

RoboCup 小型ロボットリーグにおける相手戦略の分析と学習

Analyzing and Learning Opponent's Strategies in the RoboCup Small Size League

安井興太郎、小林邦和、村上和人、成瀬正

Kotaro Yasui, Kunikazu Kobayashi, Kazuhito Murakami and Tadashi Naruse

愛知県立大学 情報科学研究科

Graduate School of Information Sciences and Technology, Aichi Prefectural University

im121007@cis.aichi-pu.ac.jp, {koboyashi, murakami, naruse}@ist.aichi-pu.ac.jp

Abstract

In the RoboCup Soccer we can dominate a game by choosing an effective strategy if we can analyze and learn opponent's strategies in advance. Solving this problem is a challenging task, so we attack the problem in this paper. We propose a dissimilarity function which shows the difference between opponents' deployments at two different times, and extend it to the difference between those of two different time intervals. Then, we analyze opponent's strategies by using the dissimilarity function. As a first step we try to classify the opponent's strategies used in set plays. Employing the dissimilarity matrix generated from the dissimilarity function, we take the cluster analysis and classify the opponent's strategies. We apply this method to the logged data of the small size league's games played in RoboCup 2012. By the experiments, we show we can effectively classify the attacking strategies used in set plays. We also discuss a method to learn the opponent's attacking strategies and to deploy the teammates in advantageous positions on-line in actual games.

1 はじめに

RoboCup 小型ロボットリーグ (RoboCup Small Size League) とは、1 チーム 6 台のロボットが対戦するロボットサッカーリーグである。ビジョンプロセッサは、約 6m × 4m のフィールドの上空に設置された 2 台のカメラから送られてくる画像を処理して、ロボットとボールの座標位置をチームコンピュータに 1 秒 60 回の割合で送る。また、Referee Box コンピュータからスローインやコーナーキックなど、試合をコントロールするコマンドがチームコンピュータに送られる。これらの信号を受け取ったチー

ムコンピュータは、戦略を計算し、チームの各ロボットに移動位置を無線で送信する。このように、システム全体がコンピュータでコントロールされ、Referee Box への指示以外は、人手を介さないシステムになっている。

ロボットの速度は年々高速化してきており、2012 年の優勝チームは最高速度 3.5m/s で制御されている[1]。また、味方ロボットへパスを行う時のボールの速度は 4m/s を超えることも多くなってきている。こうした環境下、相手の行動の予測は非常に重要な課題である。

昨今の RoboCup 小型ロボットリーグでは、主に「ボールの位置・速度」「味方ロボットの位置・速度」「敵ロボットの位置・速度」「レフェリーボックスからの信号」の 4 つの情報をもとに、自チームの戦略を決定している。一方、人間のサッカーでは、各プレイヤーはこれらの 4 つの情報に加えて、敵が過去にどのような行動をしたのかという、「敵プレイヤーの過去の行動情報」を認知した上で、これらをもとに行動の決定を行っている。ロボットのサッカーでも、人間と同じように敵が過去に行った戦略を認識し、それによって優位な行動を選択することができれば、より効果的な戦略でゲームを支配していくことができると考えられる。

本論文では、この課題を解決するために、2 つの試合局面間の非類似度 d を定義し、 d を用いて過去に実行された複数の試合局面のクラスター分析を行うことで、相手戦略を分類する手法を提案する。また、分類結果を用いて、相手がリアルタイムに実行している戦略を予測し、味方に優位な行動を選択させる手法についても考察する。

2 関連研究

Bowling ら (2004) は、RoboCup 小型ロボットリーグにおいて、相手に適合した戦略の選択手法を提案している[2]。彼らの手法では、実行した戦略が成功したか、失敗したか、完了したか、中断したかによって、その戦略に報酬を与えることで、相手に適合していき、有効な戦略を選択

する。

Trevizan, Veloso(2010) は、RoboCup 小型ロボットリーグにおける、2 チームの戦略の比較を行う手法を提案している[3]。彼らの手法では、各ロボットとボールの距離などの 22 個の特徴量の平均値と標準偏差を用いて、戦略を Episode 行列として表現し、2 つの Episode 行列から戦略の類似度を定義している。彼らの実験では、実際の試合ログデータから、複数のチームの守備戦略に対してこの手法を適用し、守備戦略によるチームの分類を行っている。

その他に、Visser, Weland(2003) によって、RoboCup シミュレーションリーグのログから、決定木を用いて相手の行動の分類を行うための手法が提案されている[4]。例えば、ゴールキーパーに対して、ゴールに留まる、ゴールから離れる・ゴールに戻る、という 3 つの行動の分類実験を行っている。また、類似した手法を用いて、敵プレイヤーのパス行動の分析も行っている。

3 戦略の比較

この節では、2 つの試合局面間を比較する手法を提案する。まず、フィールド座標を定義する。原点 O をフィールド中心に取り、 x 軸を自ゴール中心から敵ゴール中心の向きに取って (攻め方向によって向きが変わることに注意する)、 x 軸を反時計回りに $\pi/2$ 回転させた軸を y 軸とする。以降、ロボットの座標はこの座標系で表されているものとする。

時刻 t_0 における m 台の敵ロボットの座標をそれぞれ $R_i(t_0)$ ($1 \leq i \leq m$)、時刻 t_1 における n 台の敵ロボットの座標をそれぞれ $R_j(t_1)$ ($1 \leq j \leq n$) とする。時刻 t_0 における局面と、時刻 t_1 における局面の非類似度 d を次のように定義する。

$$d(t_0, t_1) = \min_{U \in \{U_1, U_2\}} \left\{ \min_{\sigma \in S_6} \sqrt{\text{trace}(F(U) \cdot P_\sigma)} \right\} \quad (1)$$

$$U_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, U_2 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

$$F(U) = [f_{ij}]$$

$$f_{ij} = \begin{cases} \|U \cdot R_i(t_0) - R_j(t_1)\|^2 & (1 \leq i \leq m \text{ and } 1 \leq j \leq n) \\ \Delta^2 & (\text{otherwise}) \end{cases} \quad (2)$$

S_6 は 6 次の対称群、 P_σ は置換 σ の置換行列であり、 U_2 は敵ロボットの y 座標を符号反転させるための x 軸に対する鏡映変換である。 F は 6×6 行列であり、時刻 t_0 におけるそれぞれの敵ロボット座標と時刻 t_1 におけるそれぞれの敵ロボット座標間の全ての組み合わせのユークリッド

平方距離を成分としている。式 (2) のように、2 つの局面でロボットの台数が異なる場合には、その非類似度 d を定数 Δ で補正する。台数が異なる時、非類似度を高く設定したい場合には Δ を大きくし、台数の違いを非類似度に影響させたくない場合には $\Delta = 0$ とすれば良い。

式 (1) は、直感的には、時刻 t_0 におけるそれぞれの敵ロボット座標と、時刻 t_1 におけるそれぞれの敵ロボット座標間のユークリッド平方距離が最も小さくなるように対応付けた時の、それぞれの距離の総和を意味している。1 つの戦略は複数の戦術をロボットに行わせることで実行されるため、対応付けを行うことにより、時間による各ロボットへの戦術の割り当てられ方の違いを吸収できるだろう。また、全ての戦略は x 軸に関して線対称な形が存在すると考えられ¹、対称形を含めて 1 つの戦略と考えるために、二種類の U から非類似度 d を計算する。 d にボールについての直接的な情報が無い理由は、敵ロボットの動作はボールの位置によって変化するため、敵ロボットの座標のみを考えれば、間接的にボールの情報も考えたことになるからである。また、 d には味方ロボットの情報も無い。これは、敵ロボットの戦略を分析する目的として、戦略学習によって味方ロボットの動作が変わるため味方ロボットの情報を加えるべきではないからである。

次に、式 (1) から、時刻 t_0 における局面と、時刻 T_s から時刻 T_e における連続局面の非類似度 d_1 を、次のように定義する。

$$d_1(t_0, T_s, T_e) = \min_{T_s \leq t \leq T_e} \{d(t_0, t)\} \quad (3)$$

式 (3) を用いて、時刻 $T_s^{(i)}$ から $T_e^{(i)}$ における連続局面 X_i と、時刻 $T_s^{(j)}$ から $T_e^{(j)}$ における連続局面 X_j の非類似度 d を、次のように定義する。

$$d_2(T_s^{(i)}, T_e^{(i)}, T_s^{(j)}, T_e^{(j)}) = \min_{T_s^{(i)} \leq t \leq T_e^{(i)}} \{d_1(t, T_s^{(j)}, T_e^{(j)})\} \quad (4)$$

2 つの連続局面に含まれる、最も類似した単一局面同士の非類似度としている。敵が戦略を実行している複数の連続局面に、式 (4) を適用すれば、戦略の比較が可能となる。次の節では、式 (4) によって得られた複数の連続局面間の非類似度を基に、クラスター分析を行って相手戦略を分類する手法について議論する。

4 クラスター分析

戦略を実行している N 個の連続局面に対して式 (4) を適用すると、各戦略間の非類似度を要素とする、 $N \times N$ の

¹ 人間には個体差が存在するために、得意なフィールドサイドがあり得る。しかし、特に小型ロボットリーグでは、各ロボットに個体差があることは少なく、フィールドサイドによって動作が変わることは稀である。

非類似度行列が生成される。この非類似度行列を距離行列と見て、クラスター分析を行うことで、戦略の分類が可能になる。

4.1 分析手法

クラスター分析にはいくつかの手法が存在する。代表的には、次のようなものがある。

k-means 法 非階層的手法の一つで、あらかじめクラスター数 k を決めておき、初期クラスターとして k 個の標本点をランダムに選択する。クラスターに含まれる要素と、そのクラスターの重心点の距離が、他のどのクラスターの重心点の距離よりも小さくなるように、未所属の標本点を振り分けていく[8]。

Ward 法 郡内平方和と群間平方和の比を最大化するように、2つのクラスターを結合していく、階層的手法である。階層的手法の中では最もバランスが良い[9]。

群平均法 2つのクラスターのそれぞれの中から1つずつ要素を選んで要素間の距離を求め、それらの距離の平均値を2つのクラスター間の距離とし、この距離が最も小さいものから順に結合していく、階層的手法である。

k -means 法及び Ward 法は、どちらも実用性の高い手法であるが、クラスターの重心を求める必要がある。今回の問題の場合、クラスターの各要素は連続局面を表しており、複数の連続局面の重心となる連続局面を求めることは困難である。一方、群平均法は任意の要素間の距離のみが定義されていれば適用が可能である。以上の議論から、本論文では群平均法を用いてクラスター分析を行う。

4.2 クラスター数の推定

群平均法では、クラスターの階層構造が抽出されるだけで、クラスターの数分かるわけではない。そこで、Davies-Bouldin index (DBI)[5]を用いてクラスター数の推定を行う。

DBIでは、 K 個のクラスターに対し、次のように定義される $DB(K)$ を計算する。

$$DB(K) = \frac{1}{K} \sum_{i=1}^K \max_{j \neq i} \frac{S_i + S_j}{M_{ij}} \quad (5)$$

$\bar{x}^{(i)}$ はクラスター C_i の重心である。また、 M_{ij} はクラスター C_i, C_j 間の分離性 (Separation) を表しており、 S_i はクラスター C_i の凝集性 (Cohesion) を表している。 M_{ij} と S_i は、自由に定義することができる (但し、満たさなければならないいくつかの条件がある[5])。 $DB(K)$ を、ある範囲内で最も小さくなるようにとる K が、最適なクラスター数となる。しかし、定義上単独クラスターが多すぎ

る場合には、 $DB(K)$ は 0 に近づくので、範囲の選択が重要である。

式 (4) を用いて、 M_{ij} と S_i を次のように定義する。

$$S_i = \frac{1}{|C_i|(|C_i| - 1)} \times \sum_{X_k \in C_i} \left\{ \sum_{X_l \in C_i, X_l \neq X_k} d_2(T_s^{(k)}, T_e^{(k)}, T_s^{(l)}, T_e^{(l)}) \right\}$$

$$M_{ij} = \frac{1}{|C_i||C_j|} \sum_{X_k \in C_i} \sum_{X_l \in C_j} d_2(T_s^{(k)}, T_e^{(k)}, T_s^{(l)}, T_e^{(l)})$$

S_i は同一クラスター内の任意の要素間の平均距離、 M_{ij} は群平均法による2つのクラスター C_i, C_j 間の距離である。これらの S_i, M_{ij} は、[5]に示される条件を満たす。

5 実験

昨今の RoboCup 小型ロボットリーグでの主要な得点シーンは、サイドラインにボールをセットして、ボールが止まった状態から試合を再開する、セットプレー (スローイン、コーナーキック、ゴールキック) であり、各チーム多種多様な戦略が用いられている。そこで、以降の実験では各チームのこれらのセットプレー時における攻撃戦略に対して、これまでの提案手法を次のように適用し、戦略を分析する。

X_i を i 回目 ($1 \leq i \leq N$) のセットプレー局面とし、 X_i の開始時刻を $T_r^{(i)}$ (Referee Box から指示コマンドを受け取った時刻)、 X_i においてキッカーがキックした時刻を $T_e^{(i)}$ とする。

$$T_s^{(i)} = \max(T_e^{(i)} - T_{behavior}, T_r^{(i)}) \quad (6)$$

とし、式 (4) より、非類似度 $d_2(T_s^{(i)}, T_e^{(i)}, T_s^{(j)}, T_e^{(j)})$, ($1 \leq i \leq N$ and $1 \leq j \leq N$) を計算する。そして、生成された $N \times N$ の非類似度行列を基に、群平均法によってクラスターの階層構造を得る。最後に、式 (5) によってクラスター数 K を推定し、 K 個の戦略に分類する。

$T_{behavior}$ は、キック直前にとった行動によって戦略を分類するために必要な定数で、適用しているチームのロボットが一つの行動にかかる時間を指定する。ロボットの速度が速いチームに対しては $T_{behavior}$ を小さくすればよい。 $T_{behavior}$ を大きめの値にしておくことで、同じ戦略の局面ごとのばらつきを吸収できるが、未知の相手に対しては、小さめの値にしておけば問題ない。本論文では、 $T_{behavior} = 1.0\text{sec}$ を使用する。また、式 (5) による K の推定範囲には、スタージェスの公式[6]

$$1 \leq K \leq \lceil \log_2 N + 1 \rceil \quad (7)$$

を用いた。ここで、 $\lceil x \rceil$ は x の天井関数で、 x 以上の最小の整数を返す。

5.1 自チームへの適用

まず、戦略が既知である、我々の所属チーム RoboDragons に対して分析を行う。

サイドラインにボールをセットして試合を再開するセットプレーにおいて、RoboDragons が 2012 年の世界大会で使用した攻撃戦略は合計 4 つである。これをそれぞれ A_i , ($1 \leq i \leq 4$) とする。RoboDragons のシミュレーションシステムによって、RoboDragons (Blue) 対 RoboDragons (Yellow) の疑似試合を 6 台同士で実行し、Yellow チームを相手チームと仮定して、4 つの攻撃戦略をそれぞれ 6 回ずつ、計 24 回行わせた²。セットプレー開始時のボールの y 座標は 2012 年のルール[7]に則り 1915mm とし、 x 座標は 4 つの戦略が実行可能な範囲をそれぞれ 6 分割した位置とした。

群平均法によるクラスター分析結果のデンドログラムを Fig.1 に、DBI を Fig.2 に示す³。

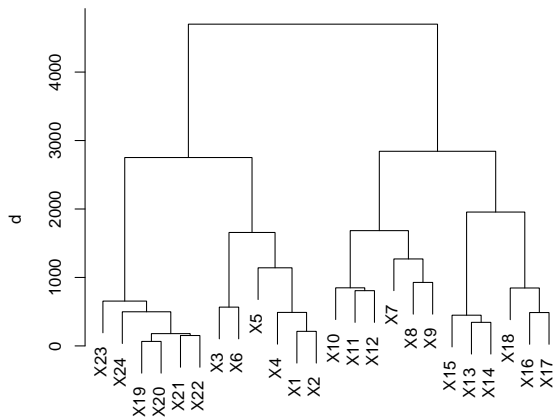


Figure 1: Dendrogram (RoboDragons)

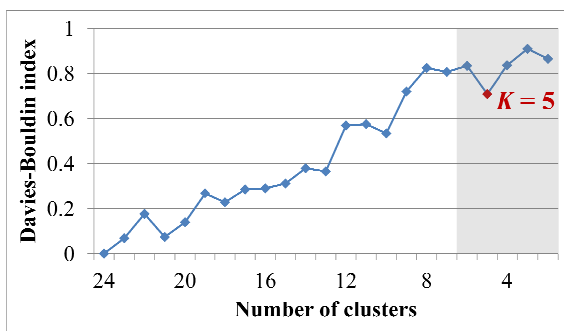


Figure 2: Davies-Bouldin index (RoboDragons)

Fig.2 から、クラスター数は $K = 5$ であると推定され

² 各攻撃戦略には与えるパラメータがいくつかある。世界大会で使用していたパラメータでは、提案手法を適用すること簡単に分類できてしまったため、パラメータを変更してシミュレーションシステムで実験を行った

³ 本論文内のデンドログラムは、統計解析ソフト R を用いて生成している。

ていることが分かる。Fig.1 のデンドログラムを、クラスター数が $K = 5$ となるとところで切断すれば、戦略が次の 5 つのクラスターに分類される。

$$C_1 = \{X_1, X_2, X_3, X_4, X_5, X_6\}$$

$$C_2 = \{X_7, X_8, X_9, X_{10}, X_{11}, X_{12}\}$$

$$C_3 = \{X_{13}, X_{14}, X_{15}\}$$

$$C_4 = \{X_{16}, X_{17}, X_{18}\}$$

$$C_5 = \{X_{19}, X_{20}, X_{21}, X_{22}, X_{23}, X_{24}\}$$

4 つの攻撃戦略は順に 6 回ずつ行ったので、戦略 A_1, A_2, A_4 は C_1, C_2, C_5 に正しく分類されていることが分かる。戦略 A_3 は 2 つのクラスター C_3, C_4 に分かれてしまっている。しかし、戦略が混合したクラスターは見られず、また $K = 4$ とすれば、分割されてしまった C_3, C_4 は結合されることから、手法の有効性が確認できる。また、Fig.1 のクラスター C_5 の高さから、戦略 A_4 は他の戦略に比べてばらつきが少ない戦略であり、再現性の高い戦略であることも分析できる。

5.2 他チームへの適用

次に、我々の所持している、2012 年世界大会の決勝 Skuba(Blue) 対 ZJUNlict(Yellow) の試合ログデータから、Skuba, ZJUNlict, 双方のチームのセットプレー戦略へ適用した。サイドラインにボールをセットして試合を再開するセットプレーが、Skuba 側からは 37 回、ZJUNlict 側からは 25 回行われていた。これらのセットプレー局面をクラスター分析した結果、得られたデンドログラムを Fig.3, 4 に示す。

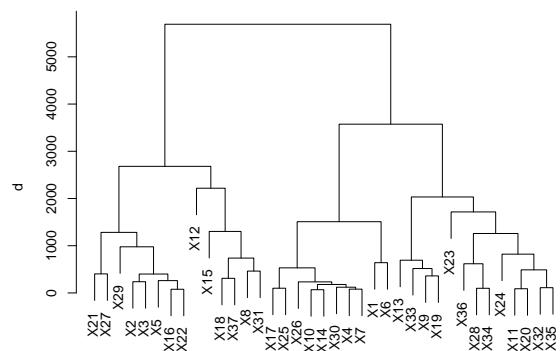


Figure 3: Dendrogram (Skuba)

式 (5) により推定されたクラスター数は、Skuba が $K = 5$, ZJUNlict が $K = 6$ であった。これにより、以下の分類結果が得られる。なお、 i 回目のセットプレー局面 X_i について、キッカーがキックした時刻 $T_k^{(i)}$ における試合状況を、我々のシステムを用いて 2 次元描画した図を同時

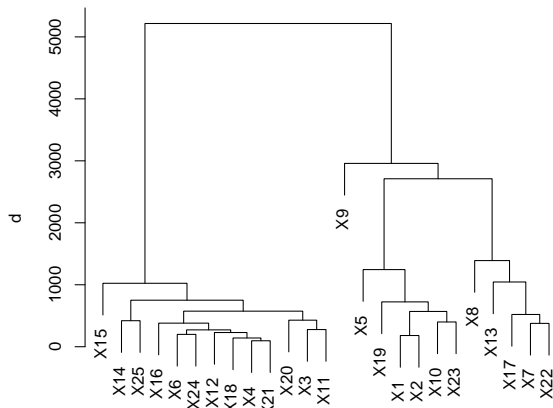


Figure 4: Dendrogram (ZJUNlict)

に並べて示している。この図では、視認性向上のために、各ロボットの大きさを通常の3倍に、ボールの大きさを5倍にしてある。また、ロボットに表示されている数字は、そのロボットのIDを表している。

この試合は、前後半でゴールを入れ替えているので、Skubaのセットプレーでは X_{17} から、ZJUNlictのセットプレーでは X_{16} から、 x 軸の向きが反転することに注意する。

戦略パターンの分類結果から、戦略の分析が容易になる。Skubaから見ていくと、クラスター C_1 はパスをせずにゴール方向へ直接蹴りだす戦略、クラスター C_2 はコーナーキック時に敵ディフェンスエリア付近ファーサイドの味方ロボットへパスする戦略、クラスター C_3 は敵陣側からのスローイン時にセンターライン付近ニアサイドの味方ロボットへパスする戦略であると推測できる。また、クラスター C_4 はクラスター C_1 と同じような戦略が一部含まれてしまっているが、センターライン付近からファーサイドの味方ロボットへパスする戦略であり、単独クラスターとなった C_5 は、フィールド中央の味方ロボットへパスする戦略であると推測できる。

次にZJUNlictについて見ていくと、クラスター C_1 はコーナーキック時に敵ディフェンスエリア付近ファーサイドの味方ロボットへパスする戦略、要素数が最多のクラスター C_2 は敵ゴール方向にいる味方ロボットに縦パスを行う戦略、クラスター C_3 はクラスター C_1 に似た戦略、クラスター C_4 は敵陣側からのスローイン時にファーサイドの味方ロボットへパスする戦略、クラスター C_5 は、クラスター C_4 に似た戦略と推測できる。クラスター C_6 は、クラスター C_4 に似ているが、逆サイドに2台のロボットが配置されているので、別の戦略の可能性も考えられる。

以上のように、提案手法によって戦略パターンの分類が可能となり、また、得られた分類結果は、人に目による戦略分析にも大いに役立つことが分かる。

6 リアルタイム学習への応用

5節の実験結果から、提案手法によって戦略パターンの分類が可能となることが分かった。試合中、敵によって $N+1$ 回目の戦略が実行されているときに、それまでに実行された N 回の戦略の分類結果を使うことで、リアルタイム学習を行って味方に優位な行動をさせることが可能になる。例として、5節と同じくサイドラインにボールをセットして試合を再開するセットプレーの学習手法について考察する。

クラスター C_i に含まれる複数の連続した試合局面 X_j と、現在時刻 t における試合局面の非類似度 d_3 を、次のように定義する。

$$d_3(t, C_i) = \frac{1}{|C_i|} \sum_{X_j \in C_i} d_1(t, T_r^{(j)}, T_e^{(j)}) \quad (8)$$

現在時刻 t における試合局面と、クラスター C_i に含まれる X_j の非類似度の平均値を求めている。式(8)を、提案手法によって分類された全てのクラスターに対して計算することで、敵が現在実行している戦略が、それまでに実行したどの戦略に近いかをリアルタイムに求めることができる。式(8)に式(6)で定義した $T_s^{(j)}$ ではなく、 $T_r^{(j)}$ を用いる理由は、キック直前にとった行動だけではなく、準備のために行動している局面とも比較を行うことで、その戦略が実行される前兆を検出したいためである。

式(8)を、5.2節と同じく2012年世界大会決勝のSkubaの37回目のセットプレー X_{37} に対して計算した。なお、1~36回目のセットプレー局面 $X_1 \sim X_{36}$ の分類結果は、5.2節のSkubaの実験結果と同じものが得られているものと仮定した。但し、クラスター C_2 からは X_{37} を除外した。結果をFig.5に示す。また、セットプレー X_{37} においてSkubaのキッカーロボットがキックするまでの状態を4秒前から1秒ごとに2次元描画した図を同時に並べて示す。

