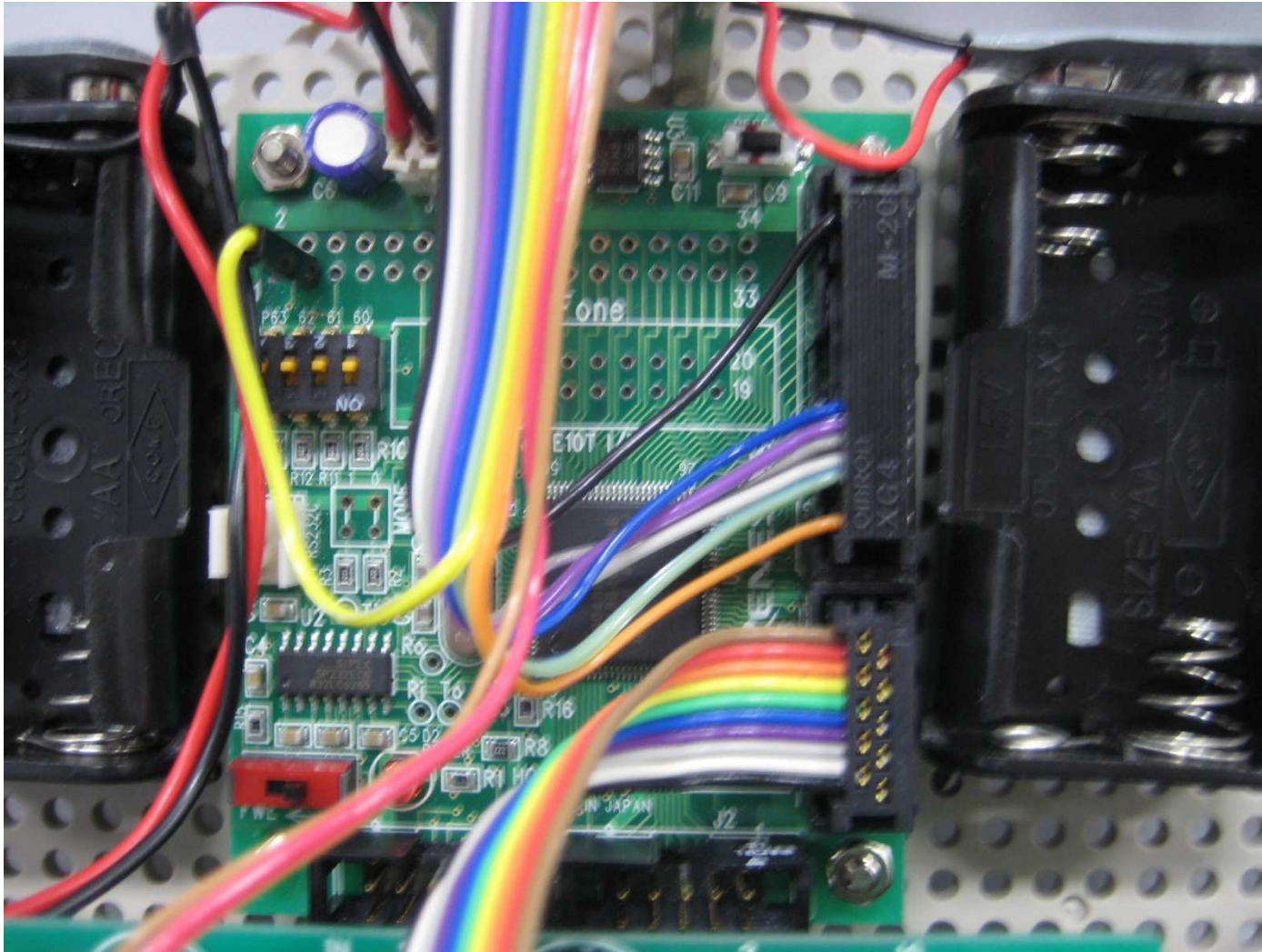
信号名	カメラのCN	ケーブル色	CPUのCN	備考
GND	1	茶	J1-10	カメラ電源-(0V)
Vout	2	赤	J1-9	1画素の電圧出力
GND	1	橙	J6-20	
READ	3	黄	J8-3	
XCK	4	緑	J6-17	SioのCLK端子(シャッター)
XRST RESET	5	青	J6-13	
LOAD	6	紫	J6-14	
SIN	7	灰	J6-15	
START	8	白	J6-16	
VDD	9	黒	J6-1	カメラの電源+(5V)

