NAIST-IS-MT9751105

修士論文

視覚と環境モデルとのマッチングによるサッカーロボットの位置同定手法に関する研究

望月 恒治

1999年2月12日

奈良先端科学技術大学院大学
情報科学研究科 情報処理学専攻

本論文は奈良先端科学技術大学院大学情報科学研究科に
修士 (工学) 授与の要件として提出した修士論文である。

望月 恒治

指導教官： 西田 豊明 教授
横矢 直和 教授
武田 英明 助教授

視覚と環境モデルとのマッチングによるサッカーロボットの位置同定手法に関する研究*

望月 恒治

内容梗概

ここ 1,2 年、人工知能やロボットの研究者の間で RoboCup が注目を集めている。この “ロボットがサッカーをする” という問題設定は、ナビゲーション、行動生成、位置同定、環境認識などの研究の例題として適しており、その中でも位置同定は重要な位置を占める。本研究では視覚と環境モデルを用いた位置同定手法を提案する。本研究で構築した環境モデルでは、コンピュータ内に準備したサッカーフィールドを用いて、位置情報からロボットの視界画像を作り出すことができる。この環境モデル内で、実ロボットの視界画像に近い視界画像を生成する位置を探索することにより、位置同定を行うことができる。今回用いたロボットはカメラ画像上のゴールの高さと縦横比より推定した位置を保持している。この位置を探索の初期値として、その近傍を山登り法で探索して最適値を求める。本手法を実装し、ロボットの視界画像のみでは 1m ~ 50cm ほどあった誤差が環境モデルを使った探索により 40cm ~ 20cm 程に補正できることを確認した。本手法はエンコーダや非接触センサとは異なり、フィールドの大きさに精度が依存しないこと、壁というランドマークを必要としない点が優れている。

キーワード

ロボット、サッカー、位置同定、探索、情報抽出

---

[*] 奈良先端科学技術大学院大学 情報科学研究科 情報処理学専攻 修士論文, NAIST-IS-MT9751105, 1999 年 2 月 12 日.

A Method of Localization for Soccer Robot Using Vision and Modeling of the Environment*

Koji Mochizuki

Abstract

In recent years, many researchers in AI and Robotics pay attention to RoboCup because realizing RoboCup games needs various techniques in AI and Robotics, such as navigation, behavior generation, localization and environment recognition. Localization is one of the important issues for RoboCup. In this paper, we propose a method of robot's localization using vision and modeling of the environment. The environment model that realizes the robot soccer field in the computer can produce an image of robot's view at any location. In the environment model, the system can search an appropriate location of which view image is similar to the view image by the real robot. Our robot can estimate location from goal's height and aspect ratio on the camera image. We search the most suitable location with hill-climbing algorithm from the estimated location. We programed this method, and tested availability. The error range is reduced from 1m~50cm by robot's estimation to 40cm~20cm by this method. This method is superior to other methods using encoders or sensors because it does not depend on the field size in precision, and does not need walls for landmark.

Keywords:

robot, soccer, localization, search, information discover

---

[*] Master's Thesis, Department of Information Processing, Graduate School of Information Science, Nara Institute of Science and Technology, NAIST-IS-MT9751105, February 12, 1999.

目 次

図目次

表目次

第1章

序論

1.1 はじめに

ここ1,2年、“RoboCup”が人工知能研究者やロボットの研究者の間で注目を集めてきており、雑誌などでも紹介されたりしている([1]、[2])。

注目を集めている理由としては、この“ロボットにサッカーをさせる”という問題が、人工知能の研究例題として、今までの研究例題にはなかった実時間性とマルチエージェントシステムを含んでいることがあげられる。

実時間性という観点から見ると、サッカーにおいて、時にはじっくり考えることもあるが、状況によっては瞬時に判断を下すことができる必要がある。また、マルチエージェントという観点から見ると、複数のサッカーロボットが動的に役割分担を決めて、一つの目的(試合に勝つ)という目標に向かって協力して行く必要がある。

その他にも、この問題はロボットのナビゲーション、行動生成、位置同定、環境認知などのさまざまな問題を含んでいる。

それらの問題に解決のための例題として適している、というのが注目を集めている理由といえる。

1.2 本研究の目的

本研究の目的は、視覚と環境モデルを使用した位置同定手法の提案と、その有効性の検証である。

サッカーをロボットがプレーする場合を考えると、各ロボットが“現在自分が

フィールドのどこにいるのか”を知ってることは、極めて重要なことのひとつである。このことが分かっている場合と、そうでない場合ではサッカーロボットができる仕事が全く違ってしまう。

ロボカップのパリ大会では、中型機部門で優勝したドイツのフライブルグ大学のチームはレーザレンジファインダーとエンコーダを使用して、位置同定を行っていた[6]。このチームの位置同定方法は、

1. 試合開始時に、各ロボットはレーザレンジファインダーをつかって自分のいる位置を計算する。
2. 試合開始後は、各ロボットはエンコーダを使用して自分のいる位置を更新していく。

というものであった。

しかし、この方法には欠点がある。

サッカーをする際には、サッカーというスポーツの持っている性質上、サッカーロボット同士の接触、衝突などを避けることが非常に難しい。このため、エンコーダを使用して位置同定を行っていると必ず位置情報がずれてしまい、位置同定が試合時間が経過していくにつれて難しくなり、サッカーロボットは自分の位置が分からなくなっていってしまう。

この欠点を解消するには、エンコーダを使用しない位置同定の手法を開発することが必要になる。

本研究で使用するサッカーロボットには位置推定を行うモジュールが動作しているが、出力結果において実測値との誤差が1mから50cm程あり、あまり有用であるとはいえない。

そこで、本研究では、サッカーロボットで動作している位置推定のプログラムが誤差を含んだ値を出力する問題を解決するため、ロボットの視覚とサッカーフィールドのモデルを使用した位置の補正を行う方法を提案する。

その誤差を補正するために、サッカーフィールドの環境を持った位置補正のためのモジュールを作成し、実験を行って、提案手法を使うことによってロボットがより正しい位置同定が行えることを検証した。

また、環境モデルにより作成された画像とロボットの画像を比較することにより、画像中の敵ロボットの認識の検証も合わせて行った。

1.3 本論文の構成

本論文は全6章からなる．第2章ではロボットサッカーについて述べる。第3章では本研究で使用したハードウェアと、ソフトウェアについて述べる。第4章では本研究で作成した、視覚と環境モデルを使った位置補正モジュールについて述べる。第5章では本研究で構築したシステムの実験とその考察について述べる。第6章では本研究の結論と展望を述べる。

第2章

ロボットサッカー

2.1 ロボットサッカーとは

ロボットサッカーは、人工知能とロボティクスという2つの研究分野の共通研究課題として、サッカーを例題として選択し、技術的な問題の解決を目的として提唱された。

日本の研究者(ソニーCSLの北野宏明氏、大阪大学の浅田稔氏、電総研の野田五十樹氏らが中心メンバー)が提案し、それに世界中の人工知能やロボティスクの研究者が呼応する形で盛り上がっていった。

研究成果を評価する場所として、1996年のpre-RoboCupを経て1997年8月に名古屋で第1回ロボットワールドカップサッカー大会(以後ロボカップ、もしくはRoboCupと略記する)が開催され[1]、世界中の人工知能やロボティクスの研究者から注目を浴びた。その後1998年7月には第2回ロボカップがワールドカップが行われているフランスで開催された[2]。

第3回のロボカップは1999年7月28日から8月6日まで、スウェーデンのストックホルムで開催される予定になっている。

2.2 ロボットサッカー研究の意味

今までの人工知能・ロボティクス研究では、

- 単一のプログラムが
- 静的な環境で

- じっくりと
- 完全情報を扱う

といったものを研究対象として行われてきた。(例：チェス)

しかしロボットサッカー研究に代表される新しい人工知能・ロボティクス研究では

- 複数のプログラム (マルチエージェント) が
- 動的に変化する環境下において
- 実時間の中で
- 不完全な情報である
- 実データを処理する

という、実世界の複雑さが含まれた環境を研究対象とすることになる。

また、AIの研究の例題としてサッカーを考えた場合、以下のような特徴を持っていると言われている [3]。

- フィールドの状況が逐一変化するため、その変化に追従してプレーの再プランニングを行い、実行する必要がある。
- 外乱などにより受け取る情報が不完全になるので、ミスをする可能性があることを前提としてプレーのプランニングを行う必要がある。
- 複数のプレーヤによるチームプレー (協調) がチームが持つ実力の大きな要素を占める。
- 全てのプレーヤの動きを監督やキャプテンなどが指示することはほとんど不可能であるので、分散制御や自律的に制御をすることが必須となる。
- 一人のプレーヤが得ることのできる情報は限られているので、チームメイトである他のサッカーロボットとの間で情報の共有をする必要がある。

これらの特徴は、これまでの人工知能の研究の例題として長年研究されてきた、積木の問題やチェスにはない物であり、上であげた実世界性やマルチエージェントシステムを考えた場合、欠かすことのできないものである。

また、“ロボットにサッカーをさせよう”、“「人間のワールドカップ優勝チームに勝つロボットチームを作る[2]」”という問題設定は多くの人にとって分かりやすく、娯楽性も含んでいるという特徴も持ち合わせている

これらの特徴を考えると、昔からのAIの標準問題として研究されてきた、チェスのグランドマスターに勝つプログラムを作るという目標が探索手法を中心とした問題解決アルゴリズム研究の発展に多いに貢献したように、ロボットサッカー研究は、現在の情報処理の研究での重要なキーワードである実世界性とマルチエージェントシステム、という問題の解決に適切な例題として位置付けることができる、と考えられる。

第3章

サッカーロボット

本研究で作成したシステムは、奈良先端大のロボットサッカープロジェクト([4]、[5]、[10])で作成しているサッカーロボットの上で動作する。
本研究で使用したサッカーロボットはこのような(図3.1)概観をしている。このサッカーロボットは市販のラジコンをベースに作られており[4]、全体のシステムは大きく分けてハードウェア部分とソフトウェア部分、ソフトウェア部分が動作しているnote-PC部分とに分けることができる。