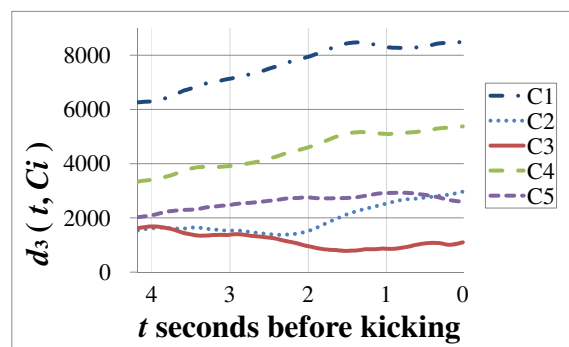
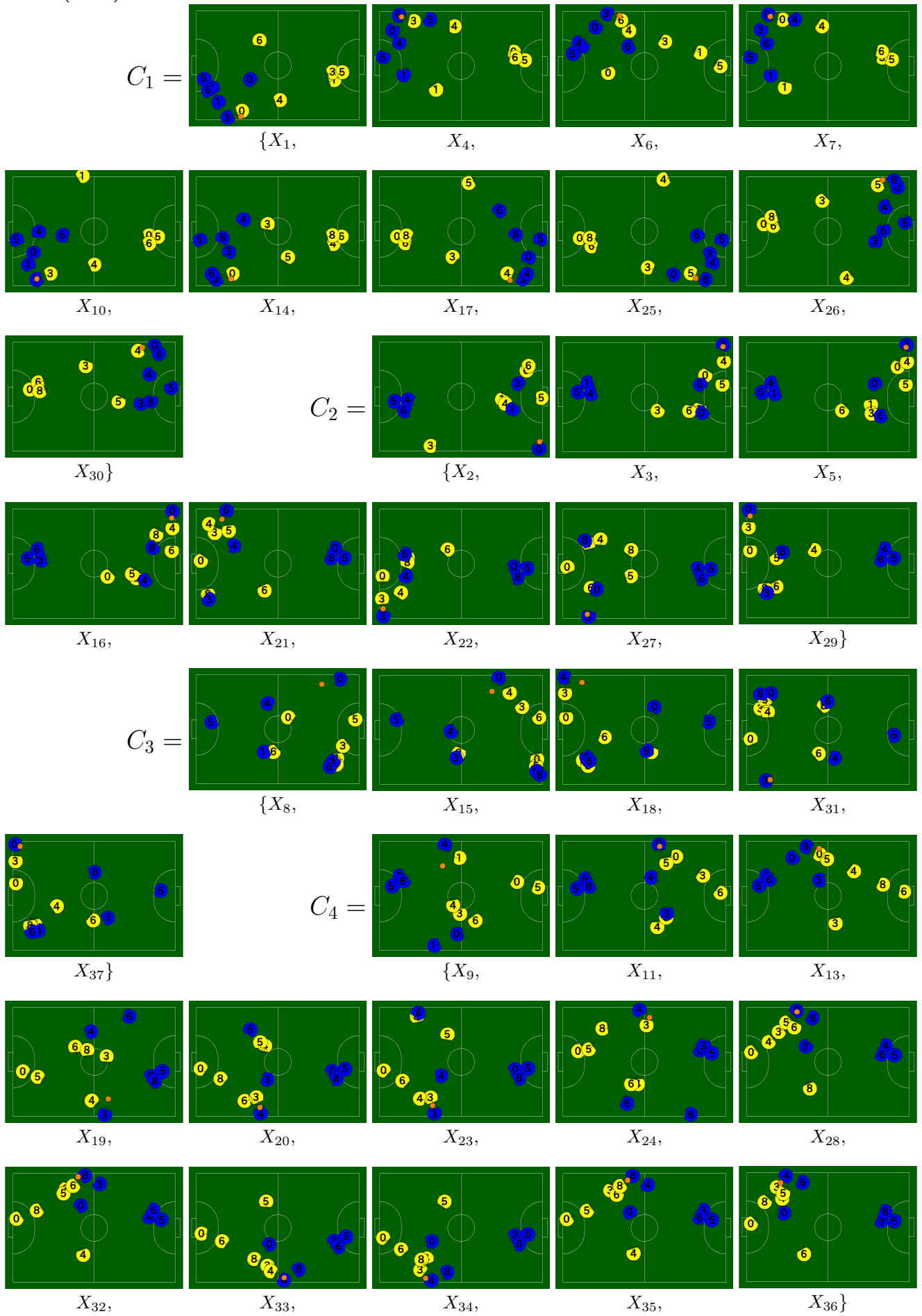
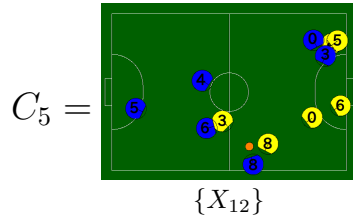


Figure 5: 各クラスターごとの式(8)の値。Referee Boxからセットプレー開始信号を受け取ってから、キッカーによるキックが完了するまでの推移。

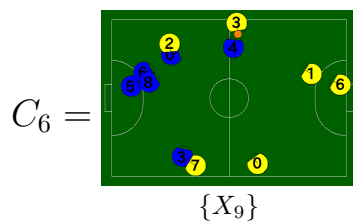
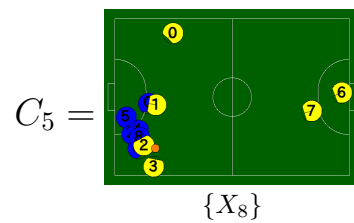
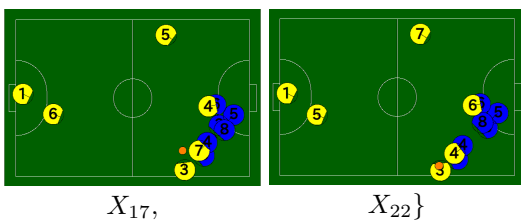
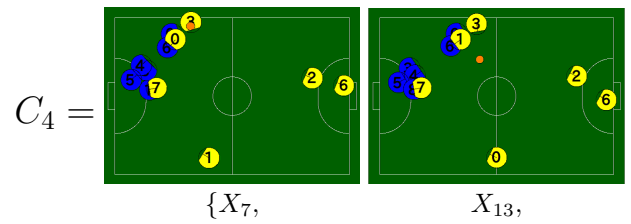
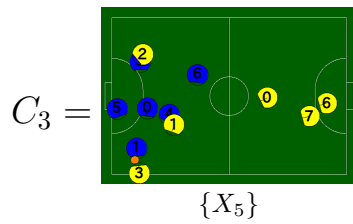
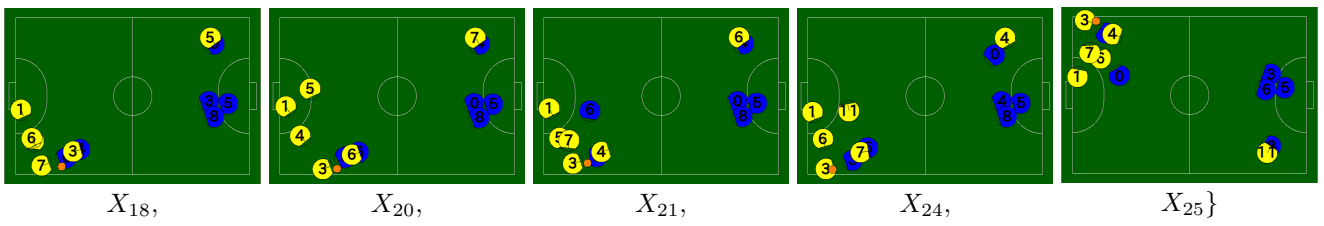
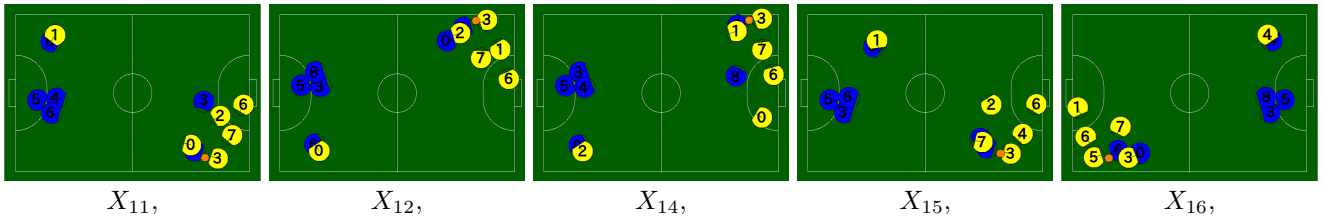
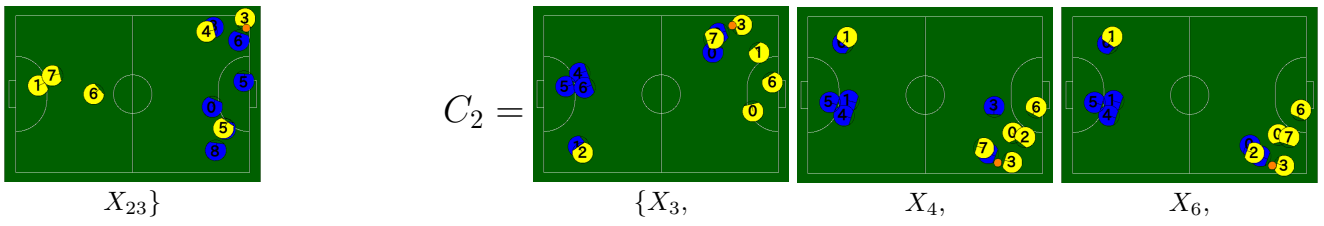
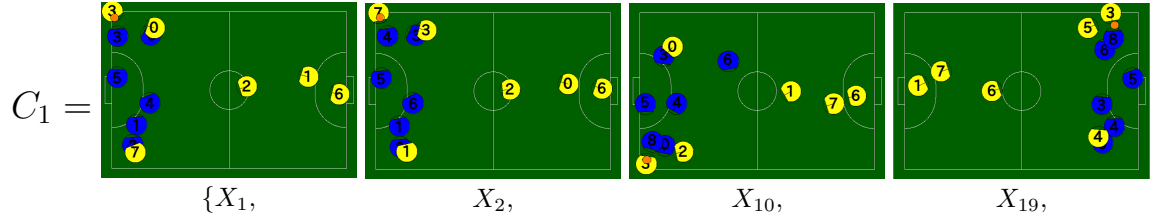
キック4秒前の時点で、クラスター C_1, C_4 は $d_3(t, C_i)$

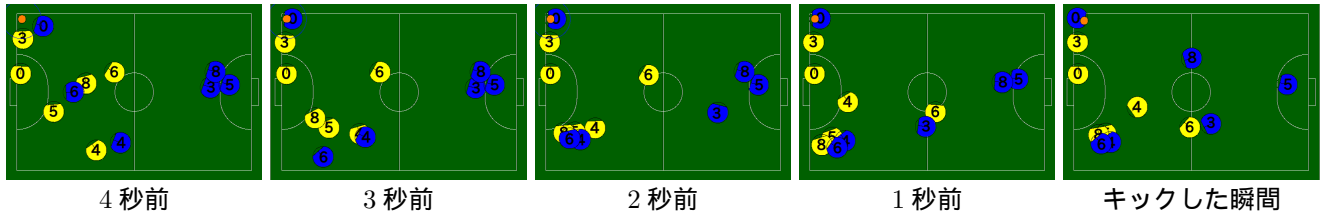
Skuba(Blue)





ZJUNlict(Yellow)





が高いことから、この時実行されている戦略は、これらのクラスターに対応する戦略とは異なることが分かる。2秒前までは、クラスター C_2, C_3 が小さな値をとっているが、以降は C_2 の値が大きくなる。これは、2秒前に Skuba の ID:3 のロボットが敵陣に飛び出してくるためである。その後、キック1秒前に同じように ID:8 のロボットが飛び出してくるが、これらの動作に関わらず、クラスター C_3 のみ、 $d_3(t, C_i)$ が Referee Box から信号を受け取った時刻から常に低い値で推移しており、式 (8) を用いれば過去に実行されたクラスター C_3 に対応する戦略が、今もう一度実行されそうであることが予測できる。

クラスター C_3 に対応する戦略は、全てディフェンスエリアから飛び出してきた、センターライン付近ニアサイドのロボットへパスを行っていた。したがって、同じ位置関係にいる、ID:8 のロボットがシュートを行う可能性が高く、2秒前にとび出してくる ID:3 のロボットはおとりロボットである可能性が高いと判断できる。ID:8 のロボットに対して、味方ロボットにマークに向かわせるなどの守備行動を、キックの2~4秒前に行わせることで、この戦略を無効化することは十分に可能である。

7 おわりに

本論文では、2つの試合局面間の非類似度 d を定義し、 d によって複数の局面のクラスター分析を行うことで、相手の戦略を分類・分析する手法について提案した。そして、提案手法を自チームの試合へ適用することで、その有効性を確認した。また、他チームの試合にも適用し、実際に敵の戦略を分類・分析することに成功した。さらに、分類結果から相手の戦略をリアルタイムに学習する手法についても考察した。今後は、戦略分類のさらなる高精度化が課題である。

参考文献

- [1] Thanakorn Panyapiang, Krit Chaiso, Kanjanapan Sukvichai and Phawat Lertariyasakchai, “Skuba 2012 Extended Team Description”, 2012
- [2] Michael Bowling, Brett Browning and Manuela M. Veloso, “Plays as Effective Multiagent Plans Enabling Opponent-Adaptive Play Selection”, International Conference on Automated Planning and Scheduling, 2004
- [3] Felipe W. Trevizan and Manuela M. Veloso, “Learning Opponent’s Strategies In the RoboCup Small Size League”, International Conference on Autonomous Agents and Multi-Agent Systems, Springer, 2010
- [4] Ubbo Visser and Hans-Georg Weland, “Using Online Learning to Analyze the Opponents Behavior”, RoboCup 2002: Robot Soccer World Cup VI, pp.78-93, Springer, 2003
- [5] David L. Davies and Donald W. Bouldin, “A Cluster Separation Measure”, IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-1(2), pp.224-227, 1979
- [6] Herbert A. Sturges, “The Choice of a Class Interval”, Journal of the American Statistical Association, Vol.21, No.153, pp.65-66, 1926
- [7] “Laws of the RoboCup Small Size League 2012”, viewed April 4th 2013, http://robocupssl.cpe.ku.ac.th/_media/rules:ssl-rules-2012.pdf
- [8] “ k -means clustering”, viewed April 4th 2013, http://en.wikipedia.org/wiki/K-means_clustering
- [9] “Ward’s method”, viewed April 4th 2013, http://en.wikipedia.org/wiki/Ward%27s_method

RoboCup SSL Humanoid のためのカラーボクセルサーバの提案と応用

A Proposal and its Application of Color Voxel Server for RoboCup SSL Humanoid

植田康生 升谷 保博

Yasuo UEDA Yasuhiro MASTANI

大阪電気通信大学

Osaka Electro-Communication University

Abstract

In this paper, the authors propose “color voxel server” as a shared vision system for RoboCup SSL Humanoid, which generates coloring voxel data of 3D space on a field from images taken with multiple external cameras and supplies it via network. Kimura proposed an original “voxel server” in 2010, which is insufficient to distinguish objects. Because it generates binary voxel data. Therefore the proposed server gives color information to voxel without spoiling realtime performance by not using full color and reducing number of colors in the camera images. Seitz’s method of voxel coloring is used to solve color of voxel. It takes about 80[ms] for a prototype program to generate 4-color 10[mm]-cubic voxels of 4050[mm]×3025[mm]×500[mm] space from 8 camera images and supply compressed data via network. Moreover the authors show that it is possible to distinguish objects and decide action of robot based on color voxel data provided from the server.

1 はじめに

RoboCup サッカーの小型ロボットリーグ (Small Size Robot League, 以下 SSL)[長坂, 2011]とそのサブリーグである SSL Humanoid[升谷, 2010]は, 他のリーグと異なり, ロボット自身にはカメラが搭載されておらず, 外部カメラを用いてフィールド上のロボットやボールの位置を認識しロボットを制御している.

現状では, Figure 1 のようにフィールド上空のカメラでロボットの頭頂部に取り付けたマーカ板のパターンを

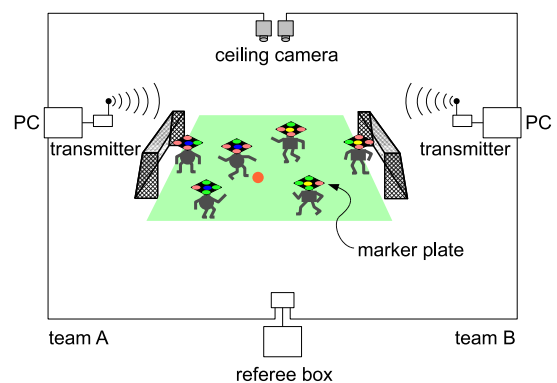


Figure 1: Initial phase of the SSL Humanoid (2009)

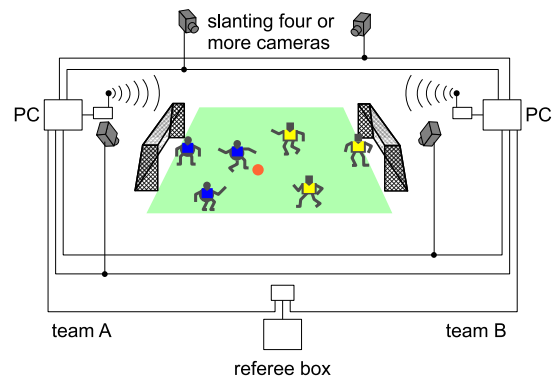


Figure 2: Final phase of the SSL Humanoid (2015)

撮影し, 2 次元的な画像処理を経て, その番号や位置, 方向を算出している. しかし, SSL Humanoid のロードマップでは, Figure 2 のように 2015 年頃には複数の外部カメラで撮影したカメラに基づいて, 複数のロボットの 3 次元的な形状を取得し, それに基づきロボットを制御することを目指している.

SSL では, 2010 年から視覚系の共有が義務付けられた. 参加チームは, カメラとその画像を処理する SSL-Vision と呼ばれるサーバプログラムを共有し, SSL-Vision から送られてくるフィールド上のオブジェクトの位置情報を利

Table 1: Identification by color of voxel

Voxel	Mode of expression
Nothing	0
Other	1
Orange	2
Blue	3
Yellow	4

用する．SSL Humanoid でも視覚系の共有は必須課題である．現状では，SSL-Vision を利用しているが，3次元の場合に何をどのように共有するかはまだ決まっていない．

これを受けて，木村らはフィールド上の3次元空間のボクセルデータを共有することを提案し，それを具体化する「ボクセルサーバ」を開発した[木村, 2010]．このサーバは，8台のカメラを用い，3025[mm]×4050[mm]×500[mm]の領域を10[mm]立方のボクセルで表現し，それをネットワークを介してクライアントへ配信している．また，クライアント側では，ボクセルデータからロボットのポーズ推定を行い，その結果からロボットの行動決定ができることを示している[植田, 2011]．しかし，木村の手法では，ボクセルデータは，個々のボクセルが空間を占めるか否かの2値の集合である．そのため，複数のロボットを区別することが難しい．また，ロボットと接触しているボールを別のオブジェクトとして識別できないという問題もある．

そこで，木村の方法を拡張し，複数のカメラ画像から色付けしたボクセルデータを算出し，それをネットワーク経由で配信する「カラーボクセルサーバ」を提案する．本稿では，その仕様や実現方法について述べ，実装したプログラムを評価する．さらに，サーバから提供されたカラーボクセルデータに基づいて，オブジェクトの識別とロボットの行動決定を行い，提案手法の有効性を検証する．

2 カラーボクセルサーバ

この節では，本稿で提案する「カラーボクセルサーバ」について述べる．カメラ画像の色情報に基づいてボクセルデータを多値化する．各ボクセルの色は，写実的に表現するのではなく，色数を数 bit に留める．ここでは，Table 1 に示す5値を用いる．これは，最低限の要件として，ボクセルなし，ボール，2つのチームを表す色，それ以外の色が表現することができれば良いからである．また，サーバ側では，多値化したボクセルデータを生成しクライアントへ送信するのみであり，色によるオブジェクトの識別などは行わない．これは，サーバ側で各オブジェクトの識別を行うと，処理時間がかかってしまい実時間での配信が難しくなるためである．また，このようにすることでボクセルデータの利用側で工夫や研究を行う余地ができる．

2.1 カメラの撮影対象

本提案では，カメラで撮影する対象に明確な色が付いていることを前提とする．オレンジ色のボールや，区別のつきやすい色のユニフォームのような物を装着しているロボット等である．ただし，ロボットにおいては全体が同じ色で覆われている必要はなく，一部で構わない．

2.1.1 複数色のシルエット画像生成

提案手法では，処理を減らすために，カメラ画像の段階で色数を減らし，必要な色だけを抽出する．カメラ画像の画素値と背景画像の画素値の差分を取り，差があった画素のみに対して色抽出を行う．画素値をHSVで表現し各成分が設定の範囲内であれば所定の色とする．もし，どの色の範囲内でもなければ，シルエット画像の画素値は白とする．カメラ画像と背景画像の差分がなければ，シルエット画像の画素値は黒となる．

2.1.2 ボクセルの生成

シルエット画像に基づき，視体積交差法的一种であるSpace Carving Method(SCM)によって，ボクセルデータを生成する．SCMは，シルエット制約に基づいて，オブジェクトの内点を残して，それ以外の点を削り落とす(ボクセルカットと呼ぶ)手法である．高速化のために，領域を分割しマルチスレッドで並列処理する．

2.1.3 ボクセルの色付け

色付けにはVoxel Coloring手法[M.Seiz, 1997]を用いる．複数の入力画像の色情報を用いて，3次元復元を行う手法である．ここでは，z軸と垂直な層を考え，カメラに近い層から順に処理を行う．各ボクセルと対応するシルエット画像の画素値をカメラ台数分取得する．取得する際に，すでに投影されているかどうかを示す可視判定マスクが1であれば，そのカメラの画素値は取得しない．取得した画素値を比較し，全てが同じ色であれば，ボクセルの色付けを行う．一つでも異なる色があれば，そのボクセルを削除する．しかし，異なる色が白の場合にはボクセルを削除しない．一つの層の処理が終了したら，取得した画素に対応する可視判定マスクを1に変更する．

2.2 通信プロトコル

クライアントへのデータ転送にはUDPマルチキャストを用いる．送信する内容はボクセルの1辺の長さ[mm]，ボクセル空間の3辺の長さ[mm]，ランレングス圧縮後の総データ量[byte]，ランレングス圧縮後のデータである．ランレングス圧縮では1つのランを2byteで表現する．上位3[bit]をラン値(ボクセルの値)，以降の下位13[bit]をラン長とする．

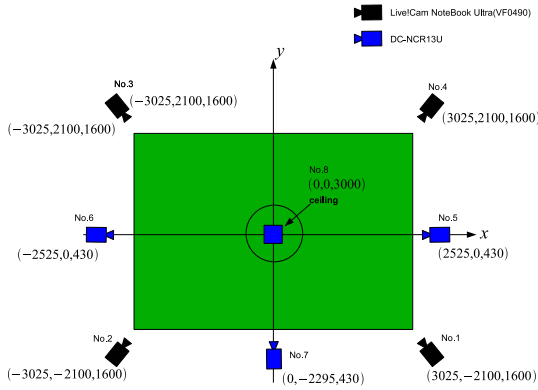


Figure 3: Camera layout

Table 2: Specifications of the experimental PCs

Server PC	
OS	Windows 7 Professional 64bit
Main Mamory	5GByte
CPU	Intel Core i7 950(3.19GHz)
Graphic Board	NVIDIA GeForce GTX260
Client PC	
OS	Windows 7 Professional 64bit
Main Mamory	4GByte
CPU	Intel Core i7 X940(2.13GHz)
Graphic Board	ATI Mobility Radeon HD 5800 Series



Figure 4: Camera images

2.3 実行例

提案するカラーボクセルサーバを実装したプログラムの実行例を示す．プログラムを実行する PC の仕様を Table 2 に示す．カメラの配置を Figure 3 に示す．Voxel Coloring に使用するカメラは，Figure 3 において黒色で表されている 4 台である．これらは，撮影対象の空間よりも上部に設置されており， z 軸と垂直な同じ平面上に設置されている．

フィールド上に青チームのロボットと黄色チームのロボットを 3 体ずつ配置する．また，ボールをフィールドの中央に置く．取得したカメラ画像が Figure 4，生成されたシルエット画像が Figure 5，それらから生成されたボクセルデータが Figure 6, 7 である．Figure 5 では差分のある画素のうちオレンジ，黄色，青色以外は黒で表し，差分のない画素は緑で表している．カラー化する前のボクセルサーバとの実行周期の比較を Figure 8 に示す．

この実行例から，オレンジ，黄，青，それ以外の 4 色でボクセルを表現できていることがわかる．しかし，多値

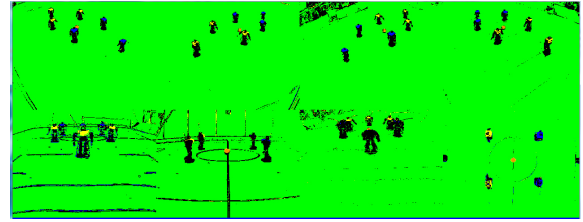


Figure 5: Silhouette images

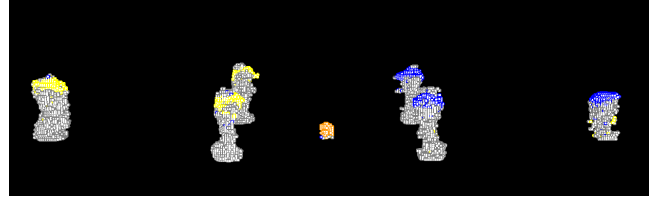
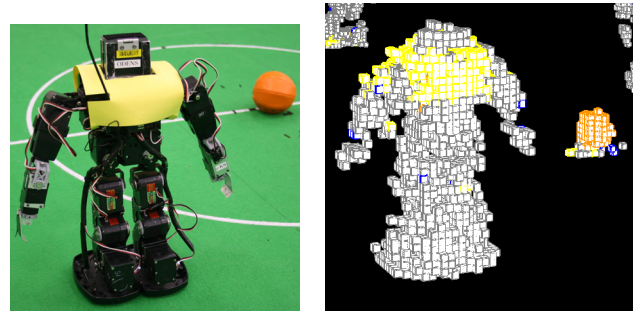