配線

8



右から黒、白、灰、紫、青、緑、黄、橙、赤、茶



ソフトウェア-画像処理概論

10

- ・「未来2009」画像処理MCRで用いた手法含む
- 1. 画像入力
- 2. 前処理- 濃度・補正・幾何学的歪み補正・ノイズ除去・濃度ヒストグラム・2値化
- 3. 中間処理- 関心領域の抽出(セグメンテーション)
 - 分析ポイント・マトリックス・ラベリング・エッジ検出・領域分割・重心検出
- 4. 画像特徴計測- ライン・色・形状・テクスチャ
- 5. アプリケーション(MCR、ロボット)

ソフトウェア-画像処理1

11

1.画像入力

(ロボットに搭載カメラ-コース路面という環境とする)

- ・画像を読む為の専用GPUは用いていない
- ・ソフトからクロック与えて、カメラからの出力電圧をA/D変換し読みこんでいる

(1クロック(XCK)を送る、次の画素の明度が電圧出力(VOUT)される)



ソフトウェア-画像処理2

12

- 撮影し画面を作る 図1(ロボットの目 図2)
- これから白線を見つけていく

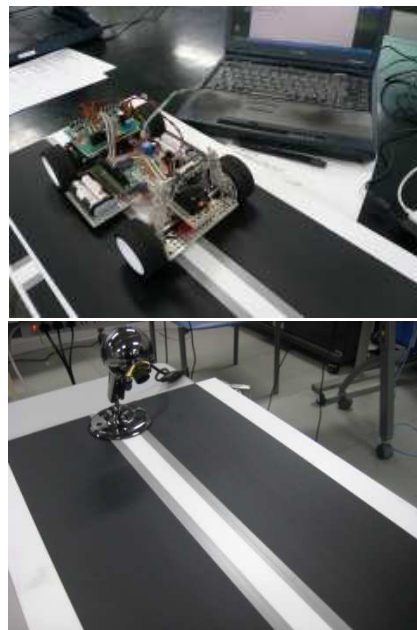
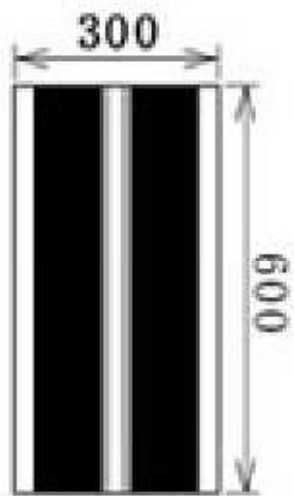