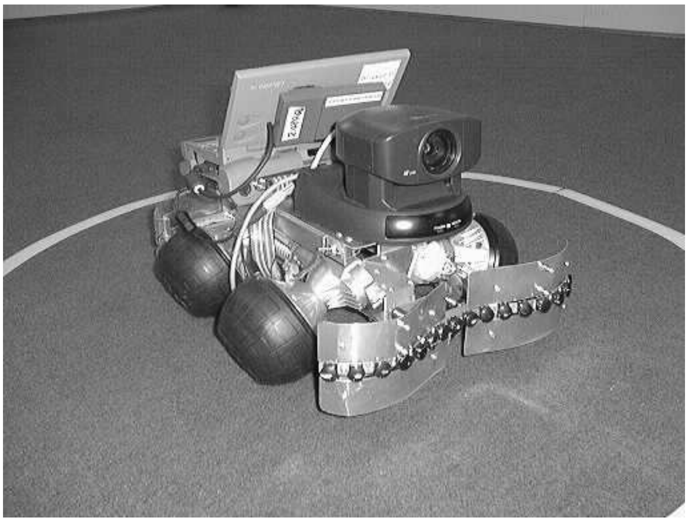


図3.1　本研究で使用したサッカーロボット

3.1 ハードウェア

ハードウェア部分は駆動システム、視覚システム、触覚システム、制御システム、通信システムの5つに分けることができる。

3.1.1 駆動システム

駆動システムは、市販のラジコン自動車の車体部分(モータ、タイヤなど)を流用し、作成されている。このシステムでは左右の駆動系が分離しており、それぞれにモータを搭載してPWS(Powered Wheel Steering)システムを構成している。これにより、その場での回転が可能となっている。

制御システムに接続しており、note-PC上のプログラムから制御システム経由で制御可能である。

3.1.2 視覚システム

この視覚システムで、現在のサッカーロボットの行う知覚処理の中で一番重要である視覚処理を行っている。

視覚センサとしてSonyのEVI D30カメラを使用している。このカメラは高性能なカラーCCDカメラであり、物体の自動追跡機能や運動検出機能が付いている。(本研究ではこれらの機能は使用していない。) また、RS-232Cインターフェースを使ってnote-PCに接続しているので、note-PC上のソフトウェアからカメラのパン、チルトなどを制御することが可能である。

3.1.3 触覚システム

接触システムは他の物体との接触を検出する機能を持っている。サッカーロボットがいるサッカーの環境下では、ロボット同士の接触が頻繁に起こることが容易に想定できるので、このようなセンサを持つことは重要になる。

ロボットは視覚システムを持ってはいるが、視界が限られているために、ロボットの回りでの接触に対して、画像情報のみでは全てを検出することが不可能であ

る。したがって、接触センサをロボットの回りに配置することにより、カメラの視界の外での衝突検出が可能になる。

3.1.4 制御システム

移動のためにモータの制御を行うシステムで、コントローラとしての note-PC と駆動システムをつなげているのがこの制御システムである。PVM をモータのコントロールに組み込むことにより、回転スピードや回転半径の調節が可能になっている。

PVM(Pulse Width Modulation) とは、モータに供給する電圧をパルス (方形波) 化しパルスの ON・OFF の時間の比 (デューティー比) を変えることによって、平均電圧を変化させモータの速度を制御する方法である。

3.1.5 通信システム

それぞれのサッカーロボット間の通信を無線で行うため、PC カードタイプの無線ネットワークカードを使用している。このカードでは 2.4GHz 帯の電波を使用する。

性能としては 2Mbps の伝送速度と見通しの良い環境での 100 数十メートルの通信可能半径を持っている。したがって、ロボットサッカーの環境下で使用するのに適しているといえる。

3.2 ソフトウェア

ソフトウェア部分は

- 画像処理モジュール
- モータ制御モジュール
- カメラコントロールモジュール
- 位置推定モジュール

- 衝突検出モジュール
- 行動生成モジュール
- 通信モジュール

で構成され、これらの各モジュールが共有メモリを介してお互いに情報のやり取りを行っている。

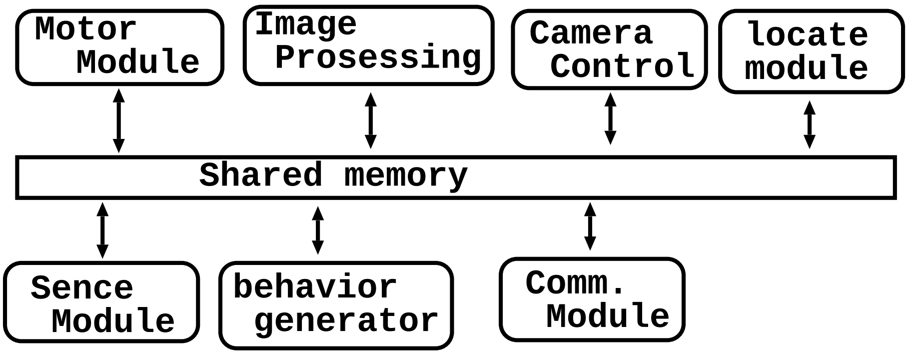


図 3.2 ソフトウェアモジュールの全体図

以下、それぞれのモジュールの説明を行う。

3.2.1 画像処理モジュール：xscc

サッカーロボットがフィールド内でサッカーを行うためには、サッカーロボットは環境(サッカーフィールド)に存在するものを識別する必要がある。サッカーフィールドの中に存在することが分かってるものは、

1. ボール
2. 壁
3. ゴール
4. 白線
5. チームメイトのロボット

6. 敵ロボット

が挙げられる。

これらを区別するために、それぞれのものが持っている色を利用して区別する方法を使用した。

カメラから入力された画像はフルカラーであり、そのままではデータ量が大きいため、処理しにくいデータである。そこで、まずはいくつかの色領域を使って入力された画像を数個の色情報で多値化する。

このモジュールでは色の分類に、多変量解析で使われる“マハラノビス距離: (Mahalanobis'generalized distance)”を用いたクラスタ分布判別法を用いて、カメラからの画像の近似色をまとめる。さらにそれをあらかじめ定められた色によって分類することにより、有限個数(このモジュールでは8つ：黒、白、赤、黄色、青、緑、紫、灰色)の色領域に分けている。

このようにすることにより、画像を多値化してデータ総量の軽減も行なうことができる。

マハラノビス距離による判別分析　各ピクセルの色情報を使って画像中の色の分布度合を調べる。カメラからの画像などの実世界をとらえた画像は、コントラストはあるが、ある色1色に対する明暗であることが多い。ならば近似する色(明るい黄色、暗い黄色)は座標の上でも近い場所に分布すると考え、色座標系(HSV系)においていくつかのグループを形成する(図3.3参照)。

このグループ間に存在する境界線を求めることが、画像における領域分割の境界を求めることになり、それは領域の切り分けにつながる。

本研究では、処理を行う際には実画像をそのまま使わずに、このクラスタ分析によって実画像を8つの色領域に分けたものを使っている(図3.4、3.5参照)。図3.4は処理前の画像であり、図3.5は色領域に分類する処理を行った後の画像である。図3.5の画像を処理に使用している。

本研究で使用した8つの色領域を表3.1に示す。この数値の方を用いて、画像データが共有メモリに収められている。

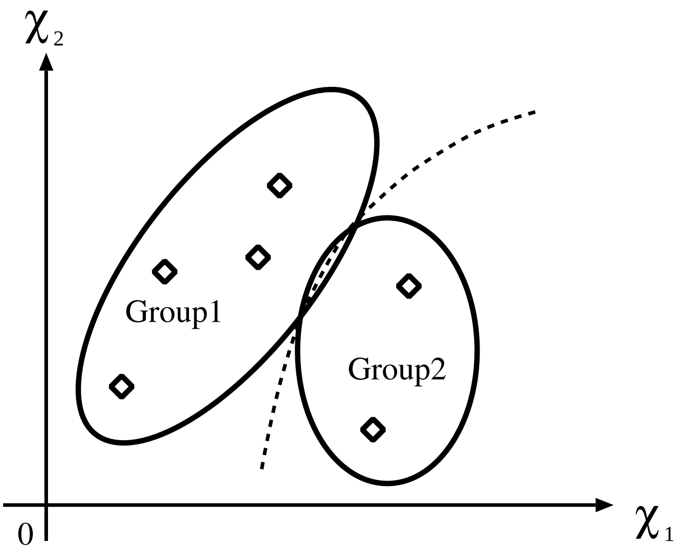


図 3.3　マハラノビス距離を用いたクラスタ分析

図 3.4　実画像

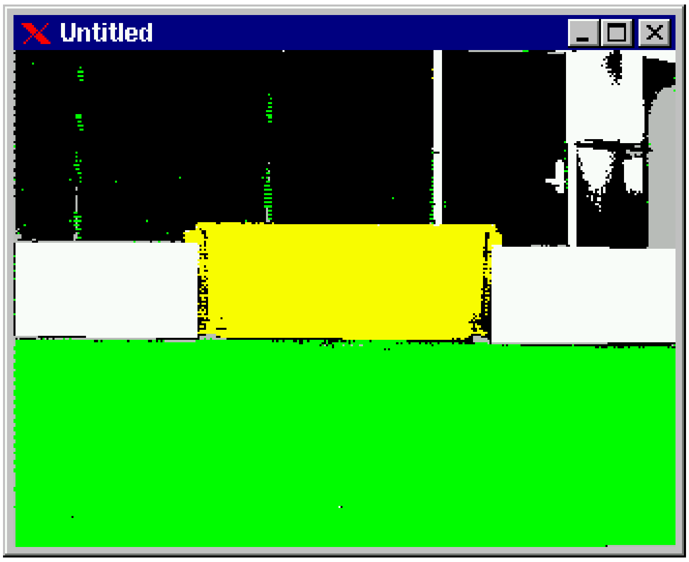


図 3.5　色領域に分けた画像

色	黒	白	赤	黄	青	緑	紫	灰色
数値	0	1	2	3	4	5	6	7

表 3.1　色の数値化

3.2.2 カメラコントロールモジュール：camera

カメラの制御を行うモジュールである。カメラの初期化、パン、チルトなどを行い、その他に

- ボールを探す。
- ゴールを探す。

などの処理も合わせて行う。

3.2.3 位置推定モジュール：locate

カメラコントロールモジュールと連携して動作し、ゴールが見えている場合にゴールの見え方から自分の位置を推定する。ここで使われてるのは、画像中のゴールの見え方から位置の推定を行う、“観測による位置推定” 手法である。

ただ単純に “自分がどこに居るのか” を求めたいのであれば、各ロボットごとに、それぞれを区別可能なマーキングをつけて、天井からフィールドを撮影するように備え付けたカメラによって可能である。

しかし、

- 人間は自分が今何処にいるかについて、自分の目の高さで得られる情報を主に使って、認識していると考えられる。
- RoboCup のレギュレーション [11] で、中型機部門では “global vision system” は使用を禁止されている。