Figure 6: Generated voxel data



(a) Real image

(b) Voxel atea

Figure 7: Camera image and voxel data

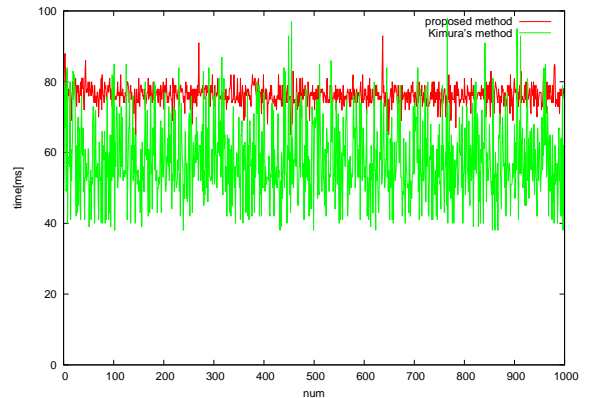


Figure 8: Comparison of the processing time of the proposed method and Kimura's method

化処理を行っているので処理時間が約 34%増加している．

3 オブジェクト識別

この節では，カラーボクセルデータに基づくボールやロボットの識別について述べる．

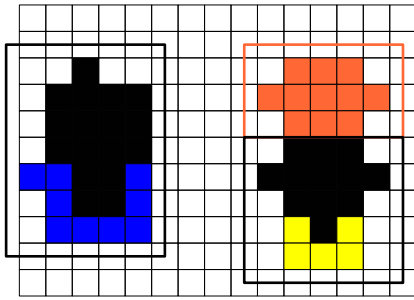


Figure 9: Two types of labeling

3.1 多値ラベリング

ボールのように、全体が1色で覆われているオブジェクトの色と、それ以外の色の2種類に分けてラベリングを行う。Figure 9のようにボールを覆っているオレンジ色の場合は、その色のボクセルだけでラベリングを行い、それ以外の色のボクセルはまとめてラベリングを行う。これにより、ボールがロボットと接触していても、同一のボクセル群にならず、ボールを認識することができる。

3.2 ボール・チーム識別

オレンジの値のボクセルのみでラベリングしたボクセル群の中からボールに近い体積のものをボールと認識する。Voxel Coloringでは表面のみ色付けされるため、ボールの内部にオレンジではないボクセルが存在するが、そのサイズが小さいため、影響はない。

ロボットに対しては、ラベリングされたボクセル群に含まれる青と黄のボクセルを数え、多い方の色のチームに決定する。

3.3 号機判定

停止状態の各ロボットにそれぞれ別のコマンドを送信する。一定時間経過後、ボクセル群の初期位置と現在位置の関係から、対応するロボットの行動を推定する。それを実際に送ったコマンドを比較し、号機番号を判定する。

3.4 ロボットの動作の監視

コマンドと動作の対応の監視を常に行う。ロボットに以前と異なるコマンドを送る際に、そのロボットの位置を動作の開始位置とする。次回以降の処理では、ロボットの開始位置と現在位置との位置関係から行動通り動作した際の想定した位置関係になっているか監視する。所定の回数連続で想定される位置関係でないと判定した場合、誤判定として検知する。

3.5 号機の再判定

号機の誤判定を起こした際に、再度号機判定を行う。また、異常行動の原因の一つとして、ロボットの方向を180[deg]誤認識している場合がある。この場合、例えば、後進コマンドを送ると、前進行動と認識される。このような場合

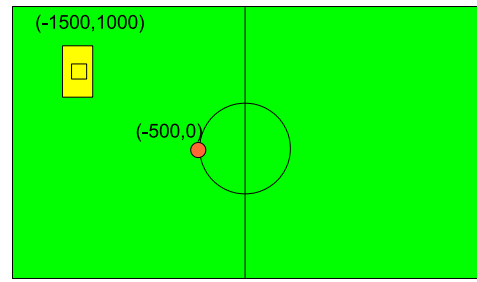


Figure 10: Initial positions of robot and ball

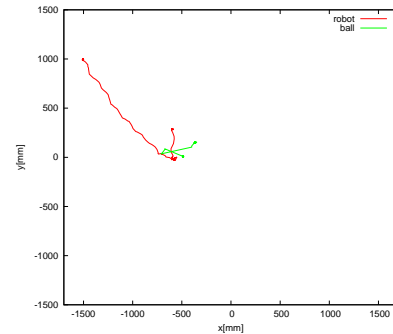


Figure 11: Trajectories of robot and ball by Kimura's method

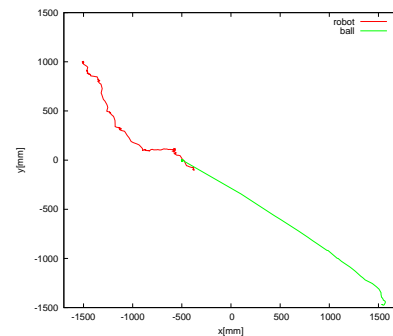


Figure 12: Trajectories of robot and ball by the proposed method

は、そのボクセル群を後進コマンドを送信したロボットの号機と判定し、方向の修正を行う。

4 評価実験

この節では、カラーボクセルデータに基づくオブジェクト識別の評価実験について述べる。クライアントPCの仕様をTable 2に示す。サーバ側では多値化したボクセルデータを生成し、クライアントに送信する。クライアントでは、受信したデータを基に行動決定を行う。

4.1 ボールキック行動の比較実験

提案手法を木村の手法と比べて検証を行う。そのための課題としてボールキック行動を用いる。ボールから離れた位置にいるロボットにボールをフィールドの中央に向けて蹴るように行動させる。木村の手法では2値のボクセル群の中から、体積がボールに近いものをそれと認識して

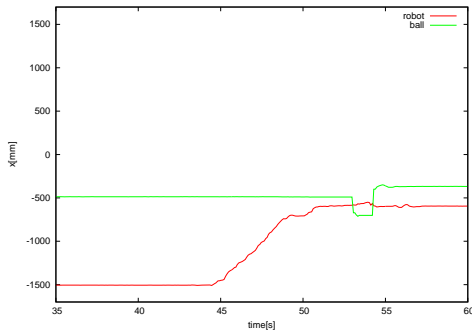


Figure 13: X coordinate of robot and ball by Kimura's method

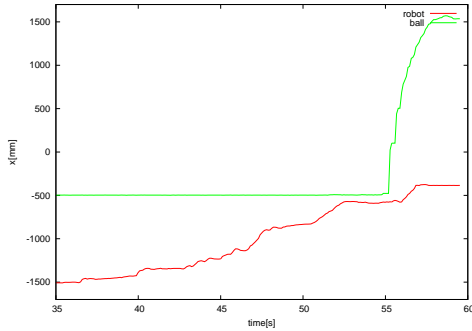


Figure 14: X coordinate of robot and ball by the proposed method

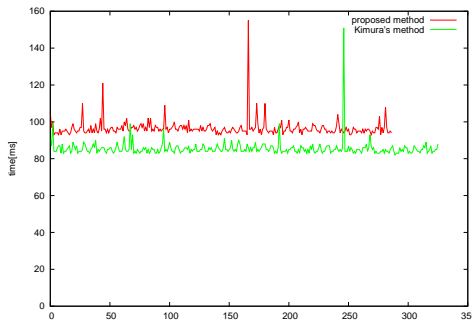


Figure 15: Comparison of the processing time of the proposed method and Kimura's method by client

いる．ロボットとボールの初期位置を Figure 10 に示す．

木村の手法によるロボットとボールの軌跡を Figure 11 に．x 座標の時間変化を Figure 13 に示す．提案手法によるロボットとボールの軌跡を Figure 12 に，x 座標の時間変化を Figure 14 に示す．両手法の処理時間を Figure 15 に示す．

木村の手法では，ボールに最も接近した 53[s] において，ロボットとボールが接触し，ボールとは別のボクセル群が誤ってボールと認識されているため，うまく蹴ることができていない．一方，提案手法では，ロボットがボールに接触しても，ボールを認識できており，ボールをうまく蹴ることができている．この結果から，木村の手法ではボールを認識できていない場面でも提案手法では問題な

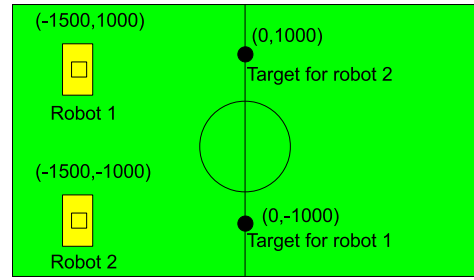


Figure 16: Initial and target positions of robots

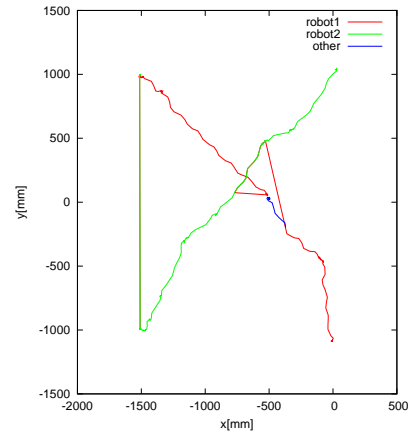


Figure 17: Trajectories of robots

いことがわかる．

また，処理時間を比較すると，提案手法の方が 1 回あたり約 10[ms] 長い．これは，ラベリング処理を 2 回行っているためとロボットの動作の監視を追加したためである．また，誤って存在しないロボットを認識している場合があり，そのロボットのポーズ推定も行うため，余分に処理時間がかかっている場合がある．

4.2 ロボットの号機判定実験

プログラム開始時や実行中にロボットの号機を正しく判定できているか検証する．フィールド上のロボットの初期位置と目標位置を Figure 16 に示す．同じユニフォームカラーの 2 台のロボットが交差するような移動経路をとるように目標位置を設定している．ここでは，2 台のロボットへのコマンド送信は 1 個のプログラムで行う．

各ロボットの軌跡を Figure 17 に示す．また，x 座標と y 座標の時間変化をそれぞれ Figure 18 と 19 に示す．また，ロボットの方向の時間変化を Figure 20 に示す．これらの図中の黒線は号機の再判定を行っているかを表し，紫線と水色線はそれぞれ号機の動作の監視結果 (ok または error) を表す．

Figure 19 によれば，24[s] に 1 号機と 2 号機の位置が入れ替わっているのがわかる．Figure 20 を見ると，1 台のみ角度が変化している．これは号機判断を行うために 1 号機に旋回行動を行わせているからである．その結果が

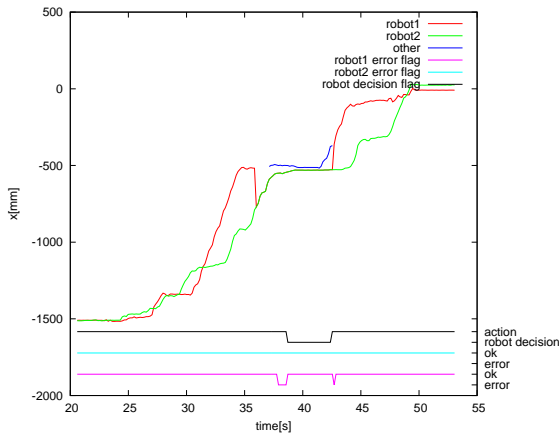


Figure 18: X coordinate vs. time of robot 1, 2, and other

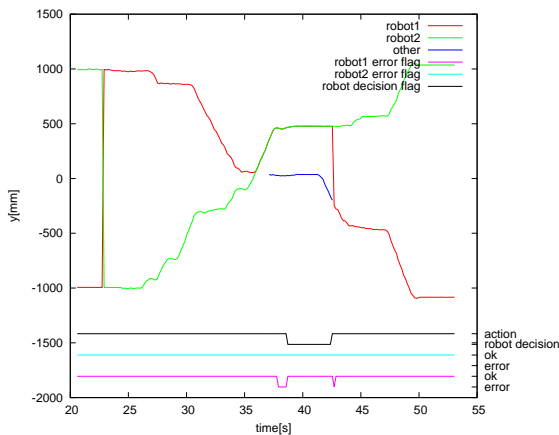


Figure 19: Y coordinate vs. time of robot 1, 2, and other

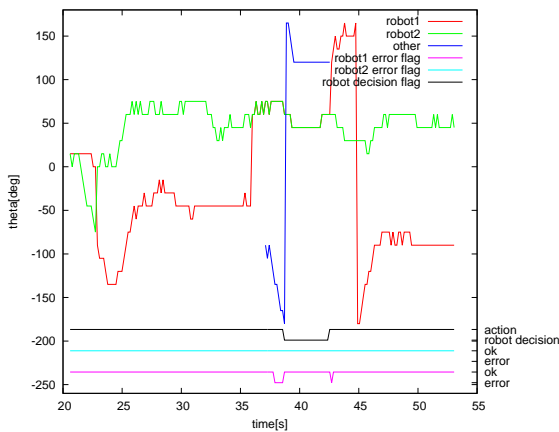


Figure 20: Direction vs. time of robot 1, 2, and other

ら、旋回行動を行っているロボットを1号機と認識することができていることがわかる。

次に、36[s]にロボットの位置が重なっている。これはロボットが接近したため同じボクセル群を自機のロボットだと判断している。1号機と2号機のどちらもが2号機のボクセル群を自機だと認識してしまい、1号機のボクセル群を別のロボットとして認識してしまっている。そのため、38[s]に1号機の異常行動を検知し、号機の再判定を行っている。このとき、1号機が後進している。43[s]に

号機判定が終わり、後進していたロボットを1号機、停止していたロボットを2号機と認識できている。その後、両機とも指定の目標位置に到達できている。

以上の結果により、ロボットが交差したり接触しても、ロボットの異常行動を検知することにより、号機の再判定ができ、その後の行動決定を正しくできていることがわかる。しかし、現段階では、ロボットの号機判定には、ボクセル群の位置と方向しか使っていない。ボクセルデータから推定されたヒト型ロボットのポーズを使うことによって、短時間で確実な判定ができると思われる。

5 おわりに

本稿では、RoboCup SSL Humanoidの共有視覚システムとして利用することを想定して、複数の外部カメラから得られた画像から色付けされたボクセルデータを生成し、それをネットワークで配信する「カラーボクセルサーバ」を提案し、実装したプログラムの実行例を示した。さらに、カラーボクセルサーバから提供された色付けされたボクセルデータに基づいて、オブジェクトの識別とロボットの行動決定を行い、提案手法の有効性を検証した。

今後の課題は、多値化処理を上手く並列処理することで実行周期を短縮することである。

参考文献

- [長坂, 2011] 長坂保典: ロボカップの道しるべ: 第3回 小型ロボットリーグ, 情報処理, vol.52 No.1, pp.95–110, 2011.
- [升谷, 2010] 升谷保博, 成瀬正: 外部カメラを用いたヒト型ロボットによるサッカー競技 RoboCup Soccer SSL Humanoid, 人工知能学会誌, Vol.25, No.2, pp.213–219, 2010.
- [木村, 2010] 木村堯海, 升谷保博: 複数の外部カメラ画像に基づくヒト型ロボットの3次元形状の実時間取得 RoboCup Soccer SSL Humanoidの3次元情報サーバをを目指して, 第31回人工知能学会 AI チャレンジ研究会, pp.33–38, 2010.
- [植田, 2011] 植田康生, 木村堯海, 勢川友樹, 升谷保博: 外部カメラ画像から得られた複数のヒト型ロボットのボクセルデータに基づく行動決定, 第33回人工知能学会 AI チャレンジ研究会, pp.7–12, 2011.

- [M.Seiz, 1997] Steven M.Seiz, Charles R.Dyer: Photo-realistic Scene Reconstruction by Voxel Coloring, Computer Vision and Pattern Recognition Conf, pp.1067–1073, 1997.

RoboCup サッカーにおける敵位置の予測モデル構築

Constructing Prediction Models of Opponent Positions in RoboCup Soccer

山下雄大 † 中島智晴 † 秋山英久 ‡

Katsuhiko YAMASHITA † Tomoharu NAKASHIMA † Hidehisa AKIYAMA ‡

大阪府立大学 † 福岡大学 ‡

Osaka Prefecture University † Fukuoka University ‡

katsuhiko.yamashita@cs.osakafu-u.ac.jp, tohomaru.nakashima@kis.osakafu-u.ac.jp,

akym@fukuoka-u.ac.jp

Abstract

In this paper, we propose a method that predicts opponent player's positions. This method is used to generate decision making and achieved by using neural networks. Three-layered neural network are used to learn the opponent positions. One neural network learns the mapping from a field status to the position of an opponent player. Thus, 11 neural networks are necessary to learn opponent 11 players. The results by the numerical experiments show that the proposed method predicts opponent positions and helps to generate appropriate decision making.

1 はじめに

ロボット工学と人工知能の領域横断型研究プロジェクトとしてRoboCupが知られている。RoboCupには様々なリーグが存在しており、それぞれにおいて活発な研究、開発が行われている。RoboCup サッカーでは、競技で勝利することが重要視され、ただ単に勝利するだけではなく、ボールを支配し、確実に勝利することが望まれている。そのためには、チームでの試合運びを行うための戦術が必要である。高度な戦術を取るためには、敵プレイヤーの位置を予測することが必要であるが、現状ではそれは困難である。そこで本論文では、敵位置決定のモデル化について調査する。本論文では、RoboCup サッカーシミュレーション 2D リーグを題材とする。敵位置決定のモデルをサッカーフィールドの状況から敵プレイヤー位置へのマッピングである、と定義する。敵位置決定のモデル化にはニューラルネットワークを用いる。数値実験では、ニューラルネットワークの予測精度を調査し、プレイヤーに組み込むことで発生するチームへの影響を調査する。

2 行動探索

RoboCupにはサッカー、レスキュー、@ホームの他に、次世代のロボット技術者育成を目的としたジュニアリーグも存在する。本論文では、RoboCup サッカーシミュレーションを研究の対象とする。サッカーシミュレーションはモデル化の形式によって2Dリーグと3Dリーグに分けられる。本論文では2Dリーグを扱う。Figure 1に2Dリーグの試合の様子を示す。2Dリーグでは、二次元平面を仮想サッカーフィールドとし、円形のエージェントをプレイヤーとして競技を行う。また、プレイヤーやボールの位置と速度は全て二次元ベクトルとして表される。試合は1サイクル0.1秒で離散化され、前後半3000サイクルずつ合計6000サイクルで試合が行われる。各プレイヤーはそれぞれ独立したエージェントとしてプログラムされており、制限された視覚情報や聴覚情報からドリブルやパス等の意思決定を行う。

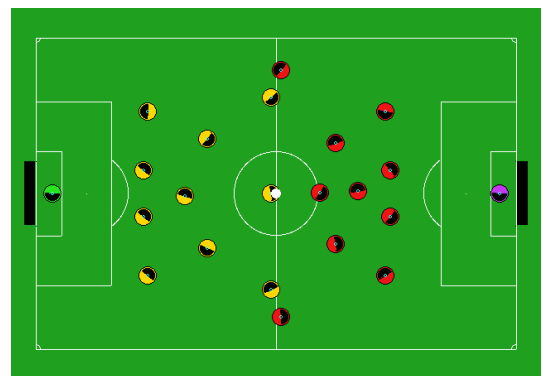


Figure 1: 2D リーグの試合の様子

プレイヤーが意思決定を的確にできるかどうかは、チーム戦略における重要な要素である。高レベルの意思決定に関しては、高度な戦略に基づいた試合が行われており、ポジショニング等に関する研究が積極的に行われている。

Luis ら[1] の, Situation Based Strategic Positioning を用いた手法や, Akiyama ら[2]の, Delaunay Triangulation を用いた手法などがある. 本論文で使用するプレイヤは, 最良優先探索を用いて行動連鎖と呼ばれる木構造を構築することで意思決定を行う[3]. まず, プレイヤの現在の状態をルートノードに入力する. 次に, ノードに入力された状態において実行可能な行動の候補(パス, ドリブル, シュートなど)を生成する. 生成された行動に対して評価値を計算し, その行動を実行した場合の予測状態と共に子ノードに追加する. ノードが追加されるたびに評価値が最大であるノードを選択し, そのノードにおける予測状態から再び実行可能な候補の行動を生成する. これを繰り返すことで, ノード数があらかじめ設定された最大値に達するまで探索木を成長させる. ただし, 木の深さがあらかじめ設定した値を越える場合や, ノードの予測状態から行動が生成できない場合, 行動連鎖の終了条件に設定されている行動(シュート)が生成された場合は, その葉ノードの子ノード生成は行わないものとする. 構築された木構造の中からノード列をつなげると, 行動連鎖が得られる. 評価値が最大となるノードから同様に実行可能な行動の候補を生成し, 探索木を成長させる. 最も評価値の高いノード列を選択することで, 戦略上良いと考えられる行動連鎖を実行することが可能となる.

行動連鎖の例を Figure 2 に示す. Figure 2 において, ポー

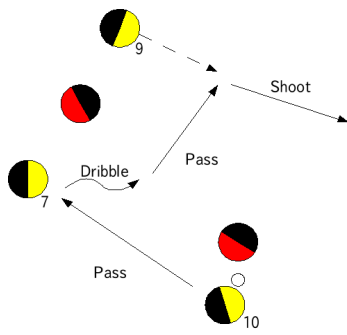


Figure 2: 行動連鎖の例

ルを保持している 10 番のプレイヤは以下のような行動連鎖を生成する. まず, 10 番のプレイヤが 7 番のプレイヤにパスを行う, パスを受け取った 7 番のプレイヤはドリブルによりボールを前へ運び, その後 9 番のプレイヤにパスを行う. パスを受け取った 9 番のプレイヤは, 相手ゴールに向かってシュートする. 10 番のプレイヤは以上の流れを考慮して, 7 番のプレイヤへのパスを実行する. 行動連鎖を生成することで, プレイヤは数手先の状況を考慮して, より戦略的価値が高い行動を選択することが可能となる.

成功確率を上げるには予測状態の精度向上が必要と予

想される. 予測の精度を上げるために各プレイヤの位置を正確に予測することが重要である. しかし, プレイヤは制限された情報しか保持していないため, 特に敵プレイヤの位置を正確に予測する事は困難である. オープンソースであるサンプルチームの agent2d [4] では, 行動連鎖生成時の敵プレイヤの予測位置は, 最後に自分が認識した敵プレイヤの位置で固定となっている. しかし, 実際には敵プレイヤはサイクル経過と共に動くため, 予測状態での敵プレイヤ位置と, 実際に行動を実行した後での敵プレイヤの位置に大きな差異が生じてしまい, それによって生成された行動連鎖の質は下がってしまう. そこで本論文では, ニューラルネットワークを用いて敵プレイヤの位置を予測し, 予測状態での敵プレイヤの位置を考慮する手法を提案する.

3 提案手法

本論文での提案手法では, ニューラルネットワークを用いた学習を行う. そこで, まずニューラルネットワークの概要について示す.

3.1 ニューラルネットワーク

ニューラルネットワークは, 人間の脳の神経回路をシミュレートしたモデルで, 高度な学習機能を備えている. ニューラルネットワークには様々な形状のものが存在するが, 本論文では三層階層型ニューラルネットワークを用いる. Figure 3 にその概形を示す. 三層階層型ニューラルネットワークには入力層, 中間層, 出力層が存在し, それぞれの層にユニットが存在する.