図.1 コース(路面)を撮影

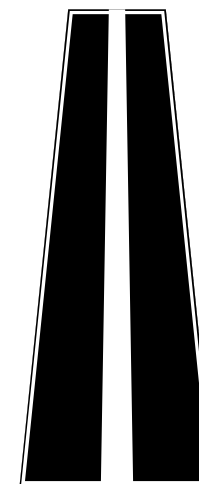


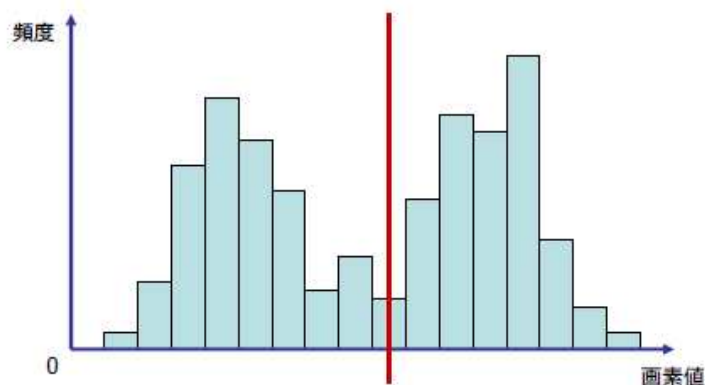
図.2 ロボット

ソフトウェア-画像処理3

13

2.前処理

- 2値化(明るさヒストグラム)
- 明るさ基準
 - ・閾値-モード法
- スライス画面を結合
- 倍率を計算し正規解像度での画像を作る



ヒストグラムが双峰性の場合、峰の谷間となる値を閾値とする。

分析



ソフトウェア-画像処理4

14

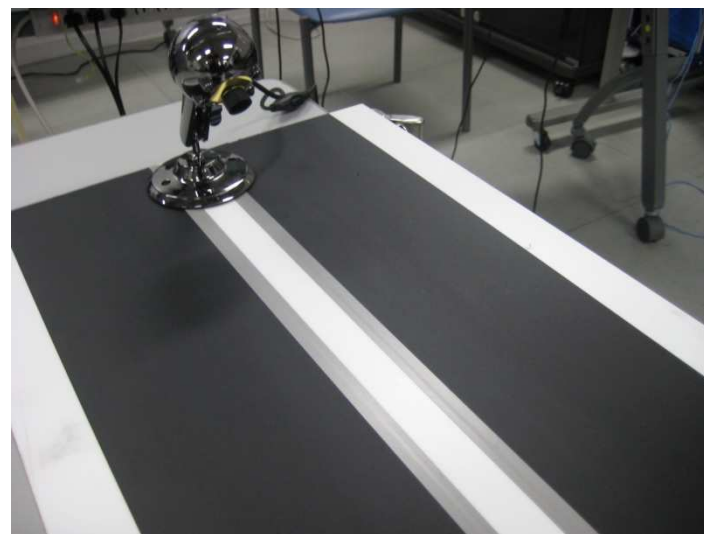
- 2値化について
 - ・2値化とは各画素の明るさを一定の基準値により2つの値(0,1)に変換する処理のこと
- 閾値を元に路面とラインを判別する