ことを考えて、本研究では “global vision system” は使用していない。

したがって、本研究では “観測による位置同定” を行っている。

観測による位置推定手法について　ここでは、ロボットが行っている位置推定の方式について述べる。

ゴールまでの距離について ゴールまでの距離に関しては、ゴールまでの距離とゴールの高さが反比例していることは容易に想像つくことである。そこで、

係数を求めるため、実際にロボットを環境中に配置し、計測を行い、その結果より係数を算出した。

図 3.6 は、画像中におけるゴールの縦方向のピクセル数と、ゴールからの距離 d をグラフにしたものである。縦軸がゴールまでの距離 (単位はセンチメートル) で、横軸が画像上のゴールの縦方向の大きさ (ピクセル数) である。

また、プロット値のうち色の濃い方 (ラベルが “measure” となっている) は実際にロボットを実環境中に配置して、その距離での画像中のゴールポストが占めるピクセル数を求めたものであり、プロット値のうち薄い方 (ラベルが “theory” となっている) は実測値から求めた理論値である。

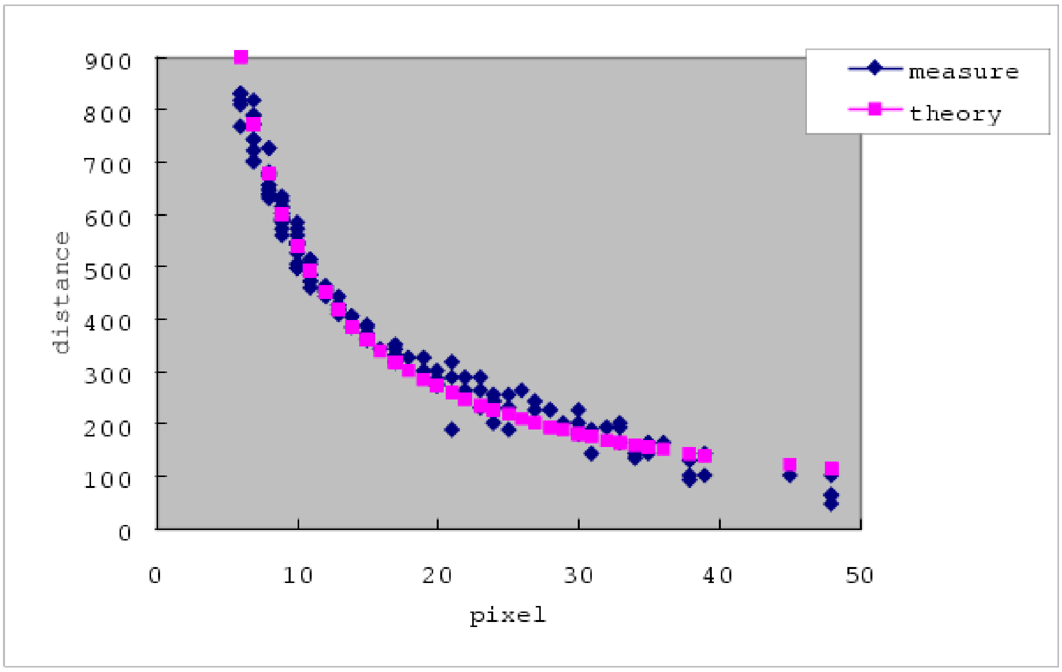


図 3.6 実測値と理論値のグラフ

図 3.6 のグラフより式 3.1 を算出した。

$$d = \frac{5391}{pixel} \tag{3.1}$$

したがって、位置推定モジュールは、図3.6をもとにした近似式、式3.1を使用して、ゴールまでの距離dを算出する。

ゴールに対する角度について 図3.7は、画像中のゴールの縦横比(アスペクト比)と、ロボットが向いている角度をグラフにしたものである。角度0度はロボットがゴールに対して向き合っている状態を表し、角度90度というのはゴールが見えていない状態を示している。

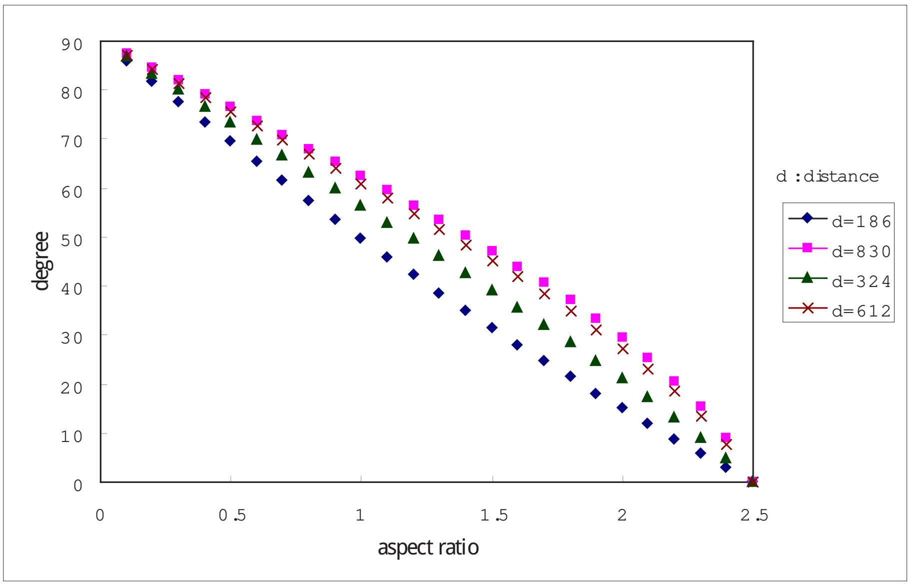


図 3.7 角度とアスペクト比の関係

角度の算出は16ページの式3.2から式3.9までで示した方法で行っている。なお、式の中で使われている記号は図3.8に示している。また、gはゴールの幅であり、定数値300である。mは画像中のゴールの幅であり、$m = ((w')/(h'))h$で求めることができる。$(w')/(h')$はゴールの縦対横のアスペクト比であり、hは見かけのゴールの高さである。

$$\tan(\psi) = \frac{m}{d} \tag{3.2}$$

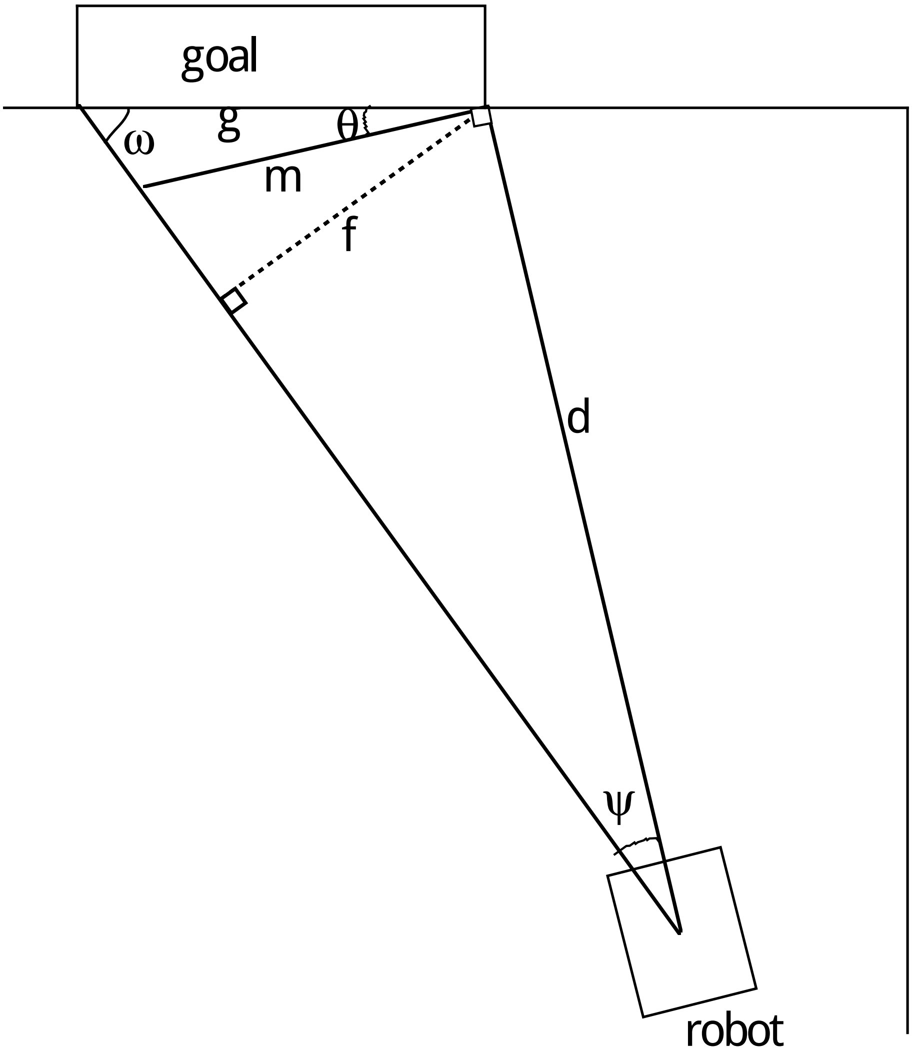


図 3.8 角度の算出

$$\sin(\psi) = \frac{f}{d} \tag{3.3}$$

$$\sin(\omega) = \frac{f}{g} \tag{3.4}$$

$$\theta = 90 - \omega - \psi \tag{3.5}$$

とする。3.2 より、

$$\psi = \arctan(\frac{m}{d}) \tag{3.6}$$

3.3、3.4 より、

$$\omega = \arcsin(\frac{d\sin(\psi)}{g}) \tag{3.7}$$

3.6、3.7 より、

$$\omega = \arcsin(\frac{d}{g}\sin(\arctan(\frac{m}{d}))) \tag{3.8}$$

従って、

$$\theta = 90 - \arcsin(\frac{d}{g}\sin(\arctan(\frac{m}{d}))) - \arctan(\frac{m}{d}) \tag{3.9}$$

となる。この様にして、ロボットの向いている角度 θ を求める。

この様にして求めた d と θ を使い、ロボットのいる座標 (x, y) を求める (図 3.9 参照)。

$$\begin{cases} x &= 4110 - d\cos(\theta) \\ y &= d\sin(\theta) \end{cases}$$

この様にして、ロボットは自分のいるサッカーフィールド上の座標を、画像上の情報から推定する。

3.2.4 衝突検出モジュール：tactile

ロボットの前の部分と、後ろの部分 (車のバンパーにあたる部分) に付いている、触覚システムからの値を受け取り、バンパーのどの部分が接触したのかを判断するモジュールである。