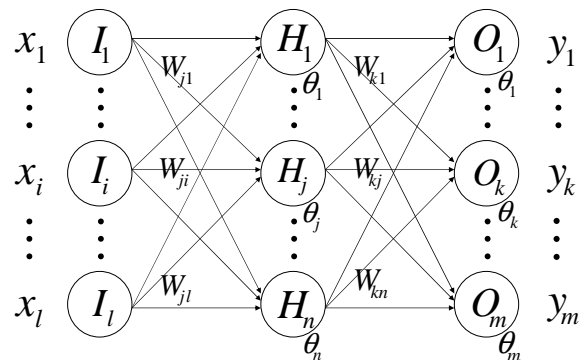


Figure 3: ニューラルネットワークの概形

Figure 3 において, 入力層ユニットを $I_i (i = 1, \dots, l)$, 中間層ユニットを $H_j (j = 1, \dots, n)$, 出力層ユニットを $O_k (k = 1, \dots, m)$ とする. $x_i (i = 1, \dots, l)$ は入力信号であり, $W_{ji} (i = 1, \dots, l, j = 1, \dots, n)$ は I_i と H_j の結合強度, $W_{kj} (j = 1, \dots, n, k = 1, \dots, m)$ は H_j と O_k の結合強度である. また, $\theta_j (j = 1, \dots, n)$ は H_j の閾値, $\theta_k (k = 1, \dots, m)$ は O_k の閾値を表し, $y_k (k = 1, \dots, m)$

は出力信号である．三層階層型ニューラルネットワークでは，入力層ユニットに入力信号 x_i が入力される．入力層ユニットの出力を $o_i (i = 1, \dots, l)$ とすると，

$$o_i = x_i \quad (1)$$

である．中間層ユニットでは入力層の出力 o_i を入力として，以下の式によって計算された $o_j (j = 1, \dots, n)$ を出力とする．

$$o_j = f(\text{net}_j) \quad (2)$$

$$\text{net}_j = \sum_{i=1}^n W_{ji} o_i + \theta_j \quad (3)$$

$$f(\text{net}_j) = \frac{1}{1 + e^{-\text{net}_j}} \quad (4)$$

出力層ユニットでは中間層の出力 o_j を入力として，以下の式によって計算された $o_k (k = 1, \dots, m)$ を出力とする．

$$o_k = f(\text{net}_k) \quad (5)$$

$$\text{net}_k = \sum_{j=1}^n W_{kj} o_j + \theta_k \quad (6)$$

$$f(\text{net}_k) = \frac{1}{1 + e^{-\text{net}_k}} \quad (7)$$

ニューラルネットワークの出力信号を y_k とすると，

$$y_k = o_k \quad (8)$$

である．ニューラルネットワークの学習には，次に示す誤差逆伝搬学習アルゴリズムを用いる．

3.2 誤差逆伝搬学習アルゴリズム

誤差逆伝搬学習アルゴリズムは，入力信号に対する望ましい出力値を教師信号として与え，出力値との誤差を最小化するように結合強度と閾値を更新するアルゴリズムである．結合強度の更新は学習と呼ばれ，出力値 o_k に対する教師信号 t_k との差を以下の評価関数で表し，これが最小となるように行われる．

$$E = \sum_{k=1}^m \frac{1}{2} (t_k - o_k)^2 \quad (9)$$

式 (9) より，結合強度 W_{ji} ， W_{kj} および閾値 θ_j ， θ_k の修正量は，学習係数を η とすると，それぞれ以下の式で表される．

$$\Delta W_{ji} = \eta \cdot \left(-\frac{\partial E}{\partial W_{ji}} \right) = \eta \delta_j o_i \quad (10)$$

$$\Delta W_{kj} = \eta \cdot \left(-\frac{\partial E}{\partial W_{kj}} \right) = \eta \delta_k o_j \quad (11)$$

$$\Delta \theta_j = \eta \cdot \left(-\frac{\partial E}{\partial \theta_j} \right) = \eta \delta_j \quad (12)$$

$$\Delta \theta_k = \eta \cdot \left(-\frac{\partial E}{\partial \theta_k} \right) = \eta \delta_k \quad (13)$$

$$\delta_j = o_j (1 - o_j) \sum_{k=1}^m \delta_k W_{kj} \quad (14)$$

$$\delta_k = (t_k - o_k) o_k (1 - o_k) \quad (15)$$

ここで，学習速度を向上させるために，慣性項係数を α として，修正量を次のようにする．

$$\Delta W_{ji} = \eta \delta_j o_i + \alpha \Delta W_{ji}^{old} \quad (16)$$

$$\Delta W_{kj} = \eta \delta_k o_j + \alpha \Delta W_{kj}^{old} \quad (17)$$

$$\Delta \theta_j = \eta \delta_j + \alpha \Delta \theta_j^{old} \quad (18)$$

$$\Delta \theta_k = \eta \delta_k + \alpha \Delta \theta_k^{old} \quad (19)$$

したがって，結合強度と閾値は次のように更新される．

$$W_{ji}^{new} = W_{ji}^{old} + \Delta W_{ji} \quad (20)$$

$$W_{kj}^{new} = W_{kj}^{old} + \Delta W_{kj} \quad (21)$$

$$\theta_j^{new} = \theta_j^{old} + \Delta \theta_j \quad (22)$$

$$\theta_i^{new} = \theta_i^{old} + \Delta \theta_i \quad (23)$$

3.3 提案手法

試合中における通常プレイが行われている状態を playon と呼ぶ．敵プレイヤーの位置予測を行うニューラルネットワークを 1 プレイヤにつき 1 つ用意し，playon 時の敵プレイヤーの位置を学習させる．ニューラルネットワークの入力として，

- 現在のボール位置
- ボールの到達予想位置
- 最も近い敵プレイヤーがボールの位置に到達するまでに要するサイクル数

を用いる．これらは予測状態を生成するのに必要な情報と考えられる．教師データは，試合の前半における playon 時の敵エージェントの x 座標と y 座標とする．ただし，ボールに最も近い敵プレイヤーは，その性質上ボールに向かって移動することが多いため，本来のフォーメーションでの動きとは別の動きをとると見なし，教師データからは除外する．各入力データは正規化された後にニューラルネットワークに入力される．各 x 座標は次のように x_{input} へと正規化される．ここで， $PitchWidth$ はフィールドの縦の幅 105m に，10m の余裕を持たせたものである．

$$x_{input} = \max\left(\frac{x}{PitchWidth} + 0.5, 1.0\right) \quad (24)$$

各 y 座標は次のように y_{input} へと正規化される．ここで， $PitchLength$ はフィールドの横の幅 68m に，10m の余裕を持たせたものである．

$$y_{input} = \max\left(\frac{y}{PitchLength} + 0.5, 1.0\right) \quad (25)$$

到達サイクル数 $cycle$ は次のように $cycle_{input}$ へと正規化される。

$$cycle_{input} = \max\left(\frac{cycle}{20}, 1.0\right) \quad (26)$$

到達サイクル数は 20 サイクルを越えるものはほとんど見られないため、20 サイクル以上は 1.0 とみなす。

ニューラルネットワークの学習は、試合中の各サイクルでのフィールド上の完全な情報を保持できるコーチエージェントによって行われる。コーチエージェントはハーフタイム中に各敵プレイヤーに対して順番にニューラルネットワークを学習させる。学習を行ったニューラルネットワークに予測状態時点での各敵プレイヤーの位置を出力させ、敵プレイヤーのフィールド上での位置を予測する。学習したニューラルネットワークの情報は、コーチから各プレイヤーに送られ、プレイヤーはその受け取った情報からニューラルネットワークを構築し、行動連鎖生成時の予測状態での敵プレイヤーの位置推定に使用する。行動連鎖生成時に使用される行動実行後の予測状態における敵プレイヤーの位置を、ニューラルネットワークで得られる敵プレイヤーの位置にすることで、より正確な予測状態を作成することを狙う。またそれにより、戦術的価値のより高い行動連鎖を生成する。

3.4 予備実験

提案手法で必要な、中間層ユニット数、学習係数 η 、慣性項係数 α 、学習のエポック数の各パラメータの値を予備実験により決定した。予備実験では、RoboCup2012 決勝戦の HELIOS2012 のログデータを用いる。ニューラルネットワークの入力は、現在のボールの x 座標と y 座標、出力は、背番号 11 のプレイヤーの x 座標と y 座標と設定し学習を行う。学習には誤差逆伝搬学習アルゴリズムを用いる。

3.4.1 中間層ユニット数

エポック数を 10000、 $\eta = 0.7$ 、 $\alpha = 0.3$ と固定し、中間層ユニット数を 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 14, 20 と変化させて学習を行い、実際の位置との誤差を出力させた。Table 1 に各中間層ユニット数ごとの距離の誤差を示す。Table 1 より、中間層ユニット数は 10 個が最適であると分かる。

3.4.2 学習率

エポック数を 10000、中間層ユニット数を 10 個、 $\alpha = 0.3$ と固定し、 η の値を 0.1 ~ 0.9 の範囲で変化させて学習を行い、実際の位置との誤差を出力させた。Table 2 に η の値ごとの誤差を示す。Table 2 より、 η の値は 0.9 が最適であると分かる。

3.4.3 慣性項係数

エポック数を 10000、中間層ユニット数を 10 個、 $\eta = 0.7$ と固定し、 α の値を 0.0 ~ 0.3 の範囲で変化させて学習を

Table 1: 中間層ユニット数の変化による誤差

中間層ユニット数	誤差
3	21.80
4	22.19
5	22.50
6	22.76
7	22.97
8	23.15
9	23.30
10	16.40
11	16.48
12	16.55
14	16.66
20	16.89

Table 2: 学習率の変化による誤差

学習率	誤差
0.1	65.51
0.2	45.07
0.3	33.92
0.4	27.00
0.5	22.30
0.6	18.92
0.7	16.40
0.8	14.44
0.9	12.89

を行い、実際の位置との誤差を出力させた。Table 3 に各中間層ユニット数ごとの誤差を示す。Table 3 より、 α の値は 0.3 が最適であると分かる。

Table 3: 慣性項係数の変化による誤差

慣性項係数	誤差
0.0	17.63
0.1	17.05
0.2	16.47
0.3	16.40

3.4.4 エポック数

中間層ユニット数を 10 個、 $\eta = 0.7$ 、 $\alpha = 0.3$ と固定し、エポック数を 100, 1000, 2000, 5000, 10000, 20000, 40000, 100000, 300000 と変化させて学習を行い、実際の位置との誤差を出力させた。Table 4 に各中間層ユニット数ごとの誤差を示す。Table 4 より、エポック数は 2000

で十分学習が行われていると考えられる。エポック数が極端に少ない場合に誤差が小さくなっているように見えるのは、ニューラルネットワークの初期値の影響により、学習が進んでいないためであると考えられる。

Table 4: エポック数の変化による誤差

エポック数	誤差
100	15.72
1000	16.29
2000	16.34
5000	16.38
10000	16.39
20000	16.40
40000	16.41
100000	16.41
300000	16.41

4 数値実験

数値実験では、提案手法を用いた敵プレイヤーの位置予測の精度と、提案手法をプレイヤーに組み込んだ場合のチーム性能を調査する。ニューラルネットワークの入力層ユニット数、中間層ユニット数、学習係数、慣性項係数の各パラメータは、予備実験の結果から、入力層ユニット数を5、中間層ユニットを10、学習係数を0.3、慣性項係数を0.9と設定する。また、入力情報は、現在のボールの x 座標と y 座標、ボールに最も近い敵エージェントがボールに到達すると予測されるサイクル数、そのサイクル数が経過したときのボールの予想到達位置の x 座標と y 座標とする。出力値は敵プレイヤーの x 座標と y 座標とする。また、ボールに最も近い敵プレイヤーは、ボールを取りにいこうとするためにボールに向かってくることが多く、本来のフォーメーションでの位置から外れた動きをすることが多いので、ニューラルネットワークでの学習要素からは除外する。

4.1 予測精度の評価

提案手法での敵プレイヤーの位置予想の精度を調査する。コーチエージェントは前半3000サイクルまでのplayon時の各敵プレイヤーの位置を記憶しておく。ニューラルネットワークの教師データは、前半3000サイクルまでのplayon時に得られた敵プレイヤーの位置情報から生成され、ハーフタイム中に各敵プレイヤーに対して2000エポック学習する。コーチエージェントが試合の後半のplayon時にニューラルネットワークに現在の状態を入力して得られる予測された各敵プレイヤーの位置と、実際に予測されたサイクルになった時点での各敵プレイヤーの位置の比較を行い、それらの誤差を計測する。

4.2 プレイヤに組み込む場合の評価

各プレイヤーが保持しているニューラルネットワークから出力された敵プレイヤーの予測位置を、行動連鎖生成時の予測状態に組み込んだチームで試合を実行し、行動連鎖1段階目の予測状態における各敵プレイヤーの位置と、実際に行動連鎖1段階目の行動を実行した後での各敵プレイヤーの位置との誤差を計測する。同様の誤差の比較を、提案手法をプレイヤーに組み込んでいないチームでも行い、これらを比較することで、提案手法の効果を確認する。こちらの実験では、試合開始時点で各プレイヤーに対して、あらかじめ敵チームのプレイヤーの位置を予測するためのニューラルネットワークを与えておき、コーチエージェントがハーフタイムに作成したニューラルネットワークへの更新は行わないものとする。

自チームは、RoboCup2012に出場したHELIOS2012 [5]とする。RoboCup2012に出場したチームとagent2d [4]を相手にそれぞれ10試合ずつ行う。

4.3 実験結果

提案手法を組み込んだHELIOS2012で試合を行い、5.1章で述べた評価方法を用いて予測精度を評価した。計測された誤差のゴールキーパーを除いたフィールドプレイヤーの平均と標準偏差をTable 5に示す。なお、誤差の平均の単位はメートルとする。

Table 5: 予測の精度

対戦相手	誤差の平均	誤差の標準偏差
agent2d	10.53	7.89
WrightEagle	11.88	8.04
MarliK	11.07	9.01
Ri-one	11.26	6.84
Oxxy	13.62	9.21
YuShan	13.57	9.30
NADCO-2D	10.85	8.00

Table 5より、どの相手チームでニューラルネットワークを用いて学習を行っても、各敵プレイヤーに対して十数メートルの誤差で予測することができることが分かった。ゴールキーパーは、その性質上あまり激しく動き回らずに定位置にすることが多いため、誤差の平均はどのチームも2メートルから5メートルの範囲で収まっている。また、相手チームやそのポジションによっては、誤差の平均値が15メートルを超える場合もあることが分かった。これは、そのプレイヤーが、チーム内でボールに積極的にからみにいくことで、フォーメーションから外れた動きをすることが多いために生じると考えられる。

次に、提案手法を組み込んだ場合のチームに対する効

果を, 5.2 章で述べた評価方法を用いて評価した. 得られたフィールドプレイヤーの誤差の平均値を Table 6 に, その標準偏差を Table 7 に示す.

Table 6: 行動連鎖 1 段目における誤差の平均

対戦相手	提案手法なし	提案手法あり
agent2d	15.00	17.80
WrightEagle	15.06	17.56
MarliK	27.55	24.97
Ri-one	12.92	16.89
Oxxy	18.27	21.40
YuShan	10.51	16.65
NADCO-2D	19.55	20.06

Table 7: 行動連鎖 1 段目における誤差の標準偏差

対戦相手	提案手法なし	提案手法あり
agent2d	78.28	27.14
WrightEagle	101.06	25.30
MarliK	100.17	31.25
Ri-one	70.30	16.10
Oxxy	78.86	16.82
YuShan	44.19	18.20
NADCO-2D	84.31	28.61

Table 6 から, 提案手法を組み込んだ場合と組み込んでない場合を比較すると, 誤差の平均値だけで見ると提案手法はあまり効果がないように思われる. しかし Table 7 から, 提案手法を組み込んだ場合と組み込んでない場合の標準偏差を比較すると, 提案手法を組み込んだほうが小さくなっていることが分かる. 提案手法を組み込んでいないほうの誤差について細かく見てみると, ほぼ一致している場合と, 大幅にずれている場合の二極化していることがわかった. これは, 提案手法を組み込んでいない場合は, 敵プレイヤーの位置を最後に認識した場所で固定してあるため, 実際に動いていなかった場合は誤差が小さく, 動いていた場合は非常に大きな誤差が出てしまうためであると考えられる. 提案手法の標準偏差が小さいのは, 敵の動きを考慮して予測しているため, 大幅なずれが生じていないからであると考えられる.

5 おわりに

本論文では, ニューラルネットの誤差逆伝搬法を用いて敵エージェントのフォーメーションのモデル化を行い, 実データから学習することで既知のチームの位置予測が可能であることを示し, 行動連鎖への組み込みで予測性能

を検証した. 今後の課題としては, 位置予測の精度の向上や, 行動連鎖に組み込んだ際のチームのパフォーマンスについて細かく分析を行うことが挙げられる.

参考文献

- [1] Luis Paulo Reis, Nuno Lau and Eugenio Oliveira, "Situation Based Strategic Positioning for Coordinating a Simulated RoboSoccer Team", *Balancing Reactivity and Social Deliberation in MAS*, 175–197, 2001
- [2] Hidehisa Akiyama and Itsuki Noda, "Multi-Agent Positioning Mechanism in the Dynamic Environment", *RoboCup 2007 : Robot Soccer World Cup XI*, 2008.
- [3] Hidehisa Akiyama, Tomoharu Nakashima and Shigeto Aramaki, "Online Cooperative Behavior Planning using a Tree Search Method in the RoboCup Soccer Simulation", *Proceedings of 4th IEEE International Conference on Intelligent Network and Collaborative Systems*, 2012.
- [4] agent2d, <http://rctools.sourceforge.jp/pukiwiki/index.php?agent2d>
- [5] Hidehisa Akiyama, Hiroki Shimora, Tomoharu Nakashima, Yosuke Narimoto, Katsuhiko Yamashita, *HELIOS2012 Team Description Paper, RoboCup2012*, CD-ROM(6 pages), Mexico City, Mexico, 2012.

ロボカップ小型リーグにおける戦略改善のための 画像処理によるボールの回転状態の推定

A Method to Estimate Ball's State of Spin by Image Processing

for Strategic Learning in RoboCup Small-Size-robot League

布目裕司, 村上和人, 小林邦和, 成瀬正

Yuji Nunome, Kazuhito Murakami, Kunikazu Kobayashi and Tadashi Naruse

愛知県立大学大学院 情報科学研究科

Graduate School of Information Science and Technology, Aichi Prefectural University

im132007@cis.aichi-pu.ac.jp, murakami, kobayashi, naruse@ist.aichi-pu.ac.jp

Abstract

A spun ball varies its speed after the ball bounced off the floor. It will be very useful to prevent disadvantageous situation if the ball's state of spin is estimated before the ball is kicked. This paper proposes a method to estimate the ball's state of spin by using inertia feature of co-occurrence matrix of the image sequences and shows the effectiveness of our proposed method by some experiments. Furthermore, this paper discusses the influences of partial occlusion and blur of the ball, and its application to the strategic learning for RoboCup small size robot league.

1 はじめに

ロボカップ小型リーグ[1]では、小型車輪ロボット 6 台対 6 台でサッカーを行う。最近では、ほとんどのチームのロボットは 2 種類のキック、すなわち、ボールが地面を転がるグラウンダーキックと、ボールを浮かせてロボットの上を飛び越えるチップキックの機能を備えている。また、ボールに強力な回転をかけてボールを保持するドリブルローラと呼ばれる機構を持つ。ボールに回転をかけたままキックする回転付きチップキックは、ボールの回転によってバウンド後にボールの速度が大きく変化する。そのため、ボールの軌跡を基に計算したボール位置の予測は困難になる。もし、ボールが回転しているかどうかを画像処理で判断することができれば、ロボットの配置を変えることによって、敵チームが有利になるような展開を防ぐことができると考えられる。

画像からのボールの回転状態の推定に関する研究は、ロ

ボカップ小型リーグではこれまでほとんど報告されていない。野球など、他の球技ではいくつか報告されている。井上らは、高速カメラを用いて硬式野球ボールの回転推定をするシステムを提案している[2]。彼らのシステムでは、硬式野球ボールの表面の縫い目を手がかりにボールの回転軸ベクトルと回転速度の推定を行っている。しかし、ロボカップ小型リーグでは公式球として使用しているゴルフボールには、野球ボールの縫い目のような特徴はないため、同様の手法で回転を推定するのは難しい。また、高速カメラは一般的なカメラより高価であり、隠蔽の発生のないように多方向から撮影するのは現実的ではない。

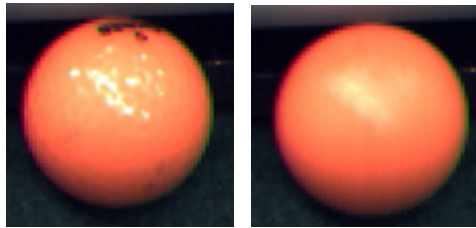
回転しているボールをカメラで撮影するとき、ブラーが発生する。ブラーの原因は、ゴルフボール表面のディンプルである。また、ブラーの大きさは、カメラの露出時間に応じて変動する。そこで本論文では、カメラ入力された画像の濃度共起行列を求め、ここから計算した慣性特徴量を用いてボールの回転状態を推定する一つの手法を提案する。

以下、本論文では、2 章で回転状態推定の基本アイデアについて、処理概要とともに述べる。3 章で濃度共起行列を用いたボールの回転状態推定手法について説明し、実験結果とともに提案手法の有効性を示す。4 章では、ボールの隠蔽やブラーの回転状態推定への影響、ロボカップ小型ロボットリーグの戦略への応用について考察する。

2 基本アイデア

ロボカップ小型リーグでは、オレンジ色のゴルフボールを使用している。ゴルフボールにはディンプルと呼ばれる小さなくぼみが多数、存在する。ディンプルにより光の反射強度が変化する。さらにボールが回転するとブラーが生じ、反射の様子が変化する。回転の有無による見え方の違いを図 1 に示す。図 1(a) は静止状態の画像の一例であり、高輝度の部分が何箇所か見られる。一方、図 1(b) は回転状

態の画像の一例である。回転状態では強く反射していた光がぼやけたように見える。この光の反射の分布の違いをとらえることにより、ボールの回転状態を推定できると考えられる。



(a) static ball (b) spinning ball

Figure 1 Differences of the images.

ボールの回転状態を推定するため、ボールの表面を確認するカメラを追加する。図 2 にシステム構成の一例を示す。図 2(a) の”Additional Camera”は、例えば、PTZ カメラを使用する。データの流れを図 2(b) に示す。現在のロボカップ小型リーグでは、SSL-Vision[3] と呼ばれるビジョンシステムが各チームのコンピュータにボールとロボットの位置情報を送っている。これらに加えて、”Additional Camera”からの画像を処理することにより、ボールの回転情報を得る。ボールとロボットの位置情報、および、ボールの回転情報をもとに、ボールのイレギュラーなバウンドに対応した戦略が可能になる。

3 ボールの回転状態推定手法

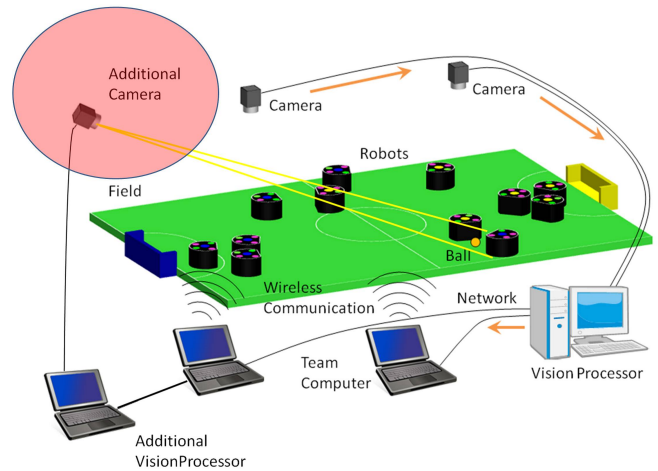
図 3 にボールの回転状態推定手法の流れを示す。本手法は、3 つのステップに分けることができる。*Step1* で処理領域を抽出する。*Step2* で濃度共起行列を作成し、慣性を計算する。*Step3* で慣性の値を閾値処理し、ボールの回転状態を推定する。以降に 3 つのステップの詳細を記す。

Step1. ボール領域の抽出

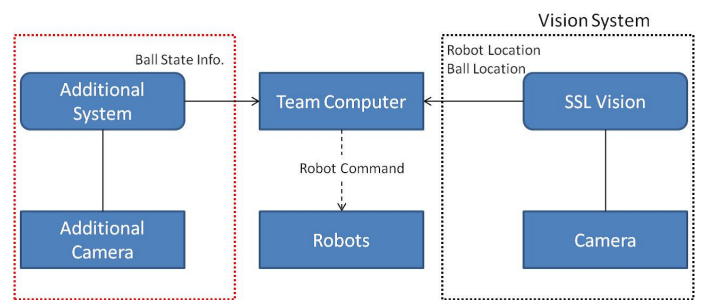
映像の各フレーム (図 4(a)) を 2 値化する。2 値化画像をラベリングし、2 番目に大きい領域をボールの領域とする (図 4(b))。領域の左上座標と右上座標からボールの映っている範囲を抽出する (図 4(c))。抽出したものを処理領域とする。

Step2. 濃度共起行列を用いた特徴量の計算

処理領域の RGB 表色系の B チャンネルの輝度値から濃度共起行列 $P_\delta(i, j)$ を作成する。ここで δ は 2 つの画素の x 座標の差分 D_x と y 座標の差分 D_y を $\delta = (D_x, D_y)$ のようにベクトル表示したものであり、 i, j はそれら 2 つの画素のそれぞれの輝度値を表



(a) cameras' layout



(b) data flow of vision system

Figure 2 System configuration.

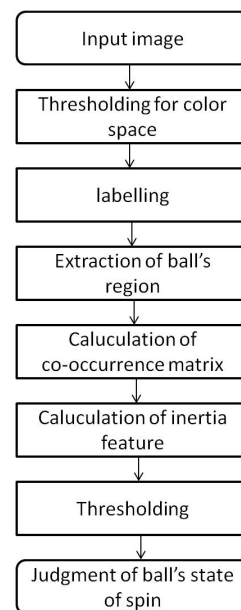


Figure 3 Flowchart of image processing.