路面(黒)とライン(白)の濃度差が大きい

2値化の実験

15

- WEBカメラとパソコンで2値化してみる
 - ・同じ考え方
 - カメラ-パソコン



2值化表

16

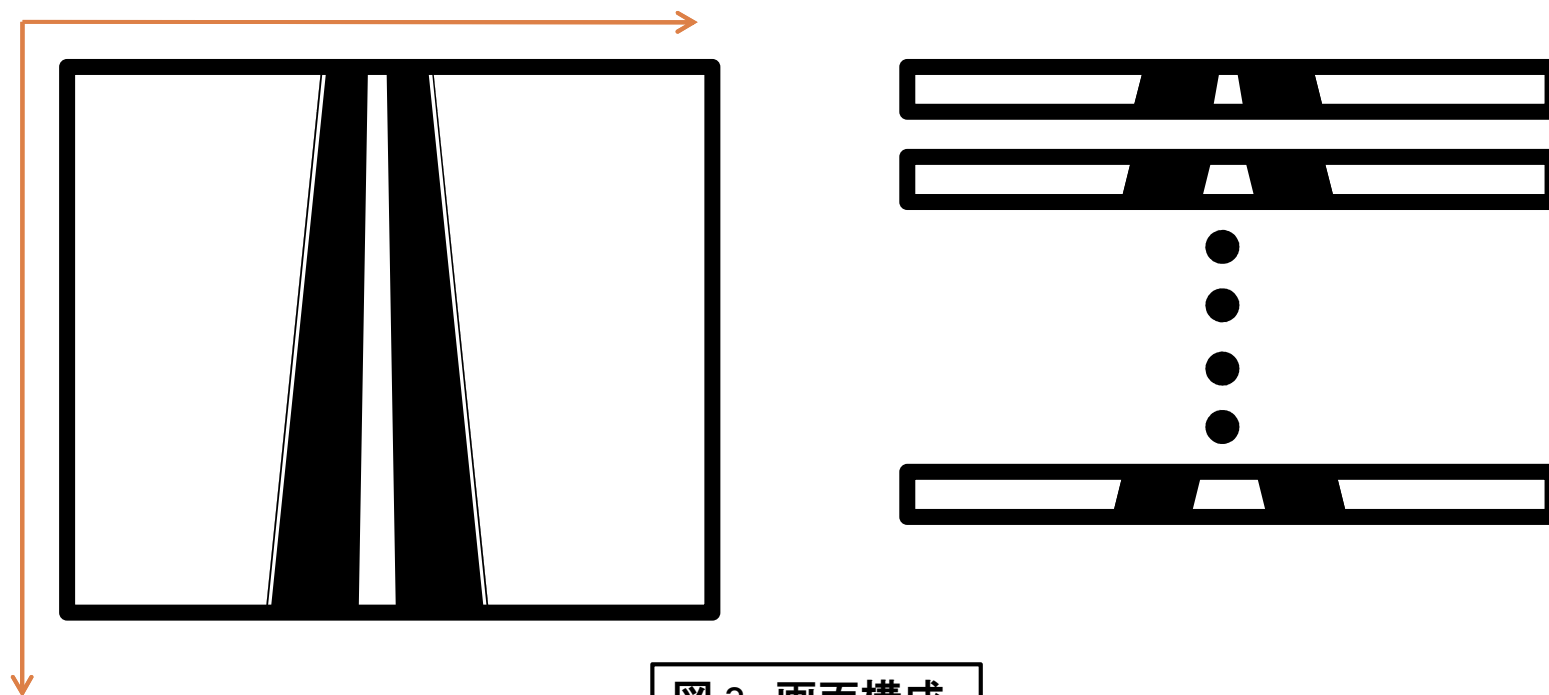
																		0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	
																			0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0
																			0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0
																			0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0
																			0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0
																			0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0
																			0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0