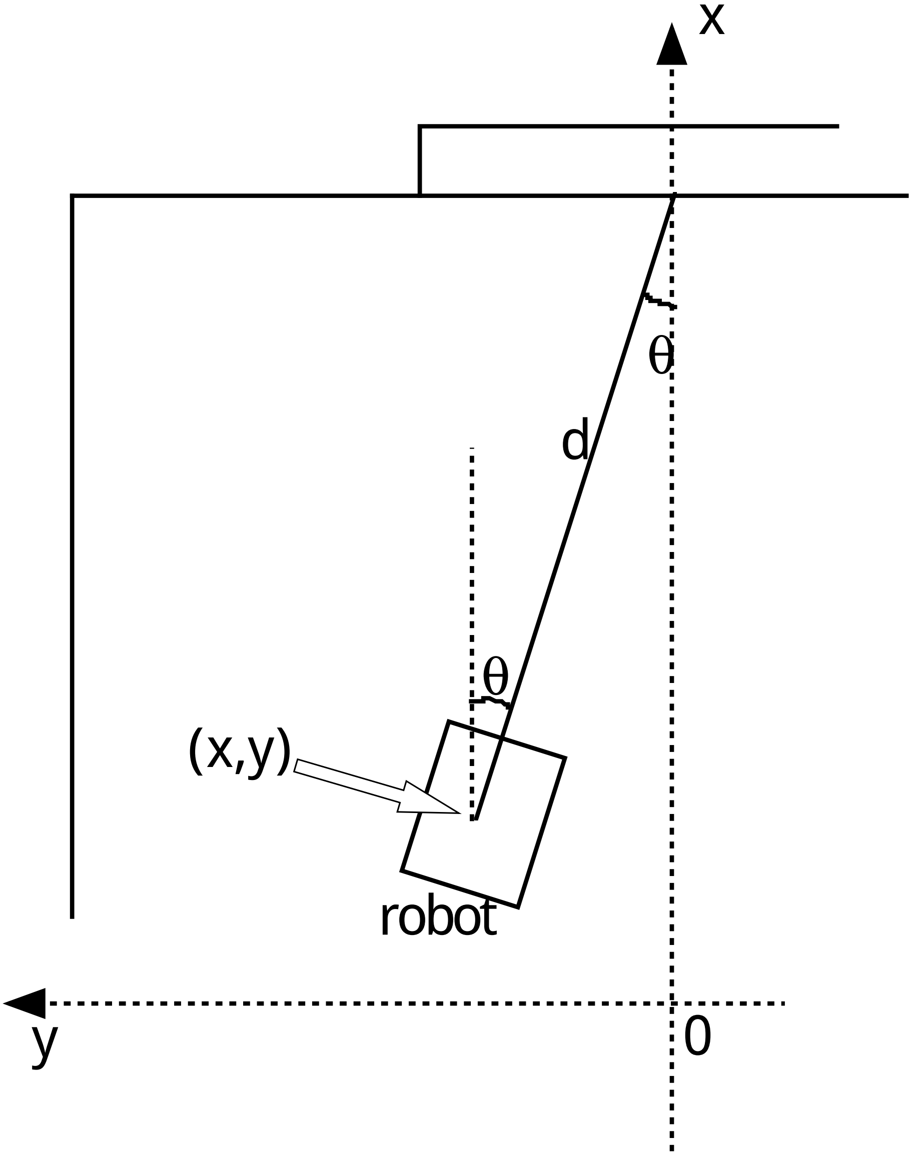


図 3.9 ロボットの座標

状況	行動
ボールとゴールが両方見えてない場合	ボールを探す
ボールが見えているが、大きさが小さい場合	ボールに近付く
ボールが見えてはいるが、ゴールが見えない場合	ボールの周りを回る
ボールとゴールが両方見えている	シュートをする

表 3.2 行動生成パターン

3.2.5 行動生成モジュール：behavior

カメラコントロールモジュール、画像処理モジュール、衝突検出モジュールからの情報を元にして、次に行う行動を生成するモジュールである。

行動生成のパターンは表 3.2 の様になっている。

この他、ロボットによってはデフォルトの行動が異なるので、それに関する行動生成も行っている。例としては、フォワードのポジションに最初付いているロボットは、試合が始まったらボールへのアプローチを最初に行うように行動が生成される。

3.2.6 モータ制御モジュール：motor

行動生成モジュールからの命令を共有メモリ経由で受け取り、駆動システムへモータの制御コマンドを送る、という処理を行うモジュールである。

モータの制御は前方向最大 255 段階、後ろ方向最大 255 段階で制御することが可能だが、現在は前後 16 段階での制御を行っている。

3.2.7 通信モジュール：communication、server

他のロボットとの情報のやり取りに使用する。やり取りする情報は

- 自分の名前
- ボールが見えているか

- ボールからどのくらい離れているのか
- 相手 (自分) 側ゴールが見えているか
- ボールを持っているのか、持っていないのか

である。これらの情報をこのモジュールが共有メモリから読み取り、他のマシンへの送信処理を行う。

また、他のマシンからも同様の情報が送られてくるので、その場合は自分が持っている味方マシンの情報を更新する処理も行う。

3.3 note-PC 部分

上で説明したモジュール群を走らせるコンピュータが必要である。そこでサッカーロボットに note-PC を搭載し、OS として Linux を使い、その上でそれぞれのモジュールを動かしている。

第4章

位置補正モジュール

本研究で使用しているサッカーロボットは、ゴールの観測による位置推定を行っているが、あまりいい精度で位置同定を行うことができない、という欠点を持っている。

そこで、サッカーロボットに搭載されているコンピュータとは別の所 (ネットワークでサッカーロボットに搭載されているコンピュータにアクセスできる場所) に位置補正モジュールを配置し、それに位置の補正を行わせることを考えた。

本研究では、ロボットの位置推定モジュール (第3.2.3節、14ページ) から位置情報を送ってもらい、その値を基にしてコンピュータ内部で作成した“ロボットの視線から見たサッカーフィールド”の画像を作成するモジュールを、新たに作成した。そのモジュールが作成した画像とロボットが実際に見ている画像とを使って位置補正を行わせることを、本研究の提案する位置同定手法とする。

また、ロボットが見ている画像とコンピュータ内部で作成した画像とを比較することによって、ロボットが見ている環境からの情報抽出を行わせることを可能にした。

4.1 位置補正モジュールの構成

位置補正モジュールは、情報をサッカーロボットから送信するクライアントと、メインの処理を行うサーバ、というクライアント/サーバの構成をしている。通信はTCP/IPを使用し、環境内にひとつのサーバを配置して各ロボットで動作しているクライアントとの通信を行っている。

クライアントとサーバの役割は、それぞれ以下のようになっている。

クライアント 各ロボット内部で動作し、それぞれのロボットの内部情報のうち、

サーバ側で必要となる情報をサーバ側に送ることを仕事とする。また、サーバから送られてくるデータの受信も行う。

サーバ サッカーフィールド環境の、ロボットに搭載されているコンピュータとは別のコンピュータで動作し、クライアントから送られてくる情報をもらう。もらった情報をもとにしてそれぞれのロボットの位置補正と、ロボットが見ている画像からの情報抽出を行う。また、位置補正を行ったら、ロボットの方へ補正した位置情報を送ることも行う。

図で説明すると図 4.1 のようになる。また、サッカーロボット全体で動作しているプログラムとの関係を図 4.2 に示す。図 4.2 において、網目模様で囲まれている部分が、本研究で作成したモジュールである。

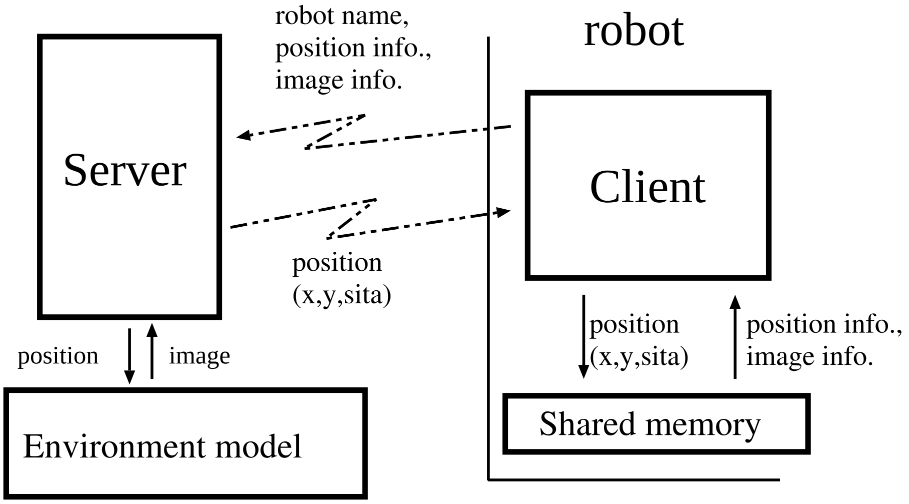


図 4.1　位置補正モジュールの全体図

それぞれ、サーバ部分とクライアント部分について説明する。

4.1.1　クライアント部分

このクライアント部分は、それぞれのサッカーロボット上で動作している。

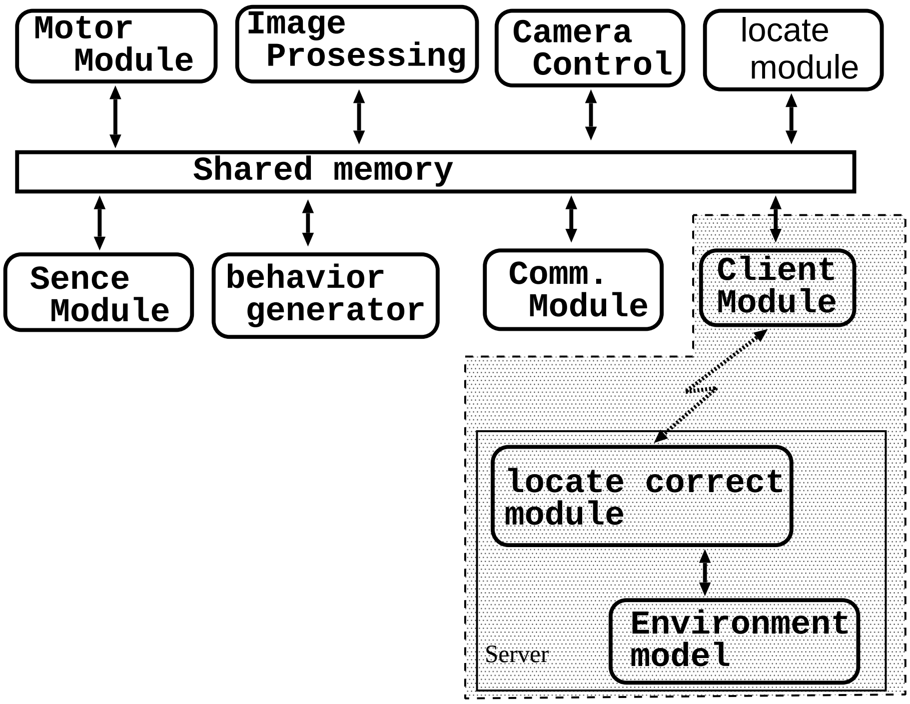


図 4.2 本研究のシステム全体図

このロボット側で動いているプログラムは、ロボットで動作している他のプログラムと同様に、共有メモリを介してデータのやり取りを行い、また、独立したプログラムモジュールとして動作している。