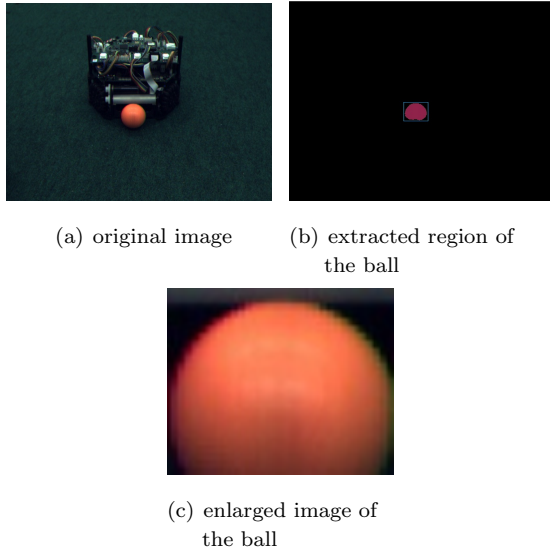


Figure 4 Extraction of ball's region.

す。慣性 Ine は、濃度共起行列 $P_{\delta}(i, j)$ から、

$$Ine = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} (i - j)^2 P_{\delta}(i, j) \quad (1)$$

のように計算する。

Step3. ボールの状態推定

慣性 Ine に対して、次式のように閾値処理を行うことで入力フレームでのボールの状態を推定する。

$$BallState = \begin{cases} StaticState & (Ine \geq Threshold) \\ SpinningState & (Ine < Threshold) \end{cases}$$

ボールが回転/停止を繰り返す映像に対する慣性特徴 Ine の変化の様子の一例を図5に示す。図から明らかなように、回転/停止によって Ine の値が大きく変化していることがわかる。ここで、カメラは Basler 社の A601fc[4]を使用した。また、 $\delta = (D_X, D_Y)$ は、映像中のボールのディンプルの間隔を基に $\delta = (3, 0)$ とした。

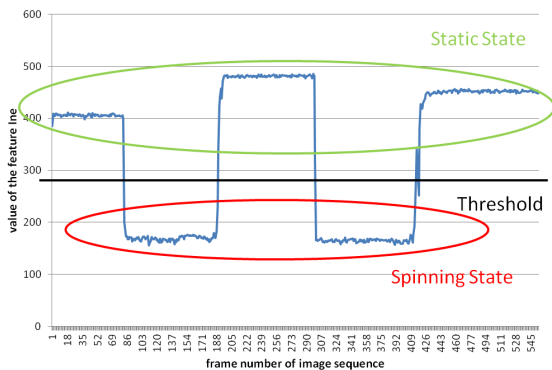


Figure 5 An example of the changes of the inertia feature for an image sequence (spin and stop is reported).

4 考察

4.1 濃度共起行列の特徴量

濃度共起行列では他にもエネルギー、エントロピー、相関などの特徴量の計算ができる。濃度共起行列のエネルギー、慣性、エントロピー、相関の4種類の特徴量を計算したものを図6に示す。エントロピーと相関は、ボールが回転しているフレームで多少の変化がみられるが、慣性の変化である図6(c)がボールの回転に対してもっとも顕著であることは明らかである。

4.2 ボールの隠蔽の影響

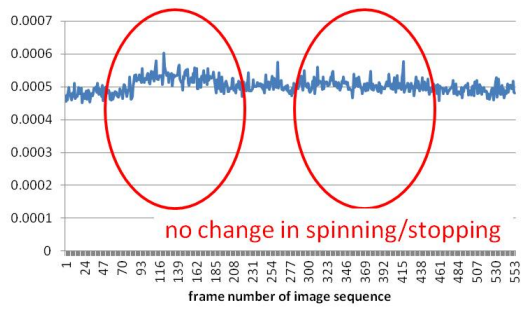
ボールがロボットなどの陰に入ってしまうとボールが隠蔽されてしまった場合を考える。図7はボールが隠ぺいされている映像に推定手法を適応したものである。図7(c)を見ると、回転状態と静止状態で慣性の値の変化が小さくなっていることが分かる。適当な閾値を設定できれば回転の有無を判断することができるが、変化が小さいため推定に失敗する可能性も高くなる。

4.3 ブラーの影響

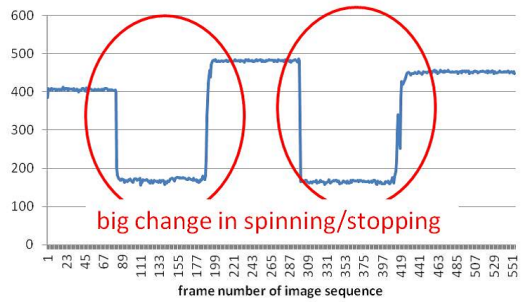
撮影に首振りカメラを使用すると、首振りに応じてブラーが生じる。また回転状態のボールを保持したロボットが左右に移動する場合もブラーが生じる。この映像内で155フレームからロボットが移動しブラーが発生している。図8(b)をみると155フレーム以降から慣性 Ine の値が小さくなっていることが分かる。ブラーやロボットの移動に伴うボールのぶれなどが原因でこのように値が小さくなると考えられる。

4.4 ボールの回転速度との関係

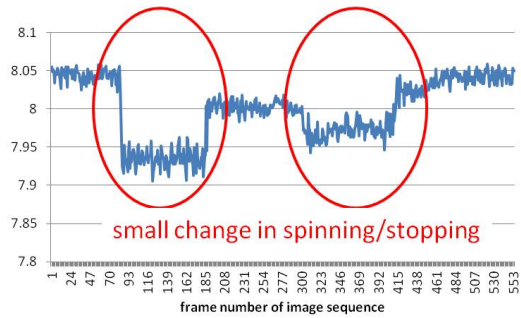
ドリブルローラーの速度と慣性 Ine との関係を調べるため、ドリブルローラーの速度を変えて実験を行った。回転速度とレベルの関係を表1に示す。また、回転速度の変化に伴う慣性の変化の結果と、慣性の変化に曲線当てはめを行い近似曲線を書いたものを図9に示す。近似曲線から明らかなように、回転を速めるにしたがって Ine の値は減少していくことが分かる。したがって、試合の会場でこの曲線をモデル化できれば、ボールの回転速度を推定できる可能性がある。



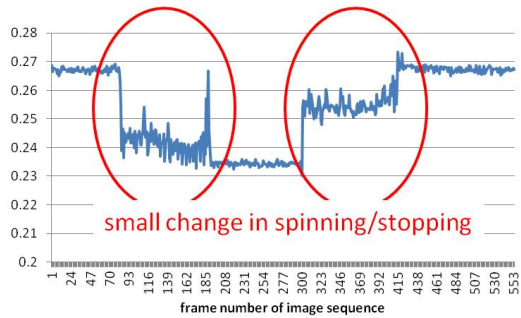
(a) energy



(b) inertia

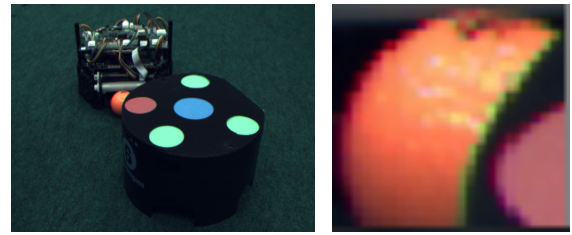


(c) entropy



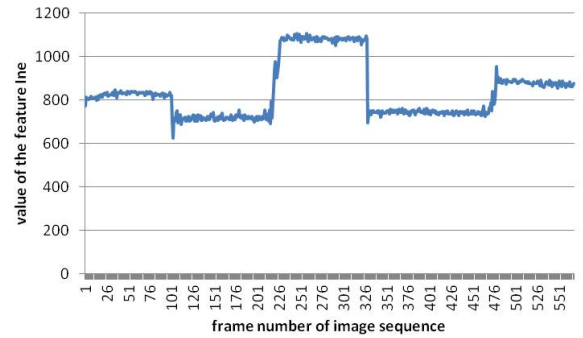
(d) correlation

Figure 6 Examples of the changes of the features for an image sequence (spin and stop is reported).



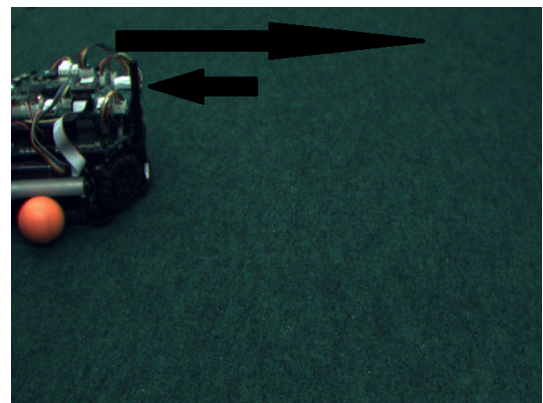
(a) occluded scene

(b) extracted ball region

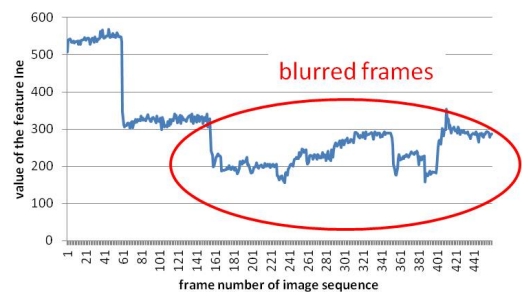


(c) changes of inertia feature

Figure 7 An example of occluded scene



(a) blurred ball's image



(b) changes of inertia feature

Figure 8 An example of the result for a blurred image sequence.

Table 1 The relation between the level and the rotation speed of dribble roller.

回転レベル	回転速度 (rps)
Level1	0.6
Level2	1.0
Level3	3.1
Level4	4.0
Level5	5.1
Level6	6.0
Level7	6.9
Level8	14.6
Level9	20.0
Level10	22.9
Level11	23.8
Level12	28.2
Level13	31.8

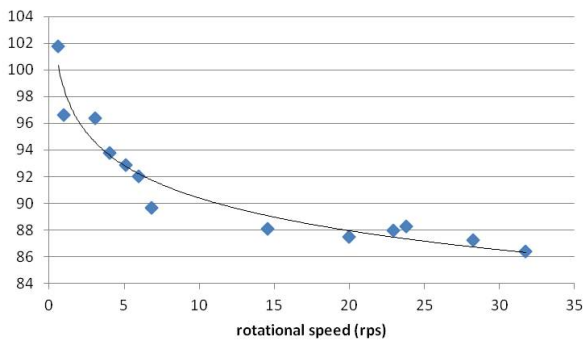


Figure 9 An experimental result of the inertia feature *Ine* and the rotation speed.

5 終わりに

本論文では、濃度共起行列の慣性特徴量からボールの回転状態を推定する手法を提案した。具体的には、フレームごとの慣性の値を閾値処理することにより、ボールの静止状態と回転状態を区別できることを示し、手法の有効性を実験的に確認した。また、ボールの隠蔽やブラーのある映像に対しても提案手法が有効に機能することを確認した。さらに、ボールの回転速度によって慣性の値に変化が現れることから、慣性の値を用いた回転速度の推定の可能性を示した。

提案手法は筆者らのプロトタイプシステムでは有効に機能しているものの、照明環境の変化に対するロバスト性を確認する必要がある。また、ボールの回転速度を推定し、バウンドした後のボールの軌跡を推定する手法を確立し、

戦略に適用することも今後の課題である。

参考文献

- [1] "Small Size Robot League - start"
<http://robocupssl.cpe.ku.ac.th/>
- [2] Inoue,T., Uematsu,Y., & Saito,H. ,"Estimation of Rotational Velocity of Baseball Using High-Speed Camera Movies",The transactions of the Institute of Electrical Engineers of Japan. D, A publication of Industry Applications Society, Vol.131, No.4, pp.608-615(April 2011).
- [3] "Small Size Robot League - sslvision:"
<http://robocupssl.cpe.ku.ac.th/sslvision>
- [4] "Basler Industriekameras - A600 Serie - A601fc:"
<http://www.baslerweb.com/products/A600.html?model=311>

Motion Optimization for the RoboCup 3D Soccer Simulation

¹Vallade Benoît, ¹Takeshi Sumitani, ¹Tomoharu Nakashima¹
²Takeshi Uchitane and ²Toshiharu Hatanaka

Department of Computer Science and Intelligent Systems, Osaka Prefecture University

{valladeben@cs, takeshi.sumitani@ci.cs, tomoharu.nakashima@kis}.osakafu-u.ac.jp

²Department of Computer Information and Physical Sciences, Osaka University

{hatanaka, t-uchitane}@ist.osaka-u.ac.jp

Abstract

This paper addresses a motion optimization method for RoboCup 3D Soccer Simulation's robots using Particle Swarm Optimization (PSO). A robot's motion is modelled by the motion trajectories of the robot using some parameters. The trajectory equations' parameters are adjusted to obtain the best motion. We use particle swarm optimization algorithms to tune the trajectory parameters .

1 Introduction

Nowadays, one of the major robotic research focuses on humanoid robots and their relationships with human and/or other robots. In the aim to give a common target and problem to all laboratories, the RoboCup has been created. This worldwide project is divided in lots of categories such as rescue and home. Among these categories, the first to have been created was the soccer category. The RoboCup Soccer's objective is to create and improve robots and their behaviours to be able to play to soccer [1] The major research points of this category are the improvement of the robot's motion and their communication. The objective is to be able to play and win against the world-cup championship team by 2050.

The RoboCup Soccer category is further divided into several leagues according to the regulation of robots: Humanoid, middle-size, small-size, standard platform, and simulation. The simulation league focuses on the intelligent aspect of soccer robots as it employs virtual soccer players of computer program that play soccer in a virtual soccer field. There are two sub-leagues in the simulation league depending on the specification of

the virtual field: 2D and 3D. While the 2D league has no concept of height as every object is represented as a two-dimensional vector, the 3D league has a more realistic virtual environment: The soccer field is placed in a three-dimensional space and soccer robots are built in the virtual field by joining multiple elemental parts with joints. Thus the soccer robots are more realistic and more complicated than 2D soccer robots. The development of 3D robots faces many problems. One of the major problems is the motion optimization. We need to improve robot motions such as walk, kick, dribble, etc., to obtain stable, quick and efficient movements. But the variety of robot's architecture, the number of parameters needed to configure a motion and the dependencies between the parameters of its motion are the root cause of the complexity of the optimization problem. To be able to answer to these problems we need to create a software able to cope with the different robots configuration and research the best values for the set of parameters in a huge research space.

2 Motion Optimization

A widely used way to optimize robot's motion is to use inverse kinematics. We could create a motion by setting all the engines of the robot, but there are too many parameters. So in order to reduce the number of parameters, we choose some points of the robot and search to optimize their trajectory. After having determined these trajectories by setting a value to the equation's parameters we use mathematics rules to deduce the trajectory of each engine used in this motion. This method is ever used by some other

teams; the innovations are to use the PSO as an evolution strategy [2] and to try to create adaptable software.

2.1 Target Trajectories

The first step is to choose the target points on the robot. These points will determine the motion that we want to obtain. For example, in case of a walk motion, the focal points will be the end of the robot's foots and the centre of the hip (see Fig. 1) [3]

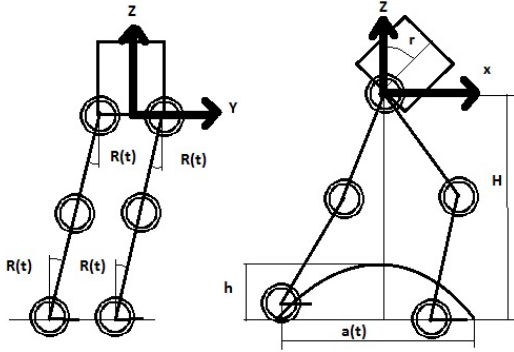


Fig. 1. Choice of target points and trajectories for a walk motion.

After having chosen the points, the second step is to choose their target trajectories. These trajectories will create the desired movement. They are represented by a movement equation such as linear movement or oscillators' movement and can be coupled together. To keep the example of the walk motion, here an equation of movement for one of the tree target points of this motion:

$$\dot{\varphi}_i(t) = \omega_i + \sum_{j(j \neq i)}^3 \omega_{ij} \sin(\varphi_j(t) - \varphi_i(t) + \delta\theta_{ij}) \quad (1)$$

Finally we obtain some equations which are dependant of a set of parameters. And to fully determine these equations we need to put value on these parameters.

2.2 Parameter Optimization

The precedent equations depend on parameters which can be separate in two kinds. There are the parameters whose values are constant, like the size of legs, arms or foot of the robot. They are based on the robot's architecture. The second kind is the

parameters which will be tune by the research algorithm. Here an example of set of parameters (for walking motion)

Parameter name	min	max
$\omega_1 (= \omega_2 = \omega_3)$	2.0	10.0
$\omega_{12} (= \omega_{21} = \omega_{13} = \omega_{31})$	-1.0	1.0
$\omega_{23} (= \omega_{32})$	-1.0	1.0
a_{max}	0.0	$2\sqrt{(L1 + L2)^2 - H^2}$
a	0.0	a_{max}
St	0.0	a_{max}
h	0.0	$\frac{L1 + L2}{2}$
H	$\frac{L1 + L2}{2}$	$L1 + L2$
r	0.0	30.0
$roll_{max}$	0.0	10.0
P_{gain}	0.0	5.0
D_{gain}	-5.0	5.0

Table. 1. List of parameters for walking motion.

Each parameter has a value and a definition space. The limits of these definition spaces are given by constant values or expressions which depend on the value of other parameters.

In order to give a value to this set of parameters the second step uses a research algorithm. This algorithm will progress in a research space define by crossing all the definition spaces. It will search the best combination of value for the set of parameters.

The simulator will be use to evaluate each solution. We will launch a simulation for each solution and get data about robot's performance and environment state. The research algorithm will use these data to evaluate the quality of each solution.

2.3 Inverse kinematics

During the third step we use the inverse kinematics in aim to find the movement parameters (angle, velocity, centre... etc) of each engines used in the motion (knee, hips... etc).

We deduce these parameters from the robot architecture, the successive positions of the target points and by using some mathematical rules like Pythagoras, and trigonometry.

3 Software Design

An important aspect of this software is that is still in development and even after being finished it could be subject to improvement (like parallelization, complex target motion). That is why we have decided to set up the design in three modules.

The first module is the PSO module. This one will be the subject of further discussions, so we won't describe it now.

The second module has many objectives; first it should be able to represent the architecture of the robot. The software will be able to use the xml file representing the robots for the simulator, but we also think to insert a module which allows the user to enter manually via a graphical interface the configuration of the robot. After the robots defined by the system, the user will be able to choose some target points on the robot and their target trajectories and indicate if they are coupled. The equation of these trajectories will be automatically generated as well as the corresponding set of parameters. Afterward the user will manually choose the definition space (max and min) of each parameter. The aim of this paper is that these values can be constant as well as an equation. This implies the set up of an equation parser. This one is for the moment in development but will not be the subject of this paper. As last but main objectives, the third module will have the ability to use the solution returned by the PSO module and to use the inverse kinematics to define the motion parameters of each engine used in the movement and generate a motion configuration file. In fact to move, our actual robocup agents use motion configuration files. It exist one file for each motion that the agent wants to execute and the AI of this one will have to choose between all its possibilities. These configuration files are xml formatted and describe the motion parameters of each engine. The engines are defined by the representation of the robot's architecture. But the problem is that each robot's configuration an each choice of target point will implies a different mathematical problem to solve. In consequence we will have to set up an IA able to solve them.

The server module is the last one. Its task is to help the user to easily use the simulator server. This interface should allow the user to launch a simulator server, to get the data from the simulation by using a log file system and finally to be able to kill the server. In the aim to be easily improved to a parallel version of the software, the interface has been design to allow managing more than one server in same time. Another little part of the software has been developed in order to easily management he

log file and extract the data. However this last one will not be describe in this paper.

Figure 2 shows is the global algorithm that the system will follow:

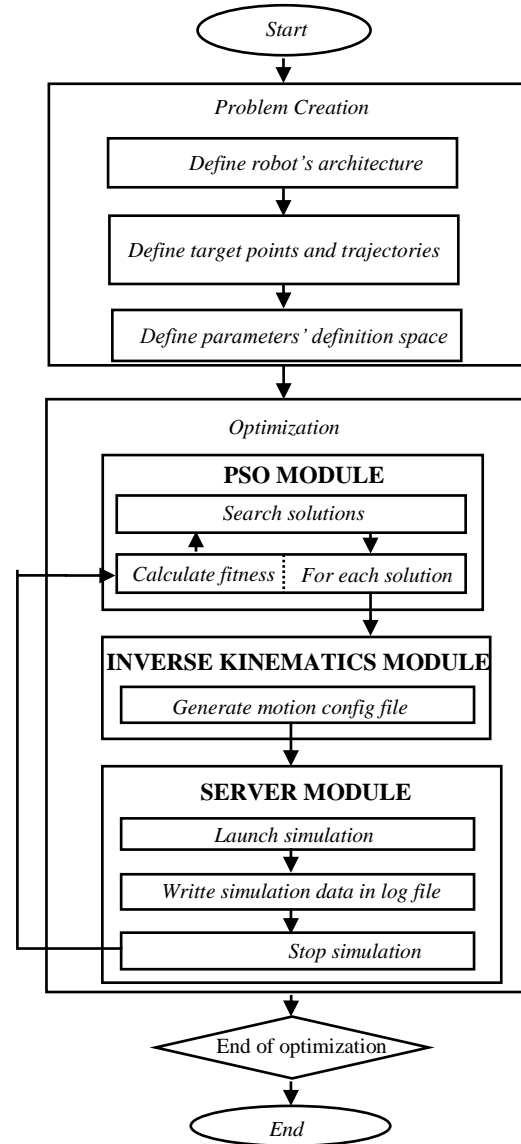


Fig. 2. Global algorithm.

4 Particle Swarm Optimization Module

4.1 Standard algorithm

The particle swarm optimization (PSO) is a research algorithm whose objective is to find the best combination of values for a set of parameters.

The particle swarm optimization is based on the idea of a flock of fishes which move in a space [4]. The aim of each fish is to find the place where there is the best food (this representing the quality of a solution calculated with the fitness function). The fishes always remember the last best place that they have found and they communicate together to share information about the best place found by the whole flock. The fish will move to a new position in function of these two informations. Finally the whole flock should finish by finding the best place of the space.

Algorithm Components. As said just before the PSO is based on a swarm of particles which are inserted in a research space. The space is created by crossing the definition spaces of the set parameters. Each particle knows its position in this space, the quality of this position, and remembers about the best position that the particle has ever found and the best position that the swarm has ever found. In addition it knows their both quality. To finish each particle has a velocity which is used to move in the space. The position and velocity of each particle have to be contained in the research space.

Initialisation. At the beginning we have to initialise each value of the particles. So we generate a random position in the space and a random velocity. The best position of the particle will take the value of its first position. The best position of the swarm will be determined by sharing information between all the particles' best position.

Evolution.

Number of Iterations. After the initialisation, we launch the research. The research will continue until a stop criterion is complete. These criteria can be the time, the number of iterations or an optimum value has been reached (with a certain error).

Update Position and Velocity. At each iteration, we calculate the new position of the particles and the quality of this position. The quality of the particles is measured by a pre-specified objective function. Then we check if the best position of the particle has been improved and share the information with the whole swarm. Finally we update the velocity of each particle.

The new position of a particle is calculated in function of the old position and the velocity (see Fig. 3).

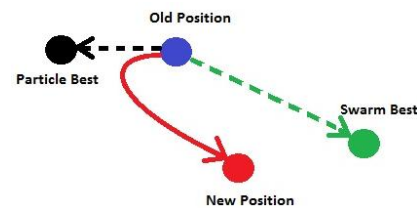


Fig. 3. Update of the particle position.

The velocity is itself calculated in function of the best position known by the particle and the best position known by the swarm [5].

Each time the value of the position and the velocity are updated parameter by parameter.

Fitness Function. The quality of a position is calculated with the fitness function. This function is defined by the user depending on the problem. For example in the case of a walk motion optimization, the quality can be calculated depending on the distance travelled in a given time, the error of trajectory of the robot and its stability.

Confinement. If the new position goes out the research space the particle becomes useless because its position is not valid. That is why the particle swarm optimization algorithm uses a confinement procedure which moves the lost particle on the closest edge of the research space and puts its velocity to 0.

Limitation. But the standard particle swarm optimization algorithm has its own limits. The fact is it is only possible to use it to search a solution for a set of parameters whose limits of definition space are constant.

4.2 Dynamic search space PSO

In our case, the parameters of movement equation are often linked each other. For example during a walk motion the maximum high authorized for the foot will depend on the actual value of the hips high. It means that each particle will have its own search space depending on its position.

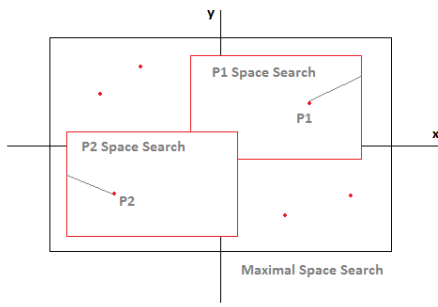


Fig. 4. Search space representation in a two-parameter optimization problem.

Two solutions are possible; the first is to use the precedent algorithm with a space equal to the maximal space search. We define it by putting the maximal value of the parameters in the expressions which determine the limits of the definition space. But use this method implied to effectuate the research in lots of positions which are not valid (they correspond to a totally impossible movement). It is a loss of time. The second one is to use one search space search for each particle. This is the subject of this part.