																			0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0
																			0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0
																			0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0
																			0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0
																			0	0	1	1	1	1	1	1	1	1	1	1	0	0	0	0
																			0	0	1	1	1	1	1	1	1	1	1	1	0	0	0	0
																			0	0	1	1	1	1	1	1	1	1	1	1	0	0	0	0

ソフトウェア-画像処理5

17

- 路面認識のみのため処理負担を考えて、このロボットでは1画面のスライスをいくつか作る 図3

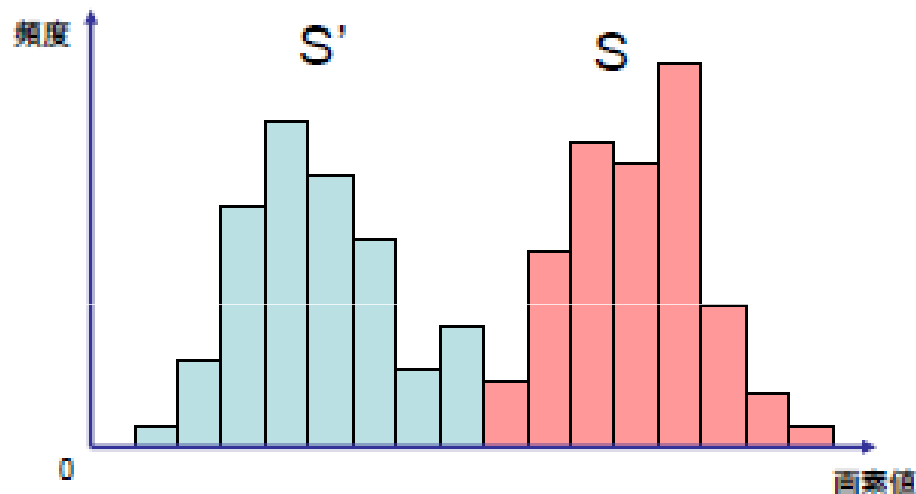
画素のカウント数



ソフトウェア-画像処理

18

□ 閾値-Pタイル法



- ・あらかじめ対象物の画面内での割合 p がわかっている場合、閾値で分けられたヒストグラムの累積度数が $p = S / (S + S')$ を満たすように閾値を決める