位置情報　位置情報に関しては、他のモジュール(位置推定モジュール)が推定値を共有メモリに格納する。よって、このモジュールのひとつの仕事はこの位置情報をサーバ側へ送ることである。この位置情報は、図4.3で示すフォーマットでサーバへ送られる。サーバ側ではこのフォーマットで送られてきたデータは“位置データである”と認識して、通信で使われたソケット番号をもとにしてロボットの識別を行い、各ロボットの位置情報として格納する。

start	x座標	y座標	角度

図4.3　位置情報のフォーマット

画像情報　画像情報に関しては、画像処理モジュールによって、マハラノビス距離を使ってクラスタ分割されている画像が共有メモリに納められているので、それを取り出してサーバ側へ送る。

画像情報は、図4.4のように2次元配列に色情報が収められているので、このデータをサーバ側へ送信している。

しかし、このデータをそのまま送信したのでは通信に時間が掛かってしまうので、本研究で作成したクライアント部分では、送信する際には1行のデータを送信バッファに整形しながら配置し、行単位で送る方式にしている。このようにすることにより、送信に要する時間と、システム自体に掛かる負荷を軽減している。

また、画像を送信する際には画像データの欠落が起こらないように、クライアントとサーバの双方で同期を取りながら、データをクライアントからサーバへ送信している。

0	0	0	0	0	
1	1	1	3	3	
1	1	1	3	3	
1	1	1	3	3	
5	5	5	5	5	

図 4.4 画像データの格納形式

4.1.2 サーバ部分

サーバ部分は、クライアントが動作しているロボット環境とは別のコンピュータで動かすことを想定して、作成している。

サーバ部分の仕事は、

1. ロボットから送られてきた位置データを、環境モデルを使って作成した画像とのマッチングを行なうことにより補正して、ロボットに送り返すこと。
2. ロボットから送られてきた画像データと、サーバ側で補正した位置情報をもとに環境モデルを使って作成した画像との比較で、情報を抽出すること。

である。

4.2 環境モデル

位置の補正を行うため、サーバ側ではサッカーフィールドの環境をコンピュータ内に持っている。これを環境モデルと本研究では定義する。

フィールドの構築には、OpenGL 互換の Mesa[9] というグラフィックライブラリを使用している。このグラフィックライブラリを使用して、コンピュータ内にサッカーフィールドを構築した [8]。

4.2.1 環境モデルの機能

環境モデルは、

- ロボカップ中型機フィールドと同じ大きさをしている。
- フィールドの大きさは設定値を変えることにより、変更が可能。
- センターサークルの中心にフィールド座標の原点を持つ。
- フィールド内の位置座標 (x,y,θ) を入力として受け取り、その位置での、ロボットの視線から見た画像を作り出すことができる。
- サッカーフィールドには複数 (無制限) の味方ロボットを配置可能。
- 各ロボットの形は自由に変更可能。(本研究では円錐形。)
- (位置情報が分かれば) 敵のロボットを配置可能。
- (位置情報が分かれば) ボールを配置可能。

といった特長、機能を持っている。

4.2.2 環境モデルの属性

ロボカップのレギュレーションで、それぞれのオブジェクト (壁、ゴールなど) の配色が規定されている [11]。

赤 ボール

青 片方側のゴール

黄色 もう片方側のゴール

緑 フィールド表面

白 フィールドに引かれている線と、壁

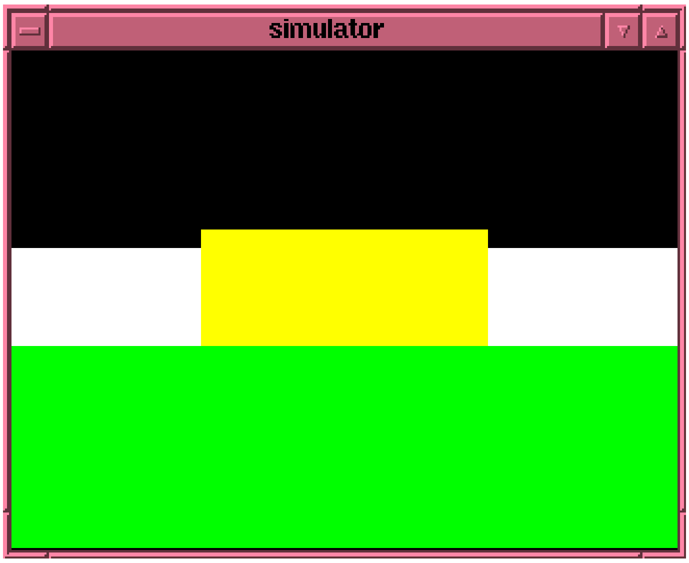


図 4.5　環境モデルで作成した画像

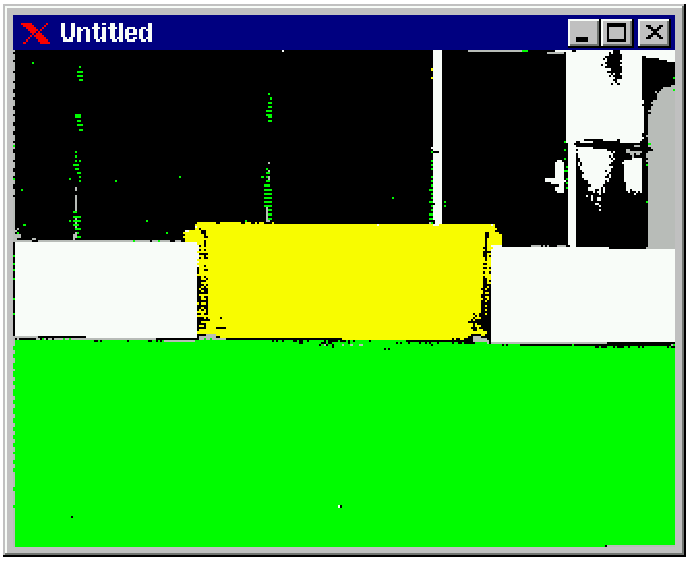


図 4.6　サッカーロボット側の画像

黒 ロボットのボディ

水色 片方のチームのロボットのマーカー

紫 もう片方のチームのロボットのマーカー

環境モデルにおいても、レギュレーションに合わせて各オブジェクトの配色をして、モデルの構築を行った。

4.2.3 環境中のオブジェクト

本研究で作成した環境モデルでは、この環境内に最初から存在するものは、壁、フィールド、ゴールと味方のロボットのみである、とした。この様にした理由として、以下のことをあげることができる。

1. 具体的な情報というのは、ロボットから送られてきた位置のデータのみである。
2. 壁やフィールド、ゴールに関しては固定値の情報を持っているので環境内に配置可能である。
3. その他の情報(ボールの位置、敵ロボットの位置)は、画像から特徴を抽出して初めて得られる情報なので、初期時点では配置することができない。

4.3 位置補正

位置の補正に使うのは、以下の2つのデータである。

- ロボットが見ている画像情報。サッカーロボットで動いているクライアントプログラムから送信されてきたもので、画像処理モジュールによってクラスタ分割されているデータである。

- ロボットの視線から見た理想的なフィールドの画像情報。このデータは、サッカーロボットから送られてきた位置情報を環境モデルのサッカーフィールドに適用して得た結果である。

これらのデータを使い、位置の補正を行う。

位置の補正には、アルゴリズムとして反復改良アルゴリズムを用いる。ロボットから送られてきた位置情報を初期地点として、評価関数による値の最大値を取る地点を、誤差を補正された位置とする。

4.3.1 反復改良アルゴリズム

位置の補正を行うのに、本研究では反復改良アルゴリズムの中で山登り (hill-climbing) 法 [7] を使用した。

このアルゴリズムは最適解を求める際によく使用されるアルゴリズムである。状態を評価したい平面に全て配置し、その平面の高さを、その点の状態を評価した評価関数の値によって決める。この表面を歩き回り、最適解に相当する最も高い頂上を見つけることが目的である。

本研究では、このアルゴリズムに変更を加えて使用している。変更点としては、

1. 現在のサッカーロボットの制御精度との兼ね合いで、評価値が 0.99 を超えた時点で最適解が求まった、として探索を終了するようにした。評価関数については、4.3.2 で述べる。
2. ポンピング現象 (探索において、一度探索を行った点にもう一度戻って来てしまう現象) が起こってしまうと、探索が無限ループに陥ってしまうことがある。そこで、時系列で探索地点を覚えておき、1 つ前に探索した地点に戻ったらそこで探索を終了する。

という所がある。

4.3.2 評価関数

本研究で使用した評価関数を説明する。

評価関数で使用したパラメータ

- 位置補正モジュールが作成した画像上における、ゴールの占めるピクセル

数を a とする。

- サッカーロボットから送られてきた画像上における、ゴールの占めるピクセル数を b とする。
- 2 つの画像 (位置補正モジュールで作成した画像と、ロボットから送られてきた画像) を、ゴールの部分だけ抜き出してから重心を一致させ、ピクセルが一致している (同じ場所で、ピクセル値が一致している) 数を算出し、それを c とする。