One Search Space by Particle.

Problem. As said before a research space if defined by crossing the definition spaces of the set of parameters. But in this case some definition space's limits are defined by mathematical expression which depends on the value of some parameters of the set. So the parameters are linked each other.

The problem is that when we want confined the particles in the search space; we change the value of the position parameter by parameter. But each time that we change the value of one parameter, the position of the particle will change. And so on for its search space. If we effectuate the change without any regards for the order of the links between the parameters there is a possibility that the procedure enter in an infinite loop.

To avoid this problem, the best way is to confine the particle following a precise order which correspond to the link order of the parameters.

Tree Representation. To be able to find the good order we can use a tree representation of the links between the parameters. Each parameter has two limits, which depends of a subset of parameters.

In this tree, the nodes will represent the parameters. A parameter is father of all the parameters it depends of. A parameter which depend of nothing is a leaf directly link to the root. The root node doesn't correspond to any parameter. It only exists to be able to create a single tree even with parameters without dependences. Here an example of tree for four parameters A, B, C, D and where to calculate B you need to know the value of C and D.

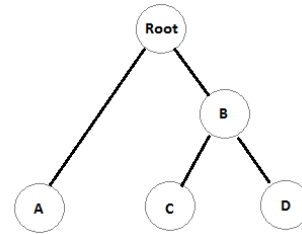


Fig. 5. Tree of parameter dependences.

Modification of the Standard PSO. Finally in the aim to use the PSO algorithm, we need to change the confinement procedure. If the particle is not in its search space, we need to put the particle on the closest edge of the search space. But we have to follow the link order.

The following figure show in red the good path to determine the parameters value after confinement.

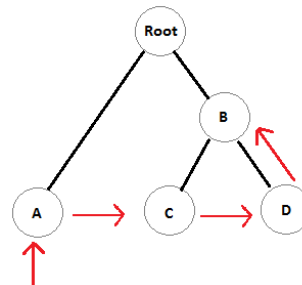


Fig. 6. Confinement parameters order.

We start from the "A" node, and each time that we move on a new node we determine the new limits of the definition space with the parameter value calculate before and put the parameter value of the actual node to the closest limit. And go to the next node until we arrive to the root node.

4.3 Experience

In order to test this module of and to compare the efficiency between the standard and the new algorithm, we have set up an experience. For a same problem composed of a set of variables, which take

their value in a defined interval, and of a fitness point, we will launch a set of simulation for both algorithms. And we will record the number of iterations needed to find the fitness point.

The problem. To For this experience we chose a two dimensional problem. It means that the problem is composed of two variables x and y . We defined a fixed fitness point, the point of coordinate (5.5 ; 0.01). In this problem, during the new algorithm test, the second variable (y) space search limits will depend on the value of the first variable (x). The limits of the variables of the standard algorithm test will be defined as the biggest space possible (biggest value for x enter in the limits equation of y). In the next graph we can visualize the space limits:

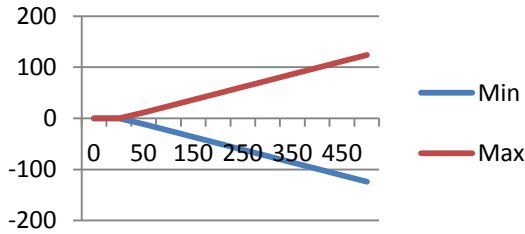


Fig. 7. Example of the space limits (case B).

We will effectuate the experiments in three situations A, B and C. Each situation corresponds to a different variable space search ($0 < x < 1000$), their limits value are presented in the following table :

	New Algorithm		Standard Algorithm	
	Min	Max	Min	Max
A	$-(0.05 * x) + 0.25$	$+(0.05 * x) - 0.25$	-49.75	49.75
B	$-(0.25 * x) + 1.25$	$+(0.25 * x) - 1.25$	-240.5	240.5
C	$-(0.5 * x) + 2.5$	$+(0.5 * x) - 2.5$	-497.5	497.5

Table. 2. List limits of definition for the variable y .

For each experience we will launch a set of simulations with 50 particles and 500 iterations (which is enough to solve the problem).

The results. For each simulation we have check how many iterations the algorithm needed to find the fitness point. Here are the results of the experiences:

	A	B	C
Standard algorithm	279.4	296.5	298
New algorithm	282.3	254.5	250.3

Table. 3. Average number of iterations needed to find the fitness point.

We remark that the standard algorithm become less efficient on a big research space (B and C) and so in these cases we have a gain of time. The gain is not so big but the problem was easy, 2 dimensions and only 50% of impossible positions. But our software will have to deal with more complicated. In addition fitness calculations need a simulation of the robots. This simulation can take 10 sec as 1 or 2 minutes in function of the need. So each iterations gain is a big victory. Finally the improvement of the algorithm still allows to parallelise it [6].

5 Conclusion

To conclude this paper, even this software project still in development some base the design have been choose and some important module have ever been implemented. The aim is to allow to easily generating optimized motions for the robot of y our choice and by the way we can use it to try different kind of architecture.

PSO is a very easy algorithm which is highly parallelizable. The experiences showed the efficiency of the new algorithm. The server module is very important because it will also help to parallelize our software by launching lots of simulation in a same time but also by following the experience of several agents in a same simulation.

References

1. Stone, P.: Layered Learning in Multiagent Systems: A Winning Approach to Robotic Soccer, MIT Press (2000.)
2. Cord Niehaus, Thomas Röfer, Tim Laue: Gait Optimization on a Humanoid Robot using Particle Swarm Optimization
3. Uchitane, T., Hatanaka, T.: Applying Evolution Strategies for Biped Locomotion Learning in RoboCup 3D Soccer Simulation, Proc. of 2011 IEEE Congress on Evolutionary Computation, pp. 179-185. (2011)
4. Kennedy, J., Eberhart, R.: Particle Swarm Optimization, Proc. of IEEE International Conference on Neural Networks, pp.1942-1948. (1995)
5. Clerc, M.: Particle Swarm Optimisation, John Wiley & Sons (2010)
6. J.F.Schutte, J.A.Reinbolt, R.T.Haftka, A.D.George: Parallel global optimization with the particle swarm algorithm, (2004)

自律移動ロボットによる人避け動作のための環境地図構築

Map Construction for Human Avoidance of Autonomous Mobile Robot

秋山英久 進藤真 荒牧重登

Hidehisa AKIYAMA Makoto Shintou Shigeto ARAMAKI

福岡大学

Fukuoka University

akym@fukuoka-u.ac.jp

概要

In this paper, we propose a method to construct an environment map by autonomous mobile robot to solve the human avoidance task. The method is based on Digital Elevation Model and we introduce the decremental model in order to eliminate ghost obstacles generated by moving bodies. We implemented our method on the mobile robot Roomba with Laser Range Finder and Kinect compatible device as sensor devices. We conducted two experiments to determine the suitable parameters of our method.

1 はじめに

近年、ロボットは人間の生活に密接に関わるようになってきた。今後、ロボットが人間と同じ空間で作業する場面は更に増えると予想される。移動ロボットが人間と同じ空間で作業を行う場合、ロボット自身が人間や動物などの自律移動する存在を認識し、それらを回避する動作が必要となる。このような動作を実現するには、ロボットが動作する環境の地図をロボット自身が構築し、動的に更新し続けなければならない。

本稿では、人間の生活空間における移動ロボットの人避けタスクを設定し、人避けに必要な環境地図の表現方法およびその動的な更新手法を提案する。実験では、提案手法を組み込んだ移動ロボットを実際に動作させることで、人回避タスクにおける地図構築のための適切なパラメータを調査する。

2 関連研究

人間の生活空間において移動ロボットが人を追尾・回避するには、周囲の情報を常に収集し、ロボット内部で環境地図を更新し続けなければならない。周囲の情報を収集するにはさまざまなセンサをロボットに搭載する必要がある。移動ロボットが環境地図を構築するためのセンサとして、レーザー式測域センサを用いた研究例が多い[2, 3, 4, 5]。これらの研究では、センサによって収集した点群情報を用いて、周囲の物体の形状再現や自己位置測定の精度改善を試みている。また、ロボットのセンサデバイスとしては、Microsoft社のKinectを用いた研究も盛んである[1, 6]。Kinectを用いることで、深度情報の計測、人骨格の認識機能、動体検知機能を安価に導入することが可能になった。センサデバイスは小型化、低価格化が進んでおり、さまざまなセンサを搭載した家庭用ロボットの登場が予想される。

本研究では、複数のセンサを搭載した移動ロボッ

トを用いて、生活空間においてロボット自身に人を認識させ、人と競合しない自律動作を実現することを目指している。本稿では移動ロボットとしてアイロボット社のルンバを用いる。ルンバ上にレーザー式測域センサと Kinect 互換の動体検知デバイスを搭載し、自律的に環境地図構築および人避けを行うロボットシステムを開発する。

3 人避けタスク

3.1 人避けタスクとは

本稿では、移動ロボットはある目的地点への到達を最終目標とする。ロボットが移動する環境には机、椅子、壁などの障害物が存在する。更に、その空間内で人が歩いて移動し、ロボットの進路上に侵入することを想定する。移動ロボットは周囲の障害物に加えて進路上に侵入する人を回避しなければならない。移動ロボットが人を避けるには、ロボットの近くに人が存在することを認識し、目的地までの経路計画においてその場所を避けた経路を生成する必要がある。さらに、人が移動して移動ロボットの進路上に侵入した場合、ロボットは新たな経路を生成し直す必要がある。

3.2 環境地図構築の必要性

本研究では、人の生活空間でロボットが自律的に動作をすることを目的としている。ロボットを自律的に動作させるには、ロボット自身の信念となる内部モデルを持たせる必要がある。本稿ではこの内部モデルを環境地図と呼ぶ。しかし、人の生活空間は時々刻々と変化してしまうため、移動ロボットが事前に環境地図を与えられていたとしても、ロボット自身のセンサ情報に基づいて地図情報を修正、更新し続ける必要がある。

本研究では複数のセンサを使用し、移動ロボット自身に環境地図を動的に作成、更新させる。動的な地図構築を実現する上で問題となるのは、センサ情

報を統合する手法および不要なデータの削除方法である。

4 提案手法

4.1 数値標高モデルによるセンサデータ統合

本研究では移動ロボット自身がセンサデータに基づいて環境地図を構築する。センサデバイスとしてはレーザー式測域センサと Kinect 互換の動体検知デバイスを想定している。しかし、これらのセンサから得られたデータをそのまま地図情報として格納しようとすると、データ量が膨大になってしまう。そこで、本研究では得られたデータを数値標高モデルを用いて圧縮する。

数値標高モデルとは、地形データをデジタル表現することを目的としたデータ表現方法である。本研究で使用するレーザー式測域センサを例にあげて数値標高モデルの特徴を説明をする。

レーザー式測域センサは、センサから障害物までの距離とその方向をデータとして取得できる(図1)。本研究で使用するレーザー式測域センサは、センサ前方を中心として-120度から120度までの範囲で平面上の物体までの距離を測定できる。得られたデータは3次元空間上の点群として表現され、点群データはロボットの座標系へ変換して使用される。センサは、0.1秒ごとに0.36度刻みで240度の範囲を走査する動作を続ける。そのため、センサが取得する点群の数は膨大なものとなり、それらすべてを格納、処理することは困難となる(図2)。さらに、レーザー式測域センサから得られる点群データに加えて、Kinect 互換デバイスによる動体検知結果を環境地図に統合しなければならない。

本稿では、得られたセンサデータを数値標高モデルによって圧縮する。数値標高モデルでは対象となる空間を平面上のグリッドで表現する。本稿では、図3に示すように対象領域を正方形のグリッドで分割する。センサから得られた情報は各グリッドに対応

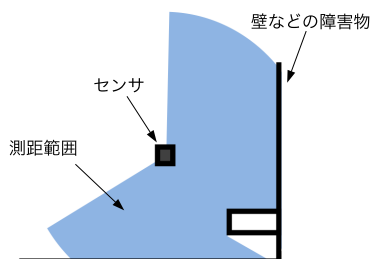


図 1: レーザー式測域センサの測距範囲。本稿で用いるセンサでは、センサを中心として 240 度の範囲の平面上で物体までの距離を測定できる。

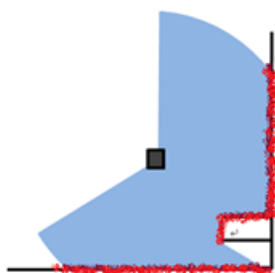


図 2: レーザー式測域センサによって得られる点群データ。図中の赤色は得られた点データを表す。1 回の走査で最大 660 個程度の点データを得られる。

付けられて処理される。

本稿で用いる数値標高モデルによる地図は、ロボット座標系で固定する。座標系の原点はロボットが起動した位置、X 軸正方向はロボット起動時のロボット正面方向とする。得られたセンサデータの位置情報は、観測された時点のロボットの自己位置測定情報を用いてロボット座標系へと変換される。そして、その位置座標に対応するグリッドの数値情報が更新される。さらに、ロボットの自己位置測定の誤差とセンサデータの誤差を考慮して、障害物が検出された周囲のグリッドにも障害物を表す数値情報を書き込む (図 4)。

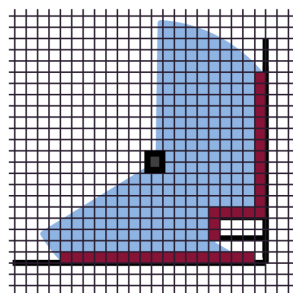


図 3: 対象領域をグリッドで分割した状態。数値標高モデルでは各グリッドに数値情報を格納することで情報量を圧縮する。レーザー式測域センサで得られた点群データは対応するグリッドごとの数値情報へと変換される。

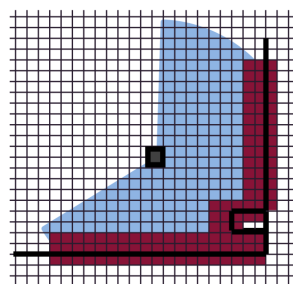


図 4: 誤差を考慮してセンサデータに対応するグリッドの周囲にも障害物を表す数値情報を書き込む。

本稿では、レーザー式測域センサで得られた点群データ、および、Kinect 互換デバイスで検出された動体の位置、の二種類のデータを数値標高モデルの地図上で統合して管理する。ロボットの制御では、統合地図を参照して経路計画を行う。

4.2 センサデータの座標変換

センサから得られるデータに含まれる位置情報は、センサを基準とした座標系で得られる。この位置情報は移動ロボットの座標系へ変換される必要がある。センサ座標系でのセンサデータの座標値を (x'_s, y'_s) 、ロボット座標系でのロボットの位置座標を (x_r, y_r) 、

ロボット座標系でのロボットの向き（センサの向き）を θ_r とすると、ロボット座標系でのセンサデータの座標値 (x_s, y_s) は以下の座標変換で得られる:

$$\begin{pmatrix} x_s \\ y_s \end{pmatrix} = \begin{pmatrix} \cos\theta_r & -\sin\theta_r & x_r \\ \sin\theta_r & \cos\theta_r & y_r \end{pmatrix} \begin{pmatrix} x'_s \\ y'_s \\ 1 \end{pmatrix}$$

4.3 センサデータの更新

4.3.1 動体のデータ更新方法

本稿で開発するシステムでは、Kinect 互換デバイスで検出された動体の情報を線形リストで管理する。このリストは検出された動体ごとに新規生成され、その動体の観測情報の履歴を保持する。既知の動体が再検出されるとその位置座標と観測時間の組がリストに追加される (図 5)。検出された物体が既知の物体と同一かどうかの判定は OpenNI¹ が提供している API によって行われる。

リストの長さには最大長を設定しておき、リストが一定以上の長さになった場合は観測情報が古い要素から削除される。また、観測から一定時間以上経過した要素も自動的に削除される。リストが空になればリストそのものが削除される。このように観測情報の履歴を保持しておくことで、ロボット上のセンサが動体を一時的に見失ったとしても、環境地図へ動体の情報を反映することができる。

4.3.2 レーザー式測域センサのデータ更新方法

レーザー式測域センサで得られた点群データに基づいて、各グリッドに対応する点の数を数値情報としてグリッドに格納する。動体がセンサの測距範囲に存在した場合、その物体に対する点データが観測される。しかし、その物体が移動すると、観測された点データは存在しない障害物の情報として残されることになる。本稿ではグリッドに格納された数値情報を減衰させるモデルを導入することでこの問題を解決する。

¹<http://www.openni.org/>

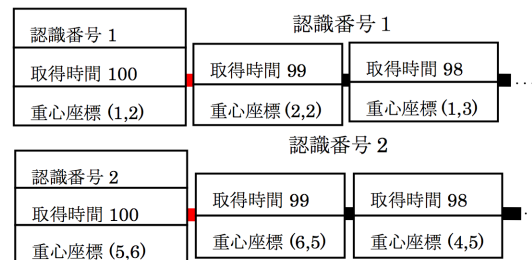


図 5: 線形リストによる動体データの履歴管理。検出された動体ごとにリストを作成し、観測情報の履歴を保持する。

図 6 に減衰モデルの動作例を示す。提案する減衰モデルでは、一定時間ごとにすべてのグリッドが持つ数値を減衰させる。これによって、過去に観測した動体が移動していればその情報は時間が経過するにつれて信頼性が低下し、最終的に地図上から障害物としての情報そのものが消える。ここで、一定時間の減衰量を決定する減衰率のパラメータを導入する。新規に点群データが観測された場合は、過去の数値情報に点の数を足しあわせ、グリッド毎に累積値として保持する。ただし、グリッドが保持する数値情報には最大値を設定する。単位面積当たりで保持する点の数の最大値を設定し、この値に基づいて各グリッドが格納する点の数の最大値が決定される。

5 実験

4 節で述べた地図構築手法を実ロボット上で実装し、以下の二つの調査を目的とした実験を行う:

1. 数値標高モデルで用いるグリッドの適切な大きさ
2. 減衰モデルにおける適切な減衰率

5.1 実験設定

いずれの実験も移動ロボットとしてルンバ研究開発キット²を用いる。ルンバ上に台となるフレームを

²<http://science.irobot-jp.com/>

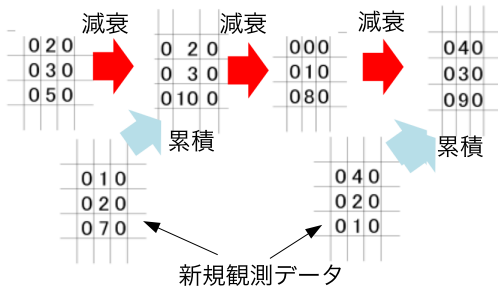


図 6: 点群データによるグリッド情報の更新. 一定時間経過すると各グリッドが持つ数値は減衰される. 新規観測データが得られた場合は, グリッド毎に新規数値を足しあわせる. この例ではグリッドの最大値を 10, 減衰率を 10%としている.

取り付け, 制御用 PC とセンサを設置する. 今回の実験ではデータ収集用 PC と制御用 PC の 2 台の PC を搭載する. データ収集用 PC にはレーザー式測域センサとして北陽電気社製の URG-04LX-UG01 を, Kinect 互換デバイスとして Xtion PRO LIVE を接続する. 制御用 PC とルンバはコミュニケーションケーブルで接続される. データ収集用 PC と制御用 PC との通信は, 実験環境内に設置された無線アクセスポイント経由で行う. 最終的な移動ロボットの外観を図 7 に示す.

移動ロボットはあらかじめ設定された目標位置へ到達するための経路計画を実行し, 生成された経路に従って移動動作を繰り返す. 経路が一度決定されれば, その経路上に障害物が新しく現れない限りその経路に沿った移動を続行する. 経路上に障害物が発見されれば, 経路計画を再度実行する. 経路計画の空間として数値標高モデルのグリッド分割をそのまま用い, 経路計画は A* アルゴリズムで行う. 今回は初期位置から 4m 離れた位置を目標位置とし, ロボットの移動途中で人が経路上に侵入する.

データ更新における各パラメータの設定は次のようにした. 動体の観測情報の履歴を保持するリストの最大長は 10, 動体の観測情報の最大保持時間は 10 秒とする. グリッドが保持する点の数の最大値を決

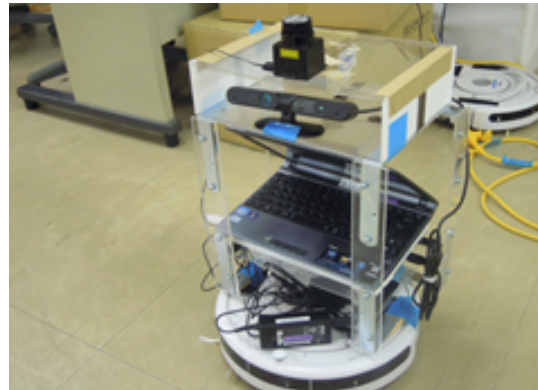


図 7: 移動ロボットの外観. ルンバに台を取り付け, PC2 台とセンサを設置する.

表 1: 実験 1 の結果.

	5cm	10cm	15cm	20cm
1 回目	不可	55 秒	52 秒	55 秒
2 回目	不可	不可	41 秒	48 秒
3 回目	不可	67 秒	44 秒	50 秒

定するパラメータとして, 10cm^2 あたり 10 個を最大値とする. 動体の情報と点群データのいずれも, データ更新の時間間隔は 1 秒とする.

5.2 実験 1: グリッドの適切な大きさの調査

グリッドの辺の長さを 5cm, 10cm, 15cm, 20cm に変更し, 目標位置までの到達時間をそれぞれ計測した. 実験結果を表 1 に示す. 表から分かるように, グリッドの辺の長さが 15cm の場合にもっとも安定した結果が得られた. 図 8 に生成された環境地図と経路の例を示す.

5.3 実験 2: 適切な減衰率の調査

グリッドが保持する点の数を減衰させる割合を 0%, 30%, 50%, 100% に変更し, 目標位置までの到達時

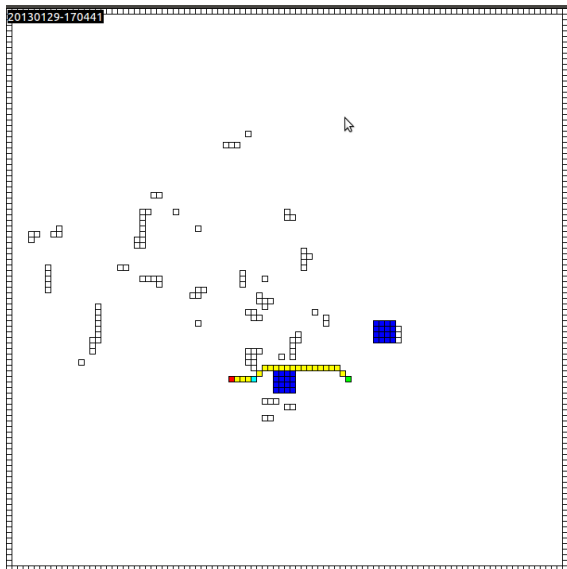


図 8: グリッドの辺の長さが 15cm の場合に生成された環境地図と経路の例. 赤色はロボットの現在位置, 青色は動体の位置, 黒枠に灰色は障害物, 黄色は生成された経路を表す.

表 2: 実験 2 の結果.

	0%	30%	50%	100%
1 回目	65 秒	52 秒	67 秒	60 秒
2 回目	不可	41 秒	57 秒	60 秒
3 回目	不可	43 秒	55 秒	65 秒

間をそれぞれ計測した. グリッドの辺の長さは 15cm とし, その他の設定は実験 1 と同じとする. 各グリッドが保持できる点の数の最大値は 23 となる. 実験結果を表 2 に示す. この結果から, 減衰させる割合は 30%程度が妥当であると考えられる.

6 まとめ

本稿では移動ロボットによる人避けタスクを設定し, これを解決するために数値標高モデルをベースとした環境地図構築手法を提案した. 実験では, 数

値標高モデルにおける適切なグリッドの大きさの調査, および, 導入した減衰モデルにおける適切な減衰率の調査を行った. 今後の課題として, ロボット本体の制御プログラムの安定性向上, より多様なセンサデータへの対応, 動体の移動予測, などが挙げられる.

参考文献

- [1] Piyush Khandelwal, Peter Stone: A Low Cost Ground Truth Detection System for RoboCup Using the Kinect, Proc. of the RoboCup International Symposium 2011 (2011).
- [2] Genya Ishigami, Keiji Nagatani, Kazuya Yoshida: Path Planning for Planetary Exploration Rovers and Its Evaluation based on Wheel Slip Dynamics, 2007 IEEE International Conference on Robotics and Automation (ICRA2007), pp. 2361–2366 (2007).
- [3] 古川奈々恵, 五島美沙, 三田彰: レーザーレンジファインダ搭載ロボットを用いた日常生活における動線把握, 社団法人日本建築学会 学術講演梗概集. A-2, 防火, 海洋, 情報システム技術 2011, pp. 537–538 (2011)
- [4] 大野和則, 永谷圭司, 秋山英久: レスキューロボットの地図構築, 日本ロボット学会誌, Vol.28, No.2, pp. 169–172 (2010).
- [5] 奥迫伸一, 坂根茂幸: レーザレンジファインダを用いた移動ロボットによる人の追跡, 日本ロボット学会誌, Vol.24, No.5, pp. 605–613 (2006).
- [6] 吉田聡, 泉正夫, 辻洋: 人物判定における Kinect の骨格線情報の利用, 映像情報メディア学会技術報告, Vol.36, No.8, pp. 123–126 (2012).