ソフトウェア-画像処理6

19

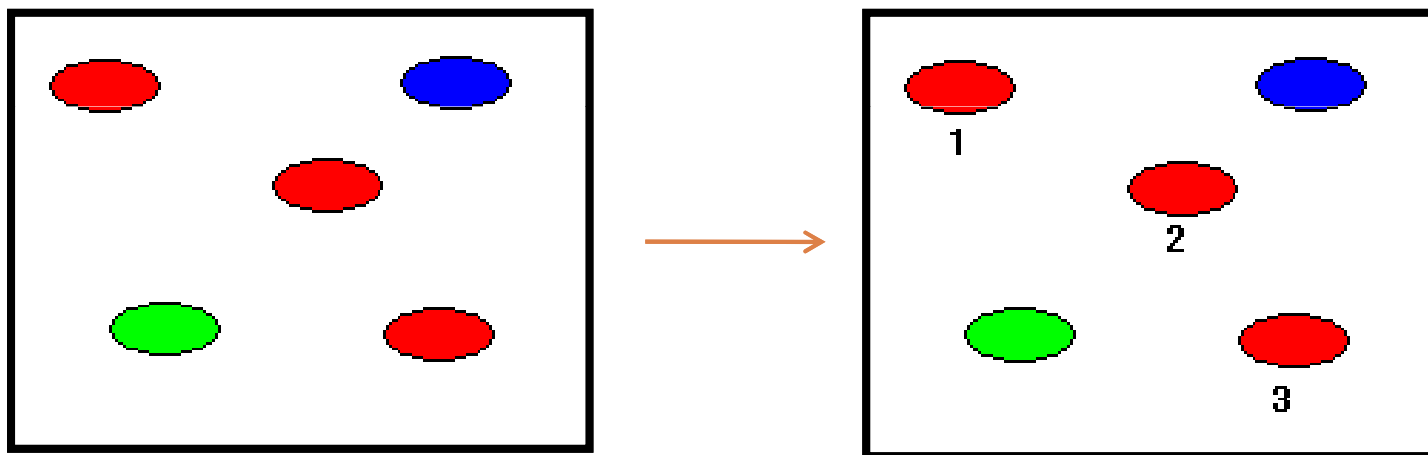
3.中間処理

- データ化
- ラベリング処理
- 領域分割
 - ・主にライン部分と路面

ソフトウェア-画像処理7

20

□ ラベリングイメージ



赤い領域に対して1.2.3とラベルを付けた

ソフトウェア-画像処理10

23

- 隣り合う画素の塊をブロックとしてとらえて、上下のブロックの最大値,最小値の位置関係で繋がりを調べた

通常のラベリング処理より高速に行える
・外周を探索する方法等

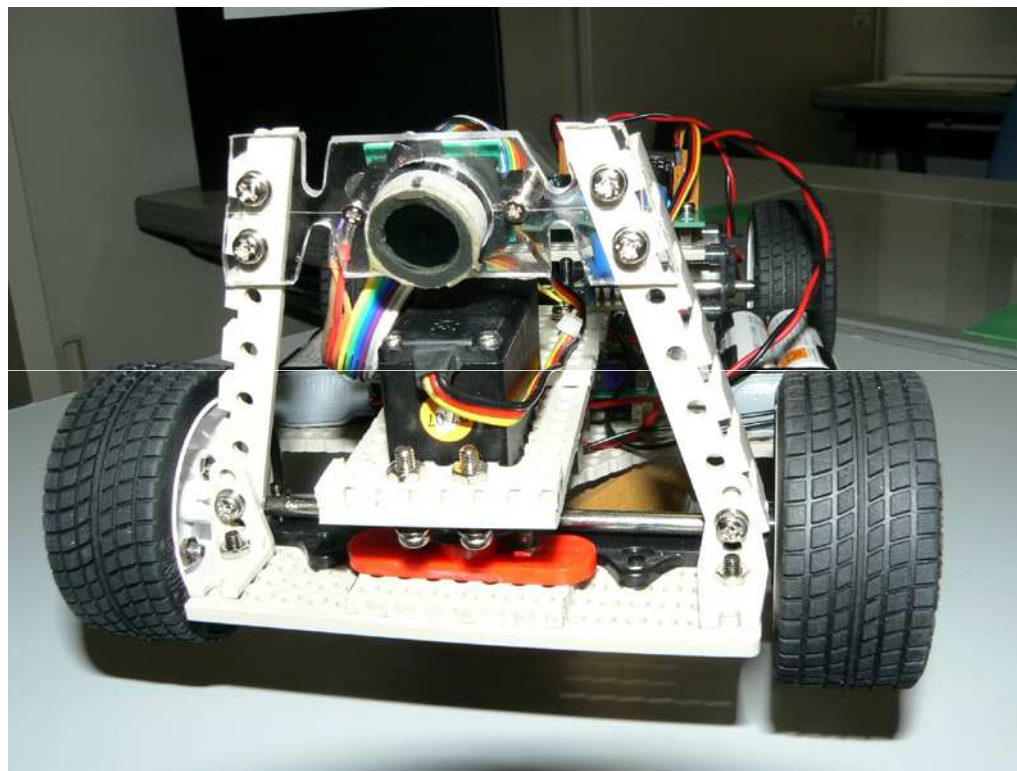
画像処理MCRには適している

ソフトウェア-画像処理11

24

□ ラベリングソフトを考えてみる

```
λ=0;
for(y=0;y<yz;y++) {
    for(x=0;x<xz;x++) {
        if(b[y][x]==1) {
            //路面空間にターゲットがn種類あるとしてnを求め
            //付与していく
            //(n=0,1,2) L_1<L_2
            if(n==0) {λ++;g[y][x]=λ; T[λ]=λ;}
            //新しいラベル
            else if(n==1) {g[y][x]=L_1;} //ラベルの延長
            else {g[y][x]=L_1;//小さい方へマージ
            for(l=1;l<=λ;l++) {if(T[l]==L_2) T[l]=L_1;}
        }
    }
}
```