図でパラメータの説明をすると、図 4.7 のようになる。これらの a, b, c を使用

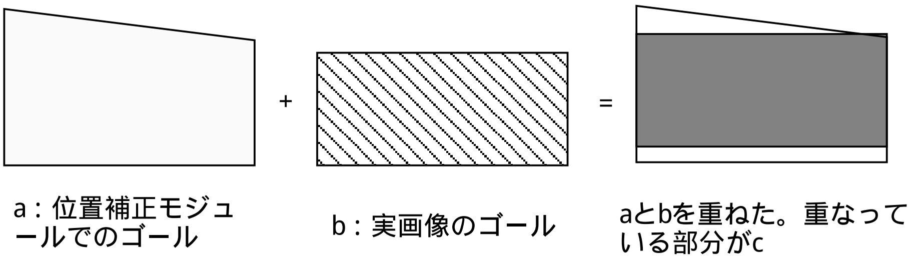


図 4.7 評価関数のパラメータ

して、評価関数 $func(a, b, c)$ を

$$func(a, b, c) = \frac{2c}{a+b} \tag{4.10}$$

と定義した。

この式 4.10 は、

$$func(a, b, c) = \begin{cases} 0 & (c = 0 \text{ の時}) \\ 1 & (a + b = 2c \text{ の時}) \end{cases}$$

となり、それ以外の時は $0 < func(a, b, c) < 1$ となる値をとる。

この評価関数は、

- パラメータ a が b が近い値を示すということは、実際にロボットが位置している点からゴールまでの距離を半径とする円の上に、補正中の値があるということであり、その円の上で、c が最大の値を取る位置が、最適値である。

と考え、作成した。

探索アルゴリズムは、この評価関数として定義した式4.10を使用し、探索を行う。

4.3.3 位置補正のアルゴリズム

探索は、評価関数として式4.10を使い、以下の順序で行われる。

なお、探索の中で使われる“近傍8か所”は、現在地点を a とし、a を中心として点1から8までの各近傍点のことを指している。(図4.8参照)。各近傍点は、a を原点として1辺の長さが初期値20cmの正方形の各点と、正方形が x, y 軸と交差する点を使用してる。なお、正方形の1辺の長さは変更可能である。

1. Step1：探索開始地点 (現在地点) の評価
   (a) 探索開始地点 (現在地点) での、ロボットから送られてきた画像と位置補正モジュールで作成した画像とを、評価関数を使用して評価する。
   (b) その結果を現在地点の評価値として保存する。
2. Step2：次候補の算出
   (a) 現在地点として評価した地点の近傍8か所 (図4.8参照) をピックアップする。
   (b) 近傍8か所からゴールを見た画像を作成する。この際、ゴール方向の角度は座標点 (x, y) からゴールの方向を向くように評価関数内で計算される (図4.9参照)。
   (c) (b) で作成された画像と、ロボットから送られてきている画像を評価関数を使って評価する。
   (d) 8か所のうち、最大の評価値の所を次候補地点の評価値として保存する。
   (e) 8か所のうちの最大の評価値を出した地点の位置情報を次候補地点の位置情報として保存する。

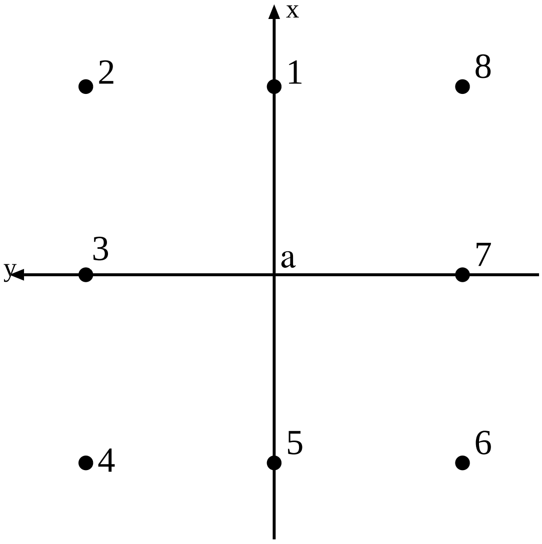


図 4.8 探索時の近傍 8 ヶ所

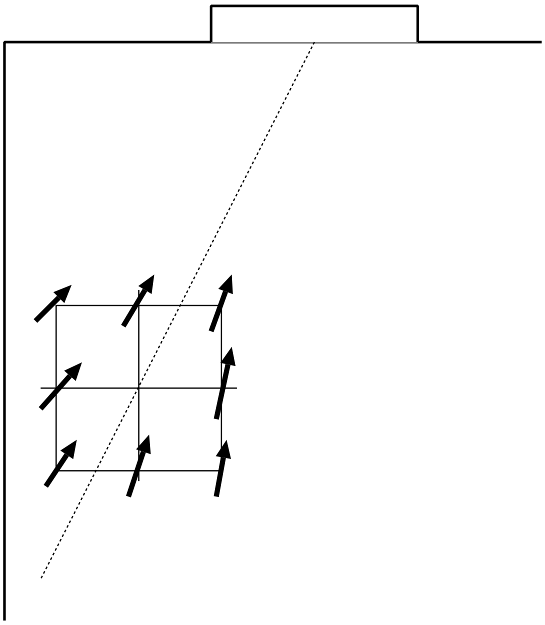


図 4.9 ゴール角度

3. Step3：比較
   - (a) 現在地点での評価結果と、次候補地点での評価結果を比較する。
   - (b) もし現在地点の評価値の方が高い値であれば、現在地点として評価した地点を最大地点として、探索を終了する。
   - (c) もし、現在地点の評価値が 0.99 を超えていれば、探索を終了する。
   - (d) もし、次候補地点が一つ前に探索を行った地点へ戻っている場合は、探索を終了する。
   - (e) もし次候補地点の評価値の方が高い値であれば、その地点へ探索位置を移動する。
   - (f) 次候補地点の位置情報を、現在地点の位置情報として更新する。
   - (g) Step2 へ戻る。

この様にして探索を行い、評価関数が最大の位置を探すことによって、位置の補正を行う。

4.4 画像からの情報抽出

本研究では画像からの情報抽出として、ロボットの視界上の敵の発見を行う。

人間がサッカーをする際に、フィールド上にいるプレイヤーの区別をつけることは重要である。特に、自分の前にいるのが“敵なのか、それとも味方なのか”を認識することは非常に重要である。この認識を間違ってしまうと、敵にボールを渡してしまい、味方をピンチに陥れる状態も考えられる。

サッカーロボットにとっても、敵と味方の区別をつけることは重要な能力である。自分の前にいるのが“敵なのか、それとも味方なのか”の区別をつけることによって、その後のロボットの行動生成にも関係してくることが考えられる。

本研究でとる手法　本研究でとる手法は、環境モデルの

> 環境中には、味方のロボットしか存在しない。

という特長を利用する。つまり、環境中に敵は存在しない、ということである。

この特長と、ロボットから送られてきた、“ロボットが見ている画像”とを組み合わせることにより、敵の発見を行うことができる、と考える。

画像処理モジュールによって色分類が行われるが、その際に画像中のオブジェクトは次のような色分類が行われる。

フィールド　緑色

ゴール 黄色。反対側のゴールは青色

フェンス 白

ボール 赤色

サッカーロボット　灰色

このうち、サッカーロボットを灰色に分類する特徴を使用して、画像中から敵ロボットの認識を行う。処理手順は以下のようになる。

1. 処理には、位置補正モジュールで作成した、位置補正後の画像と、ロボットから受信した画像の2つを使用する。
2. それら2つの画像の、灰色(色番号7)に分類されている画素の数をそれぞれ算出する。
3. その処理結果(画素数が数え上げられている)を比較する。
4. 値の差があらかじめ設定した閾値より大きい場合は、その画像内に敵が存在している、とする。
5. 差が閾値より小さい場合は、その画像内には敵は存在しない、とする。

上記の処理手順により、画像からの情報抽出として敵の認識が可能になる。本研究では、敵がいるか、いないかの閾値を画素数差100と設定した。

考えられる状況としては、表4.1に示した4つが考えられる。それぞれについて、説明する。

状況	味方	敵
A	いない	いない
B	いない	いる
C	いる	いない
D	いる	いる

表 4.1　考えられる状況

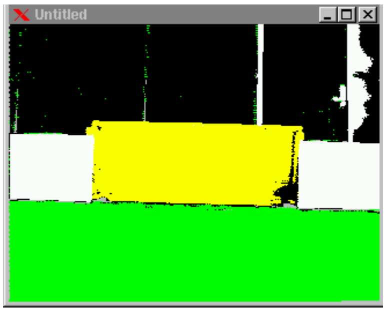


図 4.10　ロボットから見た画像

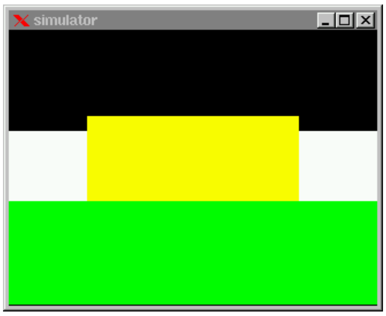


図 4.11　位置補正モジュールでの画像

状況 A(図 4.10、図 4.11)　図 4.10、図 4.11 の場合には、2 つの画像について灰色に分類されている画素数の差は、閾値の範囲内に収まる。よってこの画像(このサッカーロボットの向いている方向)には、敵は存在していない、と処理される。

灰色になるピクセル数を比較してみると、表 4.2 の様になる。表 4.2 の画素数を見てみると、両者の値の差は閾値より小さいことがわかる。

画像	灰色の画素数
ロボットの画像	66 個
位置補正モジュールの画像	0 個

表 4.2　画素数の比較 (状況 A)

状況 B(図 4.12、図 4.13)　図 4.12、図 4.13 の場合、図 4.12 の画像には、視界の左側にロボットが入ってきている。この部分は色分類では灰色になる。それに比べて図 4.13 ではその部分には何も存在していない。この 2 つの画像の、灰色に分類されている画素数を表 4.3 に示す。表 4.3 の画素数の差を見てみると、設定した閾値を超える。閾値を超えて灰色の数の差があると、敵が画面中に存在している、と判断する。

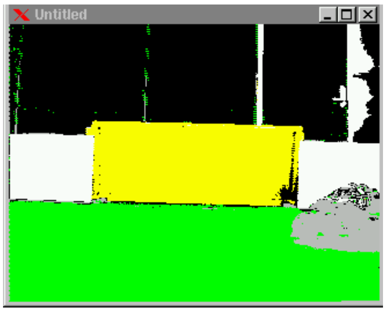


図 4.12　ロボットから見た画像

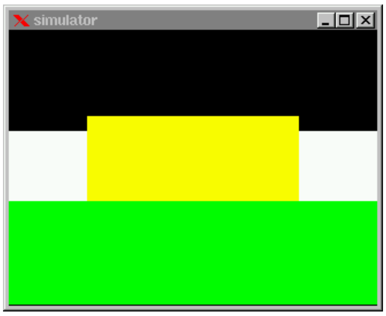


図 4.13　位置補正モジュールでの画像

表 4.2、表 4.3 を見て分かる通り、ロボットのカメラの画面中に敵が入ってきた場合とそうでない場合には画素数に違いが出てくることが分かる。

画像	グレーの画素の数
ロボットの画像	184 個
位置補正モジュールの画像	0 個

表 4.3　画素数の比較 (状況 B)