LLSF におけるロボットの初期方向同定に関する一考察

The estimation of robot's location at Logistics League Sponsored by Festo.

植村 渉

Wataru UEMURA

龍谷大学

Ryukoku University

wataru@rins.ryukoku.ac.jp

Abstract

RoboCup のリーグの一つである LLSF (Logistics League Sponsored by Festo) では、今年からフィールドにロボットを投入した時の自己位置同定の技術向上を求めるルールになった。試合開始時は 5 分間の人の手によるセットアップを認めているが、試合中にロボットを再投入するときやテクニカルチャレンジにおいては、15 秒以内のクイックセットアップを必要とする¹。このリーグで使うロボットには 9 つの距離センサが円形に均等に配置されているので、これらを有効利用することが期待されている。そこで本研究では、LLSF におけるロボット投入時に距離センサを利用して簡単に自己位置、特に角度を推定する方法を検討する。ロボット投入エリアには、後方に壁があるため、後方の 4 つの距離センサを使うのが効果的である。距離センサは、距離に応じた電圧値を出力し、距離が 4cm 以上離れているときは単調減少であるため、一意に距離を決めることができるが、0cm から 4cm の間は単調増加しており、全範囲では 4cm において上に凸な関数となり、一意に距離を決めることができない。それゆえ、壁に近い時は、4 つのセンサの値の組み合わせが複雑になり、距離への換算が困難である。本研究では、センサ値と距離の換算を行うため、識別機である SVM やニューラルネットワークの導入を検討する。探索空間を減らすため、2 つのセンサの値と角度の関係性を調べ、導入に必要な事項を検討する。

¹ なお、人の手によるセットアップが 15 秒以内に求められているだけであり、その後ロボットが自律的に自己位置を同定するのは制限されていない。

1 はじめに

2012 年から正式リーグとなった LLSF (Logistics League Sponsored by Festo) では、自律移動する無人搬送車 (Automated Guided Vehicles) の実現を目指し、競技を行っている。特に、今年には試合の進行や記録を行う RefBox (Referee Box) を設置し、競技内容の自動化が進んでいる。また、試合には Festo 社製のロボットである Robotino[®] を 3 台用いる。このロボットには、3 軸のオムニホイールが搭載されており、それらの回転角とジャイロを用いたオドメトリが実装されている。昨年の決勝戦に進んだチームの動きを見ている限りでは、試合開始時の 5 分間の人手によるセットアップのみで位置と角度の調整を終えており、その後の試合においてはロボットが位置調整を行っておらず、オドメトリによる位置推定の精度は高いものと考えられる。ホイールの空回り等によって位置情報に誤差が生じるが、優勝チームの動きはなめらかであり、そもそも空回りが生じていない。オドメトリの精度にはそれらの効果も含まれている。

試合開始時のロボットの設置には、5 分間の人の手による調整が認められている。しかし、試合中のフィールドへのロボットの再投入やテクニカルチャレンジにおけるロボットの投入においては、15 秒以内のクイックセットアップを必要とする。Robotino[®] において距離や位置を推定する道具として、円形に均等に配置されている 9 つの距離センサ (図 1) や、地面のライン認識に用いることができるオプティカルセンサ、そしてカメラなどが装着されている。ルール上は、規定範囲内の大きさであれば他のセンサ類の装着も可能であるが、ここでは元々装着しているセンサを対象として検討する。ロボット投入エリアでは、後方に壁があるため、後方の 4 つの距離センサを使うのが効果的である。そこで本研究では、LLSF におけるロボット投入時に距離センサを利用して簡単に自己位置、特に角度を推定する方法を検討する。

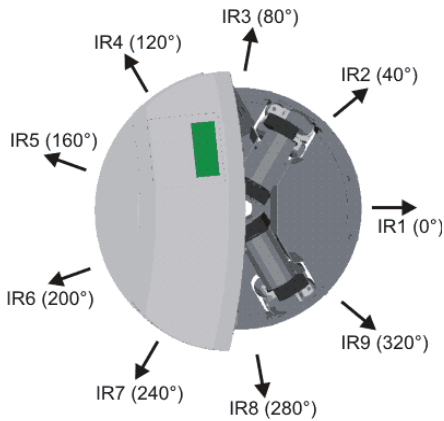


Figure 1: Robotino®に搭載されている9つの赤外線式距離センサ (Robotino View2 の Help より)

ロボットに搭載している距離センサは、シャープ社製の赤外線ユニット GP2D120 であり、4cm から 40cm までの距離を電圧値で返す。距離と電圧の関係を図 2 に示す。壁などの障害物との距離が 0cm から 4cm になると、一意に距離を決めることができない。それゆえ、壁に近い時は、4つのセンサの値が複雑になり、距離への換算が困難である。

そこで、センサ値から距離への換算を行うための関数が必要となる。ここではロボットの向きを調整したいので、どちらに回転すべきかがわかれば良い。事前に各センサの値とその時の Robotino の向きを調べることが可能であるため、教師あり学習の適用が考えられる。高次元の識別関数が必要となるので、SVM (Support Vector Machine) の導入が考えられるが、ここでは Robotino 上でのプログラムを考えてニューラルネットワークの適用を提案する。

Robotino 上でのプログラム方法にはいくつかの方法が提供されているが、オリジナルの言語としてブロックを組み合わせるプログラミング法である RoboView 2 が提供されている。RoboView2 では、1step ごとに、各ブロックの値を計算し更新する。ニューラルネットワークは、ニューロンとシナプスをモデル化して、それぞれの結合係数と閾値を随時更新するアルゴリズムであるが、ステップ毎に値を更新するという点で RoboView2 との相性が良い。

本研究では、LLSF におけるロボット投入時の位置と向き調整を対象とし、センサ値と距離の換算を行うためニューラルネットワークの導入を提案する。ニューラルネットワーク構築に必要な中間層の数を検討するため、センサの値と角度の関係を調べる。そのままでは探索空間が大きいため、2つのセンサを組として、導入に必要な事項を検討する。最後に、今後の課題を挙げて、本研究をまとめると。

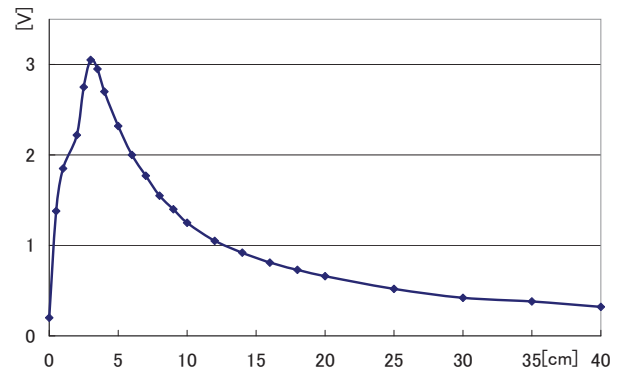


Figure 2: 赤外線式距離センサ (GP2D120) の特性 (仕様書より)

2 ニューラルネットワークの導入の準備

いくつかの入力に対して、出力すべき値の答えがあらかじめわかっているとき、教師あり学習法を用いることで、適切な関数を構築することができる。ここでは、脳細胞の接続を模したニューラルネットワークを導入する[2]。

ニューラルネットワークでは、ニューロンとシナプスをモデル化しており、ノードへの入力の合計値が閾値を超えたとき出力を行うシステムであり、各ノード間の重みを調整することで、適切な入出力の組を扱うことができる。

入力からのリンク数が同じノードを層としてグループ化し、入力層 - 中間層 - 出力層として扱う。学習結果を入力層側に反映させるためには、信号を出力側から入力側へと送る必要がある。これをバックプロパゲーションと呼び、誤差情報を用いて学習するため収束が速い。

中間層の数とその層が持つノードの数によってそのネットワークが学習できる能力が決まるが、問題環境が未知な状態でそれを決めることは難しい。ここでは、入力層と同数のノードを持つ1層の中間層を設計した。図3に RoboView2 の SubProgram の全体図を示す。

このネットワークに IR4 から IR7 の距離センサの値を入力として与え、壁に対する向きを教師信号として与え、距離と角度の関係を学習させた。残念ながらネットワークは発振し、適切に学習できなかった。そこで次節ではその原因を明らかにし、対策を考える。

3 入力値のばらつき

今回のニューラルネットワークには、4つの距離センサの値を入力とし、回転すべき向きを学習させたが、それらの入力情報から出力結果が識別可能かどうかを検証する。

入力として4つのセンサ値を同時に扱うと視覚化が困難であるため、まずは内側の2つのセンサの組と、外側の2つのセンサの組にわけ、2次元平面上に描画した。図4が内側の2つのセンサの値と、角度の関係である。なお、Robotino の回路の関係で、これらのセンサは最大値

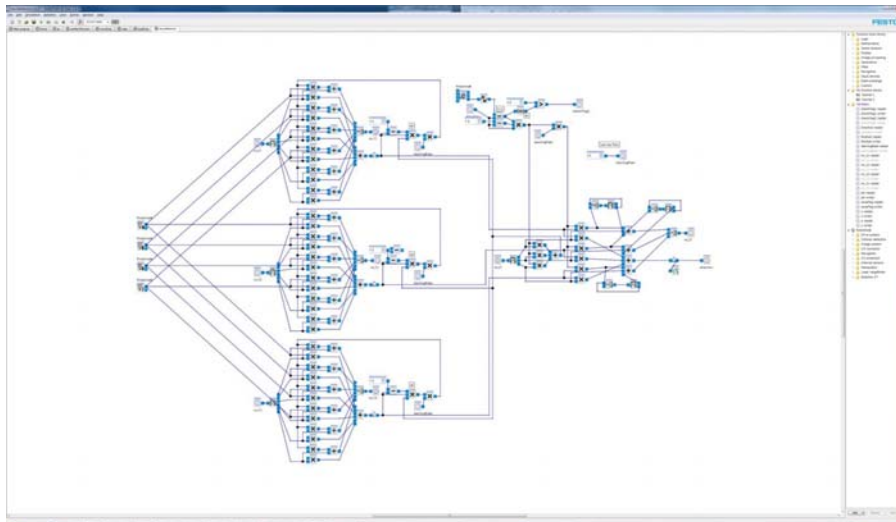


Figure 3: RoboView2 で作成したニューラルネットワーク学習器

約 2.54 である。グラフの右上側に張り付いている点は、センサの値が飽和している場合と考えられる。今回、これらは除外する。

センサは左右対称に設置しているので、 $y = x$ で対称なグラフになるはずであるが、IR5 の方が少し値が小さいすなわち遠くなっている。これは、センサの形が図 5 のように左右対称でなく、赤外線送信部と受光部が左右に並んで配置されているためと考えられる。対象物が離れているときは、ほとんど影響を受けないが、今回は対象物と数 cm の距離のため、送信部と受光部の距離の影響が表れたと考えられる。

また、4cm 以下ではセンサの値が折り返すため、正の角度の一部のデータが、負の角度のデータの領域にプロットされている。これらがニューラルネットワークの学習に対して影響を与えていたと考えられる。

次に、外側の 2 つのセンサの値と角度の関係を図 6 に示す。外側のセンサの場合、両方が同時に 4cm 以下にはならないので、内側の結果と比べてグループがわかりやすく分散している。いずれも傾き 1 ($y = x$ の直線) の直線と同じ角度で帯状に延びている。ここでも、赤外線センサの送信部と受信部の距離の差の影響が確認でき、 $y = x$ の帯は、0 度ではなく -6 度から -3 度の角度の時の点の集合である。

いずれにしても、それぞれの角度の結果は、帯状になっており、一部隣接の帯と重なっている。それゆえ、これらのセンサ値から角度を識別するには、識別関数を用いるよりも、ファジィ制御などを用いた方が適していると考えられる。

4 おわりに

LLSF におけるロボットの向き推定について検討した。ロボットの初期位置では後方の壁があるため、ロボットに搭



Figure 5: 距離センサ GR2D120 の外観。左側が受光部、右側が送信部である。

載されている 9 つの距離センサのうち、後方の 4 つを使って現在の向きの推定を行おうとしたが、分割平面が高次元になるため 4 ノードを持つ 1 つの中間層のニューラルネットワークでは発振を起こした。センサの値を 2 組ずつ 2 次元平面に表し、識別の難しさを確認した。

謝辞

本研究は、龍谷大学理工学学術研究助成基金による研究助成を受けて行われたものである。

参考文献

- [1] RoboCup Logistics League, <http://www.robocup-logistics.org/>
- [2] Lau, C. and IEEE Neural Networks Council, "Neural networks: theoretical foundations and analysis," IEEE Press Selected Reprint Series, IEEE Press, 1992.

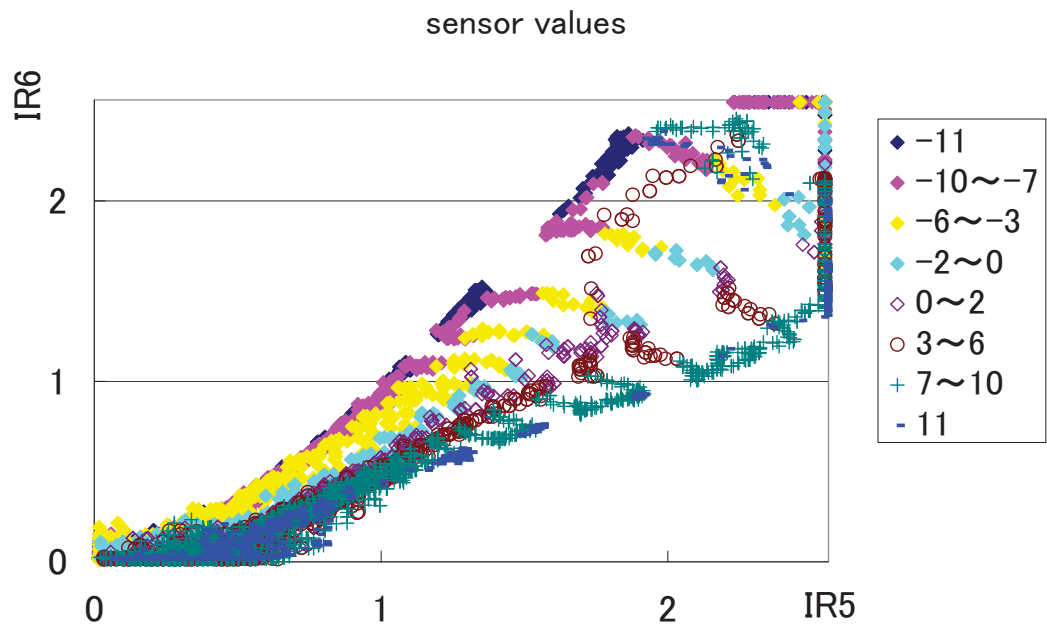


Figure 4: 内側 2 個の距離センサと角度の関係

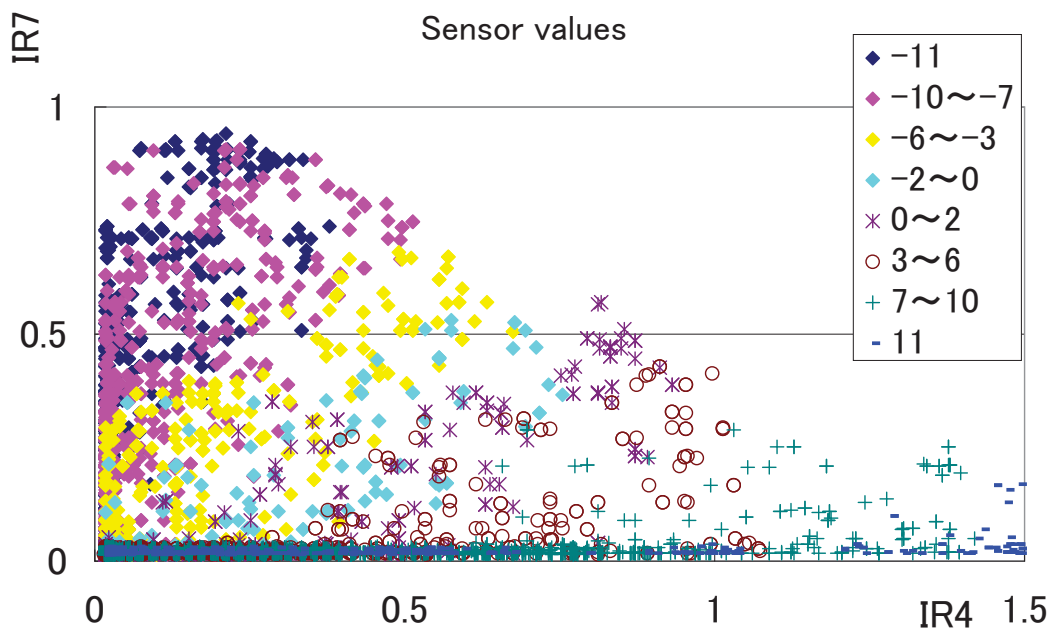


Figure 6: 外側 2 個の距離センサと角度の関係

Autonomous Humanoid Robot Playing Scenario for Autistic Children

Yu-Cheng Cheng
Dept. of Electric Eng. and
Advanced Institute of
Manufacturing with High-tech Innovations
(AIM-HI),
National Chung Cheng University
Chia-Yi, Taiwan
Email: lkkkkkkkj@gmail.com

Min-Liang Wang
IRCAD
Asian Institute of TeleSurgery,
Lukang, Taiwan,
pml.wang@gmail.com

N. Michael Mayer
Dept. of Electric Eng. and
Advanced Institute of
Manufacturing with High-tech Innovations
(AIM-HI),
National Chung Cheng University
Chia-Yi, Taiwan
Email: nmmayer@gmail.com

Abstract—Using robot playmate to help autistic children is a relatively new idea for therapy. A Robot can repeat to do the same procedure precisely and does not get tired or bored. Moreover, earlier experiments show that Autistic children show an open behavior to robots. In this paper, we want to discuss the possibility of using humanoid robot to play scenario for autistic children.

I. INTRODUCTION

Autism is a disorder of neural development characterized by impaired social interaction and communication, and by restricted and repetitive behaviors. The diagnostic criteria require that symptoms become apparent after 18 months, or before a child is three years old [11][13]. Many types of therapies are used for treating autism [4], medical medicine therapy, behavior psychology therapy, social work therapy, artist therapy, art therapy, game therapy, music therapy and others more.

The idea to use a robot for therapy has been raised for more than ten years. Kerstin Dautenhahn et al. started to use robots to help autistic children in the year 1998 [5]. This project, "The AuRoRA Project" [1], has been executed for over 10 years. One of their former experiments was using a wheeled robot, Labo-1 [6]; Labo-1's length is 40 cm, width is 30 cm and weight is 6.5 kg, Labo-1 has 8 infrared sensors (4 is in front and 4 is at back). This robot also has a temperature sensor and a speaker. Labo-1 can do behaviors such as obstacle avoidance and child tracking using the devices mentioned above. They put Labo-1 together with an autistic child in a small room which size is 2x3 square meters. Labo-1 tries to catch the child, and it makes a sound when Labo-1 is near the child. Then, they recorded the interaction between the child and Labo-1 as a video. They analyzed the record and got feedback such as whether Labo-1 attracted the child, how much time the child played with Labo-1.

After Labo-1, The ongoing development of small size robots results in better and more robust robots with the advance of the corresponding technologies. There are more and more organizations devoted to this field of research. Robota robot [16] [15] [14], a humanoid doll robot. It has an infrared sensor and some rotatable joints; Tsai, Cheng-Hung [18] used humanoid robot to play pose simulated game. They use a normal webcam and some parts (red gloves and stickers on

shoulder) that the child wore to capture the motion. Then, they use ZigBee to transmit data and compared the motion differences between each other; KASPAR [17] [9], a humanoid robot that has much more movable joints (over 11). Its eyeballs are rotatable and it has tactile sensors; NAO, a robot produced by Aldebaran, is used to teach to autistic children play the drum for rhythmic music playing interaction [8].

Francois Michaud and Catherine Theberge-Turmel [12] have concluded that a robot may be less intimidating and more predictable than a human. A robot can follow a deterministic play routine and also adapt over time and change the ways it responds to the world, generating more sophisticated interactions and unpredictable situations that can help capture and retain the child's interest. We agree with them.

The overall starting point of our research was to build a robot that can help autistic children's development. Thus, we chose the humanoid robot scenario-playing way to achieve it. The scenario-playing method is based on puppetry play.

Puppetry play has previously been used to train children to practice spontaneous, imaginary, and symbolic play [7]. It has been stated the this method can help bring them think concretely into the world of abstract concepts and ideas .

The method's main idea is to let robot play scenario with puppetry, but with humanoid robot's size we don't prepare the real puppets. Instead, we stick two piece paper of characters on robot's hands, act as real puppets.

We intent to check how much is the child willing or capable to understand what is happening. We tell the child a little fairy tale, where the story has a little twist. In the current version the robot takes a role in the fairy tale and acts as a rabbit. Suddenly, the robot switches the role and acts as a turtle. We hope to detect a possible surprise in the reaction of the child. We assume that such a surprise requires a basic understanding of the fairy tale.

II. METHOD

In our case, we use the fairy tale - "Turtle beats Hare" as our scenario. "Turtle beats Hare" is a tale talking about a turtle and a rabbit are having a racing competition, rabbit is lead at first but it lose to turtle in final because of its enemy underestimation and taking a nap in the competition.

Recent research suggests[10] that scenario-playing of a social story can improve social skills of autistic children. A social story means here a series of conversations about how people interact with each other.

Typical stories are "Buy food", "Take a train" or "Go to School". Children with autism have always proved to be a group without imagination which may mean that it does not matter whatever story or tale is. However in the current case, we chose the fairy tale without a human character. The reason is that we want to eliminate the factors of social stories and investigate if their pure curiosity and attention to humanoid robot can serve our purpose.

Our final goal is to design a robot and behavior that inspires and guides them. We attempt to utilize a humanoid robot's characteristics, but also keep the design less intimidating and more predictable to attract autistic child joint attention. Also we intent to foster autistic children in the way that we stimulate their imagination, spontaneous and symbolic play through this robot.

The technical realization is depicted in Fig.1. The child stands or sits in front of the robot, and then watches the scenario with a simple background picture on the wall (realized with a video projector) and its puppets, two stickers (see Fig.2) on his hands. After the scenario ends, the autistic child's reaction is evaluated and it is decided whether the scenario is repeated or not. If it is not repeated, a small test is conducted in which we like to check the understanding. This little test may be done by a test paper or an interactive games. During the procedure, we will also put a web cam in field to record children's reaction for analyzing their attention. The complete result will be summed up by test and related factors.

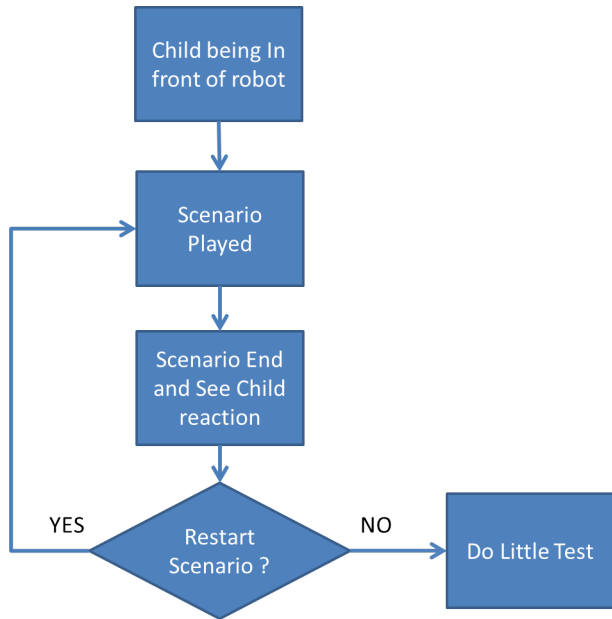


Fig. 1. Flow chart of the humanoid robot scenario-playing method.



Fig. 2. Stickers used on humanoid robot's hands.

III. IMPLEMENTATION

A. Hardware and environment

We build the overall experiment environment with the following hardware: Humanoid robot, laptop, projector, speakers, and wireless access point. The hardware's relation is shown in Fig.5.

Humanoid robot : Here we chose Darwin-OP as our humanoid robot and also our mediator with the autistic children. Darwin-OP is made by ROBOTIS, it has a camera, two microphones, a speaker, a gyroscope sensor, 3 buttons and 20 DOF. Motions are designed in sequences: The robot plays a series of actions of a role and is controlled by the commands received from console (laptop). It sends back the finished signal when the actions were done.

Laptop : A GUI serves as console of the overall environment. It plays the background of scenario with projector and speakers, it also sends commands to Darwin-OP itself for role playing.

Projector and Speakers : An assistant tool to help console (laptop) shows the screen and plays the sound to children.

Wireless Access Point : A communication bridge between console (laptop) and Darwin-OP.