・いよいよ、アプリケーション処理
実コースでの走行方法

コース走行のためのソフト実装

26

- 2点を使った経路の決定
 - ・辿るべきラインの一定間隔に2点(抽出点)を取りその2点を結ぶ直線(パス)を経路とする 図.6

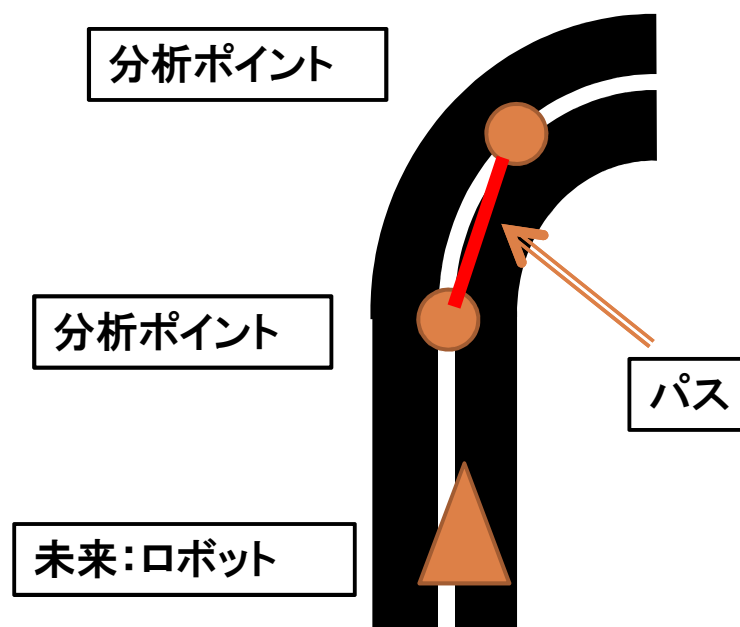
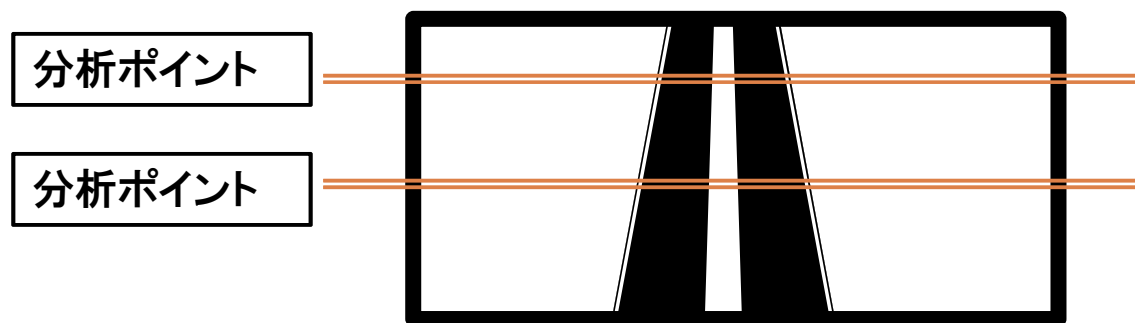


図.6 経路

コース走行のためのソフト実装

27

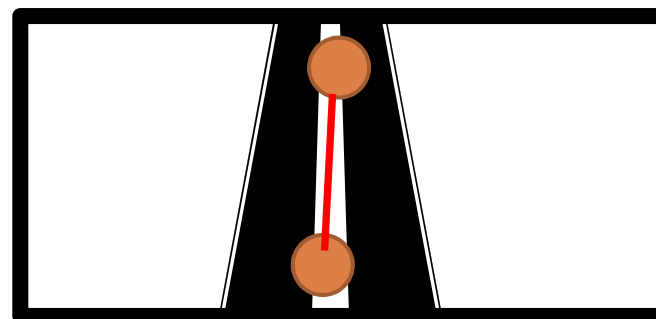
- 正規解像度画像から分析ポイントを決める
 - ・辿るべきラインとの交点から抽出点を得る
 - ・これによってロボットの位置と姿勢角度を比較することでライントレースが可能になる