次の状況 C、状況 D は味方のロボットが存在する場合である。なお、環境モデルでは、味方のロボットは円錐形で表現している。

状況 C(図 4.14、図 4.15)　図 4.14、図 4.15 の場合を考える。

図 4.14 はサッカーロボットが見ている画像であるが、ゴールの左側にロボットが写っている。図 4.15 を見てみると、こちらの画像にも、ゴールの左側にサッカーロボットが写っている。このロボットは味方のロボットであるので、位置情報を送ってきている。よって、位置補正モジュールで生成した画像に存在してるのである。

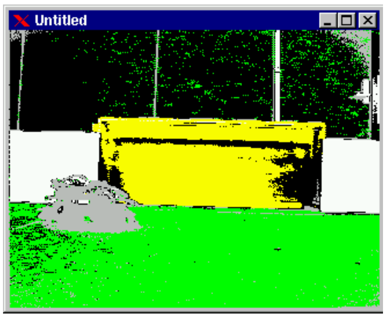


図 4.14　ロボットから見た画像

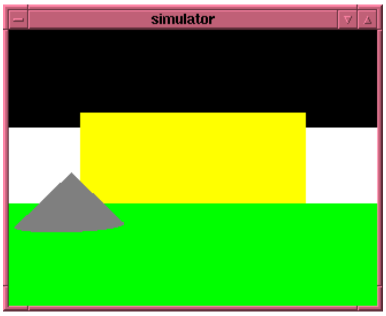


図 4.15　位置補正モジュールでの画像

図 4.14 と図 4.15 の、ロボットが分類されている色の画素数を比較してみると、表 4.4 のようになる。この場合は、両者の値の差が閾値以下であるので、視界に敵はいない、と判断する。

状況 D(図 4.16、図 4.17)　図 4.16、図 4.17 の場合を考える。

画像	画素数
ロボットの画像	155
位置補正モジュールの画像	104

表 4.4　画素数の比較 (状況 C)

図4.16はロボットからの画像であるが、ゴールの左側と右側の両方にロボットが写っている。図4.17は同じ位置からの、位置補正モジュールの方で作成した画像であるが、こちらの画像にはゴールの左側にしか、サッカーロボットが写っていないことが分かる。

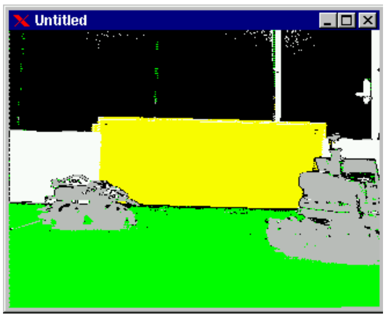


図 4.16　ロボットから見た画像

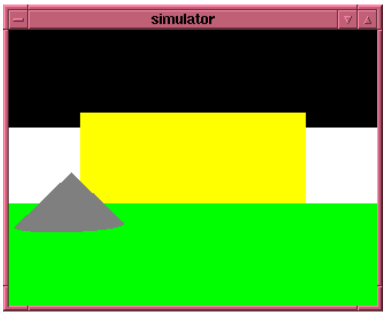


図 4.17　位置補正モジュールでの画像

図4.16と図4.17の、ロボットが分類されている色の画素数を比較してみると、表4.5のようになる。表4.5より、味方のロボットがいる状態で、敵のロボットが入ってくると画素数に変化が現れることが分かる。この場合に両者の値の差を取ってみると閾値を超えるので、この様な場合には視界中に敵がいる、と判断する。

画像	画素数
ロボットの画像	456
位置補正モジュールの画像	104

表 4.5　画素数の比較 (状況 D)

まとめ　状況AからDまでをまとめると、表4.6になる。

状況	画素数(ロボット)	画素数(位置補正モジュール)	敵	味方	敵フラグ
A	66	0	いない	いない	敵いない
B	184	0	いる	いない	敵いる
C	155	104	いない	いる	敵いない
D	456	104	いる	いる	敵いる

表 4.6　各状況と敵フラグ

この表4.6から、敵のいる、いないの列と敵フラグの列のいる、いないの結果が一致していることが分かる。

したがって、色分類された画像と、環境モデルにより作成した画像とを利用することにより、敵のロボットが視界にいるか、いないか、を認識することが可能であることが分かる。

第5章

実験

本研究で提案した、サッカーロボットの位置同定手法の有効性を検証するため、実験を行った。

5.1 実験

位置補正モジュールの有効性を検証する実験を、ロボカップ中型機用フィールド上で行った。

図 5.1 実験

フィールド上にサッカーロボットを実際に置いて、位置同定実験を行った。実験ではロボットの位置推定モジュールにおいて、ロボットの角度θを求めるパラ

データ番号	実ロボットの位置 (x,y)	位置補正前の誤差	位置補正後の誤差
1	(2000.0)	659mm	206mm
2	(1850,0)	639mm	308mm
3	(1300,0)	1113mm	320mm
4	(1150,0)	1111mm	381mm
5	(1000,0)	927mm	335mm
6	(1000,1000)	279mm	260mm
7	(0,1000)	486mm	172mm
8	(-1000,1000)	606mm	243mm
9	(1000,-1000)	341mm	141mm
10	(0,-1000)	258mm	215mm
11	(-1000,-1000)	404mm	295mm

表 5.1　実験：誤差の比較

メータの設定を行った。ページ 18 の式 3.9 において、$d = 328$ と設定した。

ロボットをフィールド上の 11 か所に置き、それぞれの位置で本研究で作成したプログラムを実行し、結果を得た。

実験結果を、

1. 実ロボットを置いた位置
2. ロボットの位置推定プログラムモジュールが出した座標値と、実ロボットの置かれている座標との距離
3. 提案手法を使用して位置補正をかけた後の座標値と、実ロボットの置かれている座標との距離

に関して表 5.1 にまとめた。本実験での、実ロボットの座標系を図 5.2 で示す。図 5.2 上にある、黒丸の場所が、今回の実験でロボットを置いた点である。ロボットが見ているゴールは、図 5.1 で、ロボットの前方にあるゴールである。

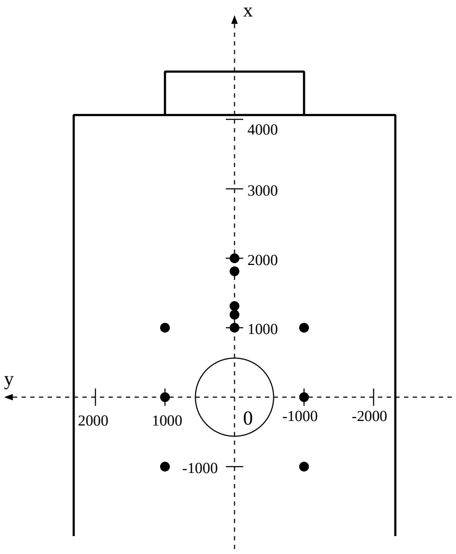


図 5.2 実験で使用した座標系

5.2 考察

表 5.1 より、全般的には実際にロボットが置かれている位置とロボットの位置推定モジュールの出力する値との誤差は平均して 60cm 程あるが、本研究の提案手法で位置補正を行うことによって、その誤差が平均して 26cm にまで少なくなっていることが分かる。

また、表 5.1 のうちデータ番号 1 から 5 を見ると、ゴールからの距離が遠くなるにつれて、ロボットの位置推定モジュールは、だんだん実際の位置と離れた場所を示す傾向が見られるので、誤差が大きくなっている。しかし、本研究で提案した手法で補正を行った後では、その誤差が実際にロボットがいる位置と 30cm ぐらいの幅で抑えられているのが分かる。また、補正後の誤差の数値を追っていくとゴールに近付いていくにつれて、誤差が小さくなっているのが分かる。これは評価関数が、カメラから見たゴールのピクセル数を使用しているので、そのピクセル数が大きくなるにつれて誤差も小さくなっていく、と考えられる。

この、ゴールから離れていくにつれて誤差が大きくなっていく特性は、データ

番号6から8を見ても存在し、ゴールから遠くなっていくにつれて誤差は大きくなっていくが、位置補正をかけることによって実際にロボットが置かれている位置との誤差が平均22cmに抑えられている。

同様にデータ番号9から11にかけても、位置補正をかけることによって21cmに誤差が抑えられていることが分かる。

また、ゴールの一部が味方ロボットに隠された状態を考えた場合、ロボットの位置推定モジュールでは、ゴールの縦横比をパラメータとして使用している影響で、隠されていない状態よりもより誤差が大きくなる。しかし、提案手法では、ゴールの一部を隠しているのが味方ロボットであれば、環境モデル内に味方ロボットが存在し、同じ状況でゴールの一部を隠しているので、環境モデル内と実環境でほぼ同じ状況を作ることができる。したがって、提案手法を適用すればその様な状態でも誤差を減少させることができる。

よって、実験よりロボットに搭載されている位置推定モジュールの誤差を、本研究で提案した手法で補正をかけることができていることが分かる。

位置補正によって、実際にロボットがいる位置との誤差が、平均して半径26cmの円にに抑えることができた。これはレギュレーションによって定められている、ロボカップ中型機の大きさ制限の、最大値にほぼ等しい値であり、実際にロボットがいる位置とロボットが位置同定した結果値との誤差がロボット1台分に匹敵する所まで小さくできたことを示している。

これによって、今まで有用でなかったロボットのフィールド内の位置に関する情報が、ロボットの行動生成のパラメータとして使えるようになる、と考える。このパラメータを使用すれば、行動生成時にポジションの役割分担なども可能になると考えられる。

第6章

結論、展望

6.1 結論

本研究ではロボットの位置同定について、ロボットの視線から見た画像とサッカーフィールドの環境モデルから作成した画像のマッチングを取ることにより、位置の補正を行うことができることを示した。

また、サッカーロボットが見ている画像と、環境モデルでの画像を比較することにより、画像中から情報の抽出ができることも示した。

この提案手法について、1章で研究の目的について述べた。3章で使用したサッカーロボットについての説明を行い、4章で位置補正モジュールの構築について述べ、5章でその有効性について実験と考察を行った。