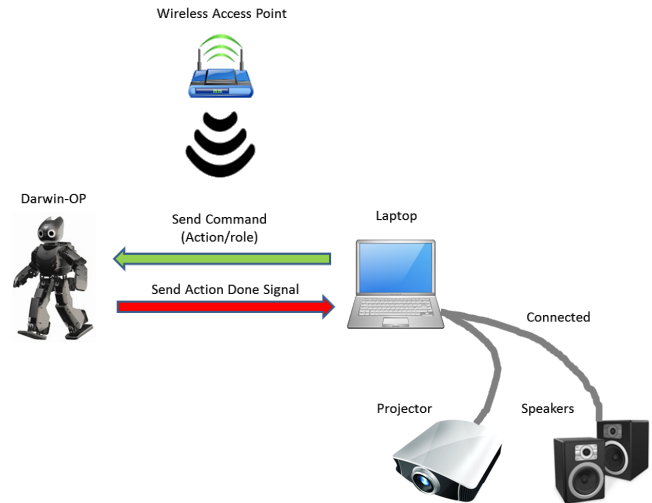


Fig. 5. Environment overall architecture

These hardware's are set down like this in Fig.6, and the dash-line is projected range of projector. Autistic child will be inside the projected range seeing robot play scenario; Assistant will control the console from laptop at side.

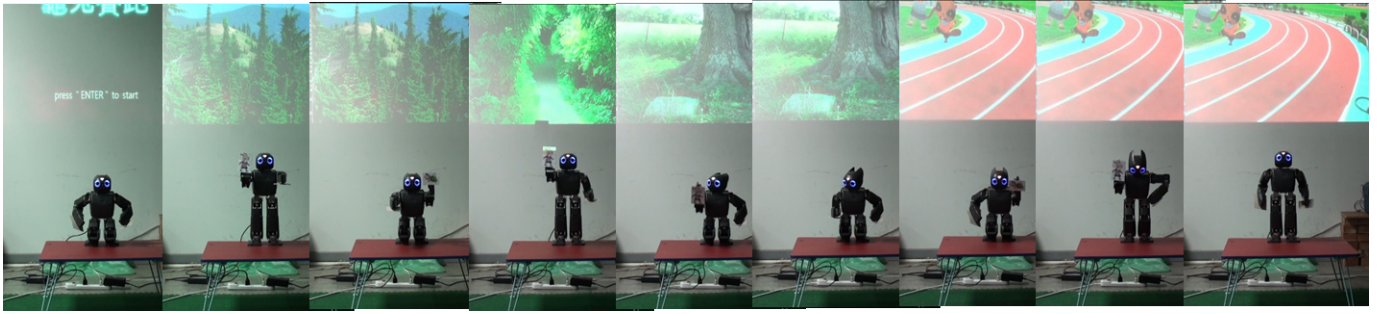


Fig. 3. Projector displayed the background and Darwin-OP played "Turtle beats Hare" scenario when system start running. Storyline from begin to end is depicted from left to right. The most left photo shows the beginning screen, it said "press ENTER to start (the program)".



Fig. 4. Background used in the fairy-tale. Forest is used in intro; race circuit is used when start racing; path and stone under a tree is used when hare sleeps; the final circuit appeared the other animals which hurry for turtle's victory.

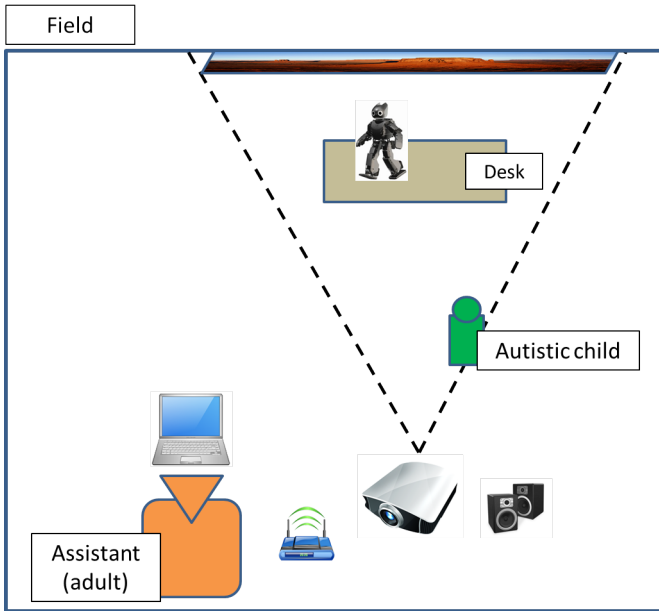


Fig. 6. Hardwares' location in the field.

B. Console system

Console system is programmed in C++. We use the SFML library [2] to handle background changed events and the background sound. The UDP-based Data Transfer Protocol (UDT) [3] handles wireless communication between the console and Darwin-OP. Darwin-OP uses the official API for controlling the motion (action).

Laptop is connected with the projector and speakers. The screen from projector plays the background of a scenario, speakers plays the sounds that belong to the background, and Darwin-OP plays the specific role when specific background

being played according to commands received through broadcast from laptop (see Fig.5).

C. Overall system running

During preparation, Darwin-OP takes two stickers in the way depicted in Fig.7. During the turtle's or rabbit's lines, the robot raises up the turtle sticker or rabbit sticker, respectively (see Fig.8). Other roles and background effects are played by console (see Fig.4). Fig.3 gives an overview of the overall system.

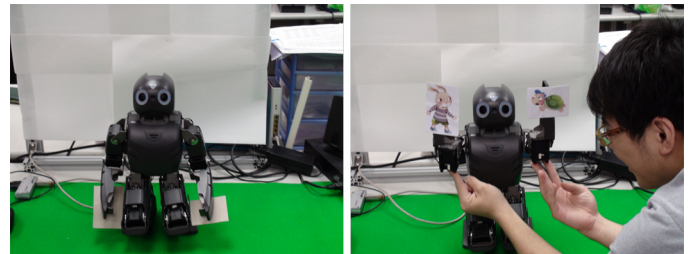


Fig. 7. How Darwin-OP takes turtle and hare, hands down (right), hands up (left).

IV. CONCLUSION AND OUTLOOK

We presented method that is specifically designed to attract the autistic children's attention and discuss methods to measure the rate of this attention.

As further steps, we intent to check to what extent the autistic children are capable to distinguish the role that is played by the humanoid robot. Currently, we use the character stickers to imply and guide child to know robot's role, but the role representation in such way may be ambiguous. To resolve the problem, we raise an idea that is using a head-changed humanoid robot, take "Turtle beats Hare" in example: The head

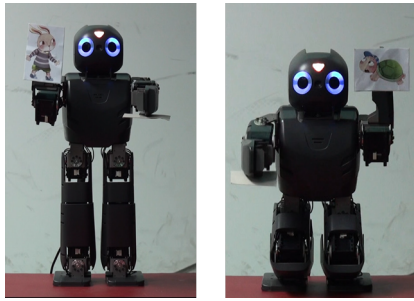


Fig. 8. Darwin-OP raise up hare (right) and turtle (left) stickers with their role action.

turns to turtle when turtle turns talk and vice versa. However, in more simple and brief way, we could prepare more robots, every role played by one robot, this should be distinguished clearly. We will consider it as our next topic.

ACKNOWLEDGMENT

We like to thank NSC (Nation Science Council) in Taiwan provided budget for our project - project number : NSC 101-2221-E-194-038- and NSC 101-2221-E-758 -001-. N. M. M. thanks Minoru Asada and M. Ogino for helpful hints and discussions. Most important we are very grateful for all help of Gary W. Chang and all his advice.

REFERENCES

- [1] The AuRoRA project. <http://www.aurora-project.com/>.
- [2] Simple and fast multimedia library (sfml). <http://www.sfml-dev.org/>.
- [3] Udt: Breaking the data transfer bottleneck (udt). <http://udt.sourceforge.net/>.
- [4] Susan E. Bryson, Sally J. Rogers, and Eric Fombonne. Autism spectrum disorders: Early detection, intervention, education, and psychopharmacological management. *Canadian Journal of Psychiatry*, 48(8):506, 2003.
- [5] Kerstin Dautenhahn and Iain Werry. Issues of robot-human interaction dynamics in the rehabilitation of children with autism, 2000.
- [6] Kerstin Dautenhahn and Iain Werry. Towards interactive robots in autism therapy: Background, motivation and challenges. *Pragmatics & Cognition*, 12(1):1–35, 2004.
- [7] Peter Dobson and Nick Abbots. How puppetry helps pupils on the autism spectrum. *Good Autism Practice (GAP)*, 9(2):26–31, 2008.
- [8] Timothy GIFFORD, Maninderjit KAUR, Dobri DOTOV, Christian WANAMAKER, Gregory DRESSLER, Kerry MARSH, and Anjana BHAT. Using robots to teach musical rhythms to typically developing children and children with autism.
- [9] Ze Ji, Farshid Amirabdollahian, Daniel Polani, and Kerstin Dautenhahn. Histogram based classification of tactile patterns on periodically distributed skin sensors for a humanoid robot. In *RO-MAN, 2011 IEEE*, pages 433–440. IEEE, 2011.
- [10] Rita Jordan. Social play and autistic spectrum disorders a perspective on theory, implications and educational approaches. *Autism*, 7(4):347–360, 2003.
- [11] Susan E. Levy, David S. Mandell, and Robert T. Schultz. *The Lancet*, volume 374, chapter Autism, pages 1627–1638. Nov 2009.
- [12] François Michaud and Catherine Théberge-Turmel. Mobile robotic toys and autism. In *Socially Intelligent Agents*, pages 125–132. Springer, 2002.
- [13] Isabelle Rapin and Roberto F. Tuchman. Autism: Definition, neurobiology, screening, diagnosis. *Pediatric Clinics of North America*, 55(5):1129 – 1146, 2008. Developmental Disabilities, Part I.
- [14] B Robins, K Dautenhahn, R Te Boekhorst, and A Billard. Robotic assistants in therapy and education of children with autism: Can a small humanoid robot help encourage social interaction skills? *Universal Access in the Information Society*, 4(2):105–120, 2005.
- [15] Ben Robins, Kerstin Dautenhahn, and Janek Dubowski. Robots as isolators or mediators for children with autism a cautionary tale. In *Proc of the AISB 05 Symposium on Robot Companions*. AISB, 2005.
- [16] Ben Robins, Paul Dickerson, and Kerstin Dautenhahn. Robots as embodied beings-interactionally sensitive body movements in interactions among autistic children and a robot. In *Robot and Human Interactive Communication, 2005. ROMAN 2005. IEEE International Workshop on*, pages 54–59. IEEE, 2005.
- [17] Ben Robins, Ester Ferrari, and Kerstin Dautenhahn. Developing scenarios for robot assisted play. In *Robot and Human Interactive Communication, 2008. RO-MAN 2008. The 17th IEEE International Symposium on*, pages 180–186. IEEE, 2008.
- [18] Cheng-Hung. Tsai and Pau-Lo Hsu. Realization of the interactive robotic education system for autistic children. Technical report, National Chiao Tung University, 2011.

The Impacts of R&D Contest on Research Activities: An analysis of RoboCup

Koichiro OKAMURA

School of Business Administration, Kwansai Gakuin University

kokamura@kwansai.ac.jp

Abstract— R&D contests have gained popularity in recent years. Empirical studies have, however, scarcely investigated their impacts on research activities to date. This study empirically analyzes the impacts of RoboCup, a robot soccer competition, to researchers' performance in general. The results show that both contest participants and non-participants benefit from it.

Introduction

Governments and other organizations have been increasingly interested in research and development (R&D) contests in recent years[1]. Witnessing the apparent success of recent cases such as U.S. DARPA's Grand Challenge or the Ansari X Prize, academic researchers have also begun initiating R&D contests in the recent years. However, to the best of this author's knowledge, [2] is the only study that has empirically and systematically examined the impacts of R&D contests on research activities. [2] examined a contest for the agricultural machinery in the U.K., which was initiated in 1839 and continued until 1939. With the exception of this study, there only exist oft-mentioned successful cases and a few case studies[3–5]. [3] examined three contests in the last century and concluded that the contests not only promoted innovative activities of sponsor interest but helped develop relevant industries as well. [4] and [5], who examined the recent two contests in space technologies and the contest initiated by the UK parliament in the eighteenth century, respectively, reached similar conclusions. Past successful contests have been designed and held in the later stages of R&D activities close to commercialization, however.

R&D contest is an attractive scheme for governments and researchers. The outcome of the contest is clear and tangible. So are the awarding criteria. The sponsor can avoid the costly and time-consuming process of selecting research projects to fund, which is necessary under the traditional R&D grant system. In typical contests, the sponsors are only responsible for the one-time cash awards for winners. They do not have to fund the contest participants' R&D activities, which can be many times as large as the cash award. It is the participants that are responsible for funding their research activities for the contests. While governments and academic researchers

have become increasingly interested in R&D contests recently, there is no evidence that R&D contests are effective in the basic or applied R&D stages.

This study aims to fill these voids by providing empirical evidence of the impacts of R&D contests on research activities. It particularly focuses on an R&D contest in the basic-to-applied R&D stage, where success is not accompanied by direct monetary rewards. This is a common format for contests initiated in academia. For analysis, this study takes a bibliometric approach and analyzes the research performance of the participants of the RoboCup Soccer contest. It is an R&D contest in robotics, the goal of which is to build a robot soccer team that is capable of playing with human teams by 2050. The RoboCup Soccer contest has been held annually since 1997 and has been growing to date.

This study firstly found that R&D contests seem to have clearer impacts on research productivity than on research quality. Secondly participation has negative impacts on research performance in the short term but that the impacts turn positive in the subsequent years. In addition, repeated participation has positive impacts. Thirdly, researchers who do not participate in RoboCup themselves but publish a scholarly paper(s) with RoboCup participants exhibit higher research performance than those who do not. It indicates spillover effects of R&D contests to the larger research community. The findings seem encouraging for policy makers and researchers who are interested in R&D contests since they have the potential to facilitate not only targeted research but research activities in general.

Data and Sample

Data of the RoboCup Participants

This study analyzes the RoboCup Soccer competition (or simply "RoboCup" hereafter). RoboCup has steadily grown since its creation. At RoboCup 2012, 145 soccer teams participated in the games, a remarkable increase from 38 teams in 1997. The popularity of RoboCup provides an opportunity to examine the roles and impacts of R&D contests in basic and applied R&D stages.

In RoboCup, each team participating in the soccer games is obliged to submit a short team description paper (TDP) to disseminate technical information about their robots or programs. Along with the soccer games, a regular academic symposium is held, in which papers selected from all of the submissions to the symposium are accepted as symposium papers (SPs) based on scientific merit and included in the symposium proceedings. The symposium proceedings keep non-technical, non-research overview papers (OPs) as well. The OPs provide the goal, rules or overview of the RoboCup contests and summarize the results of the games. This study collects information about RoboCup participants in 2001-03 from the TDPs, SPs and OPs, which is supplemented with the participation data of the RoboCup contests held in 1997 (the first contest). This study collected the SPs, OPs, and TDPs from the symposium proceedings. It extended the collection process to a supplementary CD-ROM that keeps TDPs for the RoboCup 2003 and the Internet search for the RoboCup 2002. Then, it cleaned and organized the collected authorship data.

Bibliometric Data

The study collected the bibliometric data of the papers published in academic journals in robotics to measure the performance of individual researchers. The data is used not only to measure the research performance of researchers but to identify researchers who are not participating in the RoboCup contest as well. These researchers are used as a comparison group in the analysis.

The data are obtained from Thomson Reuters' Web of Science database in November 2011 for 18 major academic journals in robotics published in 2001-05, in which not only the RoboCup participants but also robotics researchers in general publish or cite papers most frequently (WoS data). The WoS data were then organized and matched with RoboCup participants. The resulting sample consists of 14,033 researchers, among whom 1,010 researchers participated in RoboCup 2001-03 as TDP and/or SP authors. About 13,000 researchers in the sample did not participate in these RoboCup contests but published at least one paper in the 18 journals between 2001 and 2005.

Analysis

This study carried out a series of statistical analysis to examine the impacts of the RoboCup participation on research performance, particularly research productivity and research quality.

Dependent variables

The study uses two dependent variables to measure research performance in terms of research productivity (quantity) and quality following the literature on patenting and paper publishing performance[6-9].

Research productivity is measured in terms of the number of academic papers published in the 18 journals listed in Table 1 between 2004 and 2005. Researchers who published many papers are more productive than those who did not. Thus, it is natural to use the proxy to measure the research productivity of researchers. The RoboCup contest is usually held in summer every year. The submission of the papers and the registration of the contests are closed a few months in advance. It is, therefore, necessary for the prospective RoboCup participants in either TDPs or SPs to be engaged in the research and in the assembling of robots several months in advance if not within one year. They might have gained new ideas or insights about certain research possibilities through the months-long preparation and through their participation in the RoboCup around summer. The new insights they might have gained would have been incorporated in their new research then.

Table 1. Journals used in the analysis

Journal title
Advanced Robotics
Artificial Intelligence
Automatica
Autonomous Robots
Biological Cybernetics
Fuzzy Sets and Systems
IEEE Robotics & Automation Magazine
IEEE Transactions on Automatic Control
IEEE Transactions on Neural Networks
IEEE Transactions on Robotics and Automation*
IEEE Transactions on Robotics*
IEEE Transactions on Systems Man And Cybernetics, Part B
IEEE/ASME Transactions on Mechatronics
International Journal of Robotics & Automation
International Journal of Robotics Research
Journal of Intelligent & Robotic Systems
Robotica
Robotics and Autonomous Systems

* IEEE Transactions of Robotics and Automation was replaced by IEEE Transaction of Robotics in 2005.

Research quality is measured in terms of the number of forward citations that the papers published between 2004 and 2005 received from those that were published in subsequent years. High research productivity does not necessarily mean high quality of research. This study uses forward citation as proxy of the quality of research. Researchers refer to past literature in their papers if the literature is relevant and important in their study. Papers that may be able to help advance relevant research fields and that have huge impacts are more likely to be cited by subsequent papers. There is a lag time between when a paper appears in a journal and when it is cited by others, however. Sometimes, researchers may need more time to grasp the gist of a paper that they have read. They may also need more time before they can apply the newly acquired knowledge on their own research activities. Moreover, they may need ample time to evaluate the effectiveness of their newly acquired knowledge. If they judge the paper as valuable, they may cite it in the next papers that they write, which may appear in a journal only after several months or so.

Explanatory variables

Repeated SPs authorship is a dummy variable and takes 1 when a researcher participates in RoboCup 2001-03 twice or more. It takes 0 otherwise.

SP 2003 is the number of the symposium papers (SPs) that participants submitted and included in the symposium proceedings for the symposium that was held during the RoboCup 2003 contest. *SP 2003* takes a value of either 0 (i.e. no SP) or 1 (i.e. one SP) for the sheer majority of researchers. Few participants published more than one SP. Similarly, *SP 2002* and *SP 2001* are the number of SPs that were included in the symposium proceedings for the RoboCup 2002 and 2001 contests, respectively.

Collaboration with RoboCup participants is a dummy variable and takes 1 when a researcher does not participate in RoboCup him/herself but publishes a scholarly paper(s) which is co-authored with a RoboCup participant(s). It takes 0 otherwise. The variable is to assess the spillover effects of R&D contests to the larger research community.

Control variables

Seventeen control variables are also included in analysis to control for soccer leagues and years, first-mover advantage from the participation in the first RoboCup, baseline of research performance, etc.

Table 2. Summary statistics ($n=14,033$)

Variable	Mean	Std. Dev.	Min.	Max.
<i>Research productivity</i>	0.69	1.35	0	24
<i>Research quality</i>	13.35	42.86	0	1,498
<i>Repeated SPs authorship</i>	0.00	0.07	0	1
<i>SP 2003</i>	0.01	0.14	0	3
<i>SP 2002</i>	0.01	0.10	0	3
<i>SP 2001</i>	0.01	0.14	0	4
<i>Collaboration with RoboCup participants</i>	0.02	0.13	0	1

Analysis results

The dependent variables—*Research productivity* and *Research quality*—are non-negative count data of papers and forward citations with wide variance, respectively. A negative binomial model is applied to address the over-dispersion in those variables[10], [11]. In addition, robust standard errors were used to address the heteroscedasticity in the data for analysis.

Tables 2 and 3 provide descriptive statistics of the dependent and explanatory variables and the correlation matrix. Estimations were made with Stata 12. Table 4 presents the estimation results for a full model. Only the results for main explanatory variables are shown there due to space limitation. Firstly, R&D contests seem to have clearer impacts on research productivity (Model 1) than on research quality (Model 2). A larger numbers of coefficients for explanatory variables are found significant in Model 1 than in Model 2. The signs of coefficients are same in both Models.

The coefficient for *SP 2003* is negative and significant in Model 1 while it is negative but not significant

Table 3. Correlation among dependent and explanatory variables ($n=14,033$)

Variable	1	2	3
1 <i>Research productivity</i>			
2 <i>Research quality</i>	0.462		
3 <i>Repeated SPs authorship</i>	-0.016	-0.018	
4 <i>SP 2003</i>	-0.034	-0.022	0.504
5 <i>SP 2002</i>	-0.016	-0.018	0.556
6 <i>SP 2001</i>	-0.023	-0.025	0.502
7 <i>Collaboration with RoboCup participants</i>	0.029	0.028	-0.009

Variable	4	5	6
5 <i>SP 2002</i>	0.248		
6 <i>SP 2001</i>	0.267	0.272	
7 <i>Collaboration with RoboCup participants</i>	-0.014	-0.011	-0.013

in Model 2. The coefficients for *SP 2002* and *SP 2001* are positive and significant in Model 1 while they are positive but not significant in Model 2. They suggest that participation has negative impacts on research performance in the short term but that the impacts turn positive in the subsequent years.

The coefficient for *Repeated SPs authorship* is positive and significant in Model 1 while it is negative and insignificant in Model 2. Repeated participation leads to higher productivity while it may not necessarily be translated to high-quality research, which other researchers regard as new and important.

The coefficient for *Collaboration with RoboCup participants* is positive and significant in both Model 1 and 2. That is, researchers who do not participate in RoboCup themselves but publish a scholarly paper(s) with RoboCup participants exhibit higher research performance than those who do not. It indicates spillover effects of R&D contests to the larger research community.

The findings seem encouraging for policy makers and researchers who are interested in R&D contests since they have the potential to facilitate not only targeted research but research activities in general.

Acknowledgment

This work is supported by Japan Society for the Promotion of Science under Grant Number 24530511 (Grant-in-Aid for Scientific Research (C)) and Kwansai Gakuin University's Individual Special Research A Grant.

References

- [1] National Academy of Engineering, Concerning Federally Sponsored Inducement Prizes in Engineering and Science. Washington, DC: National Academy Press, 1999.
- [2] L. Brunt, J. Lerner, and T. Nicholas, "Inducement prizes and innovation." London, UK, 2008.
- [3] L. Davis and J. Davis, "How effective are prizes as incentives to innovation? Evidence from Three 20th Century Contests," in DRUID Summer Conference 2004, 2004.
- [4] L. Kay, "The effect of inducement prizes on innovation: evidence from the Ansari X Prize and the Northrop Grumman Lunar Lander Challenge," *R&D Management*, vol. 41, no. 4, pp. 360–377, 2011.
- [5] C. Stallbaumer, "From longitude to altitude: Inducement prize contests as instruments of public policy in science and technology," *Journal of Law, Technology & Policy*, vol. 1, pp. 117–158, 2006.
- [6] A. B. Jaffe and M. Trajtenberg, *Patents, Citations and Innovations: A Window on Knowledge Economy*. Cambridge, MA: MIT Press, 2002.
- [7] A. B. Jaffe, M. Trajtenberg, and R. Henderson, "Geographic localization of knowledge spillovers as evidenced by patent citations," *The Quarterly Journal of Economics*, vol. 108, no. 3, pp. 577–598, 1993.
- [8] R. N. Kostoff, "Citation analysis of research performer quality," *Scientometrics*, vol. 53, no. 1, pp. 49–71, 2002.
- [9] F. Narin and K. S. Hamilton, "Bibliometric performance measures," *Scientometrics*, vol. 46, no. 3, pp. 293–310, 1996.
- [10] W. H. Greene, *Econometric Analysis*, 6th ed. Upper Saddle River, NJ: Pearson Prentice Hall, 2008.
- [11] J. Hausman, B. H. Hall, and Z. Griliches, "Econometric models for count data with an application to the patents-R&D relationship," *Econometrica*, vol. 52, no. 4, pp. 909–928, 1984.

Table 4. Negative binomial regression of research performance

Variable	Model 1 (Research productivity)	Model 2 (Research quality)
<i>Repeated SPs authorship</i>	2.01*** (0.77)	-0.34 (0.88)
<i>SP 2003</i>	-2.26* (1.17)	-2.64*** (0.70)
<i>SP 2002</i>	1.95*** (0.68)	0.76 (0.67)
<i>SP 2001</i>	1.76*** (0.56)	0.6 (0.62)
<i>Collaboration with RoboCup participants</i>	0.28** (0.11)	0.41*** (0.14)
N	14,033	14,033
Chi ² value of Wald test	219.34	510.66
Log-likelihood	-15,006.81	-37,279.87

Robust standard errors in parentheses

* p<0.10, ** p<0.05, *** p<0.01