コース走行のためのソフト実装

29

- 図.6 の2点の定義方法から得た経路が完成する
 - ・この経路を直線・曲線共にトレースする

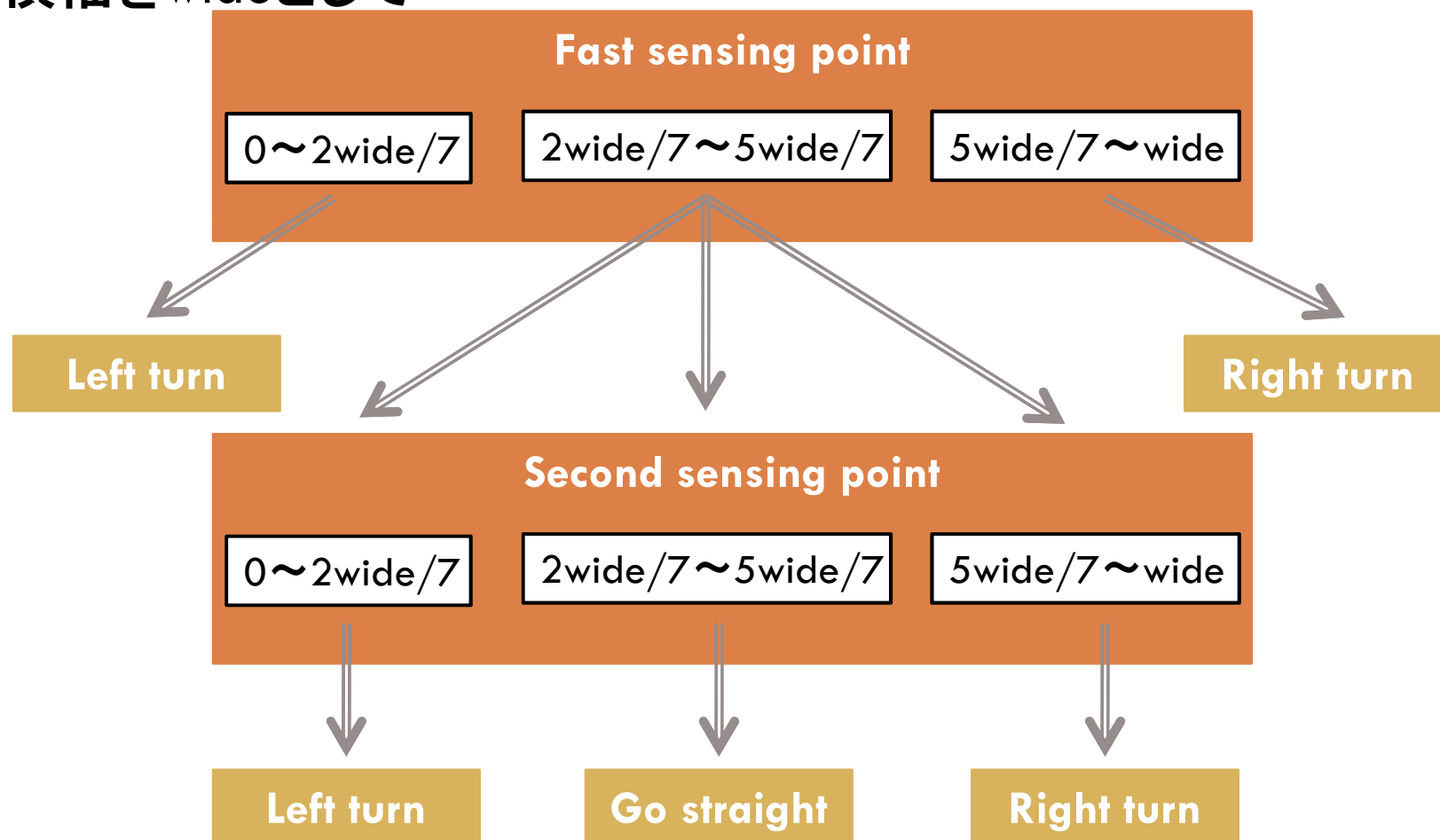


コース走行のためのソフト実装

30

□ 姿勢制御アルゴリズム

▪ 横幅を w として



コース走行のためのソフト実装

31

まとめ

未来のソフト

- 2値化
- バイナリ
- ラベリング
- 抽出点
- 経路作成
- 姿勢制御

まとめ

32



- ・連続的にカーブが続くと制御ポイントの決定が間に合わず脱輪していることもある
- ・姿勢制御角度のコースコースの対応

ストレート直後の障害カーブ

2009記録

33

1位	未来	林雄一氏:大阪電気 通信大学 自由工房	870cm
2位	SCITECS-i	登弘聡氏:神戸市立 科学技術高校	844cm
3位	e-eye-ver.1	芦野広巳氏:山形県 立山形工業高校	630cm
4位	あいちゃん2009	溝上洋三氏:個人	618cm
5位	i君	田辺仁史氏:大阪府 立淀川工科高校	570cm

<http://robot.watch.impress.co.jp/cda/news/2009/02/13/1610.html>

2009,2,7 第2回電通大杯 マイコンカーラリー大会

・画像処理の部

今後

34

- まだ1回目なので細かいバグ取り
- グラフィカルなデバック
- 走行量のフィードバック
- 次期大会も出場

ご静聴ありがとうございました

参考文献

36

- 酒井幸市:デジタル画像処理入門
CQ出版 2005
- 昌達慶仁:画像処理プログラミング
ソフトバンククリエイティブ 2008