本研究で提案した手法は、サッカーフィールドという、サッカーロボットがプレーを行う環境をコンピュータ内に作り出し、それを使用して位置の補正を行うことである。そうすることによって、サッカーロボットは環境中で今までと比較して、より正確な位置の同定を行うことができ、それによってサッカーロボットは自分の行動生成のバリエーションを増やすことができる。

また、本研究で提案した手法においては、サッカーロボットはエンコーダの類のものをを一切使用しない。よって、エンコーダを使用しなくても、位置の同定を行うことが可能であることを示してもいる。

6.2 他の方法との比較

本研究で提案した手法以外に、位置の同定を行う方法として以下のものがあげられる。

1. 非接触系センサー (音波ソナー、 赤外線センサーなど) を使用する方法。
2. エンコーダを使用する方法。
3. 画像中の特長点を使う方法。

これらの手法に対しての、本研究の提案手法のメリット、デメリットを考えてみる。

提案手法のメリットとしては、以下の 2 つが挙げられる。

1. フィールドの大きさに依存しない
   位置同定の手法として “現在位置からのゴールの見え方” を使用しているので、ロボットに積まれているカメラでゴールの全景が見えさえすれば位置同定が可能である。よって、フィールドの大きさが大きくなっても提案手法の変更無しにそのまま適応できる。
   非接触センサーを使用する方法では、センサーが届かない大きさのフィールドでは位置同定が不可能になり、またエンコーダを使用する方法では、ロボットがフィールドを動き回るにつれて誤差が積算されていき、位置同定ができなくなっていくという問題がある。
2. フィールドの壁が不要
   現在ロボカップの中型機のフィールドはフィールドの全方位に高さ 50cm の壁が設けてあるが、提案手法では “現在位置からのゴールの見え方” のみを位置同定に使用しているので、同定手法の壁に対する依存が無い。
   非接触センサーを使用する方法では、音波や赤外線などを壁に照射して、その返ってくる時間で壁までの距離を計測しているので、壁が無くなってしまうと計測不可能になる。
   将来的に壁を取り払う方向にルールが改正されていった場合、非接触系センサーを使用している場合は位置同定方法の再構築が必要だが、提案手法ではその必要が無く、壁が無くなってもそのまま適応が可能である。

3. 特長点が不用

   画像中の特長点を利用して位置を測定する方法は、コンピュータビジョンの世界では長年研究されている事である。これらの方法では、対象物の特長点の画像中の座標より、実際のロボットの位置を算出している。これらの方法では、対象物の画像中の座標を正確に求める事が求められるが、ロボットサッカーのような実世界環境では、その特長点を求める事が不可能な状況の方が多く、画像中の座標より位置を求める事ができない可能性が非常に大きいと予想される。

   本研究の提案手法では、その様な特長点を利用していないため、画像に対象物が完全には見えない状態でも位置の同定が可能になっている。

反対にデメリットとしては、

1. 非接触系センサーやエンコーダを使用する方法よりも位置同定に時間がかかる。
2. 視覚に依存している。

ことがあげられる。

6.3 展望

本研究では、ロボットが見ている画像をクライアントモジュールを介してサーバへ通信で送ることによって、位置の補正とロボットの視界上からの情報抽出を可能にすることができた。

本研究で構築したモジュールでは、探索を用いて位置の補正を行っている。サッカーというスポーツでは即座に結果が要求される場合もあり、その様な場合でも正しい位置をサッカーロボットが知るためには、高速の探索を行うことが必要になってくる。高速探索アルゴリズムの開発によって、本研究でも探索時間の短縮が計られ、それにともなって位置補正にかかる時間も短縮できることが期待できる。

また、環境モデルは複数のロボットの、それぞれの位置からの視覚を作成可能である。よって、フィールド上にいる複数のロボットの視覚情報を統合して、敵

の位置やボールの位置の同定や、自分の位置が分からなくなったロボット (視界にゴールが入っていない状態が続くと、位置が分からなくなる) に対する位置情報の提供などを行うことができると期待できる。

謝辞

主指導教官である西田豊明教授には，様々な面で御指導を頂き、また，ロボットを対象とした研究に携わる機会をつくって頂きました。ここに深く感謝致します。

横矢直和教授には，お忙しい中，本研究の副指導教官になって頂きました。ここに深く感謝致します。

武田英明助教授には，研究の副指導教官になって頂き、研究の方向性や内容についてさまざまな御指導、御助言を頂きました。また、本論文をまとめるにあたり、適切な御指導、御助言を頂きました。ここに深く感謝致します。

上野敦志助手には，本研究を進めるにあたり，さまざまな御指導と御助言を頂きました。ここに深く感謝致します。

ロボティクス講座の中村恭之助手には、本研究で作成したプログラムの動作実験のために、実験環境を提供して頂きました。ここに深く感謝致します。

博士後期過程の寺田和憲氏には、RoboCup プロジェクトのロボットに関する情報や、研究を行なうに関してさまざまな助言を頂きました。ここに深く感謝致します。

博士前期過程の片山顕正氏には、研究会などでさまざまな意見を頂きました。ここに深く感謝致します。

研究室のロボット研究グループの皆様には、研究会を通じて様々な意見や助言を頂きました。ここに深く感謝致します。

最後に，研究室での活動を支えて頂きました谷村優香里秘書に厚くお礼を申します。

参考文献

[1] 松原仁."RoboCup への道 第1章"、bit、pp.19–24、Vol.29、No.12、December、1997

[2] 松原仁、北野宏明、浅田稔. "RoboCup への道 第 2 章" bit、pp.25–29、Vol.30,No.11,November,1998

[3] 野田五十樹、 松原仁. "サッカーエージェントの研究"、人工知能学会誌、Vol.11、No.5、pp.694–701、1998

[4] 中村恭之、寺田和憲、柴田篤志、岩瀬昌治, 森本淳, 足立秀和, 武田英明. "ロボットサッカー研究のための安価な視覚移動ロボットの開発"、第3回ロボティクス・シンポジア、pp.33–36、1998

[5] 中村恭之, 寺田和憲, 柴田篤志, 森本淳, 足立秀和, 武田英明. "ロボットサッカー研究のための安価な視覚移動ロボットの開発"、第3回JSME ロボメカ・シンポジア、pages 89–91、1998.

[6] J.-S. Gutmann, W. Hatzack, I. Herrmann, B. Nebel, F. Rittinger, A. Topor, T. Weigel, and B. Welsch, The CS Freiburg Robotic Soccer Team: Reliable Self-Localization, Multirobot Sensor Integration, and Basic Soccer Skills, in: M. Asada, RoboCup-98: Robot Soccer World Cup II, Springer-Verlag, Berlin, Heidelberg, New York, to appear.

[7] Stuart Russell・Peter Norvig 著、古川康一 監訳、"エージェントアプローチ人工知能"、共立出版

[8] OpenGL ARB、OpenGL Programming Guide(日本語版)、アジソン ウェスレイ

[9] `http://www.mesa3d.org`

[10] `http://robotics.aist-nara.ac.jp/~takayuki/robocup-naist.html`

[11] `http://www.robocup.org`

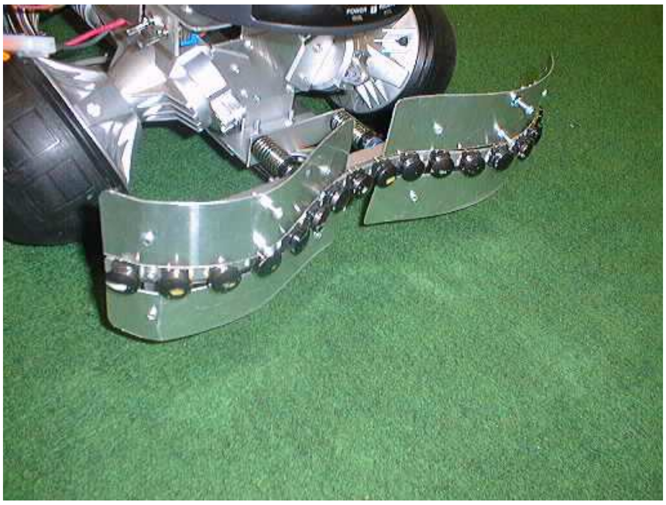
付録

1. RoboCup パリ大会のレギュレーション
    - (a) フィールドのサイズ
フィールドの大きさは 4575mm x 8220mm と決められている。この大きさは卓球の台を縦横に 3 台ずつ並べた大きさに等しい。
    - (b) 壁フィールドのまわりには高さ 500mm の壁をゴールの部分を除いた所に立て、フィールドを囲む。
    - (c) コーナーフィールドの角は、三角柱のブロックを配置し、ボールが角でスタックしにくい形状になっている。
    - (d) ゴールゴールの幅は 1500mm で、高さは 500mm とする。 各々のゴールは単色で塗られ、またその色は異なる色とする。今大会では片方のゴールを黄色、もう一つのゴールを青で塗る事とする。
    - (e) ボール使用するボールは FIFA が定めた 4 号サッカーボールを使用する。ボールは単色で塗られ、その色は赤とする。
    - (f) 色各オブジェクトは以下の色で塗られる。

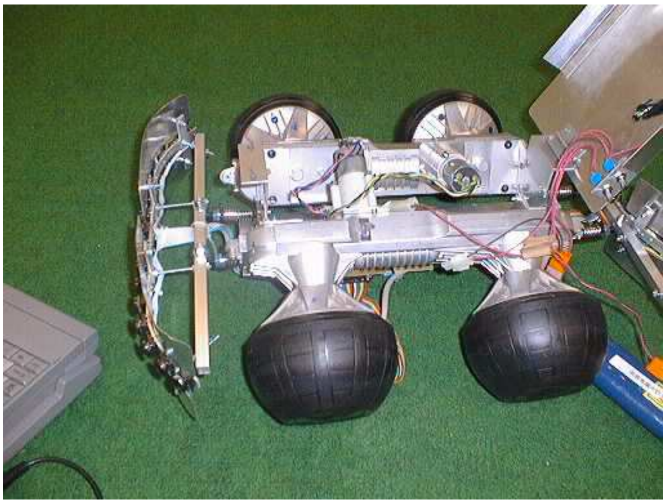
色	塗られるオブジェクト
赤色	ボール
青色	一つのゴール
黄色	もう一つのゴール
緑色	フィールドの表面
白色	壁とライン

2. 各ハードウェアモジュールの写真。

(a):視覚システム



(b):触覚システム



(c):駆動システム