

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

Deciding action based on voxel data of multiple humanoid robots
produced from external camera images

植田 康生 木村 堯海 勢川 友樹 升谷 保博

Yasuo UEDA Takaumi KIMURA Tomoki SEGAWA Yasuhiro MASTANI

大阪電気通信大学

Osaka Electro-Communication University

Abstract

RoboCupSoccer SSL Humanoid, soccer game by humanoid robots using external cameras, was proposed. The problems for the next phase of the game are real-time 3D shapes acquisition based on images of multiple cameras, deciding action based on the 3D shape, and shared vision system. One of the authors proposed that teams share voxel data of 3D space on the game field provided by “voxel server” in 2010. In this paper, the authors propose a method of estimating pose of the humanoid robot based on the voxel data. The feature vector is defined as a histogram of voxel distribution. The estimation is carried out by matching feature vectors of the provided data and previously stored data. Moreover, a method of deciding action of the humanoid robot based on the estimated pose is proposed. Under the condition of SSL Humanoid, it is shown that both the voxel server program and the client program which decides action are executed and that the robot can play soccer-like action.

1 はじめに

RoboCupSoccer のリーグの一つで、車輪型のロボットがサッカーを行う小型ロボットリーグ (SSL: Small Size robot League) [長坂, 2011] は、他のリーグと異なり、外部カメラを用いてロボットなどの位置を認識して制御を行うおかげで、パスなどの協調的なプレーができるまでに発展してきている。この競技に参加するには、様々なノウハウや経験が必要であるが、その方法は確立しつつあり、研究

としては、飽和気味である。また、技術レベルの高度化と必要な費用の増大により新規参入が難しくなっている。

このような現状を踏まえて、SSL の特徴を継承しつつ、新たな研究を展開し、かつ、参加者の裾野を広げることを目的として、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 SSL Humanoid が 2007 年に提案された [升谷, 2010]。

現状では、SSL と同様に、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られたマーカのパターンを撮影し、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画像に基づいて、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の行動決定を行う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利用し、SSL-Vision から送られてくるフィールド上の位置情報を共有する [Zickler, 2009]。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。そこで、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」を作成した [木村, 2010]。

本稿では、ボクセルデータの精度向上のためにカメラ台数を 4 台から 8 台へ増やし、ボクセルデータからロボットのポーズの推定を行い、その結果から相手ロボットのポーズを考慮した自機の行動決定についても検討する。

2 ボクセルサーバ

ボクセルサーバ [木村, 2010] は、SSL Humanoid の視覚の 3 次元化のために、チームがボクセルデータを共有する具体的な方法として開発された。背景差分法を用いてシ

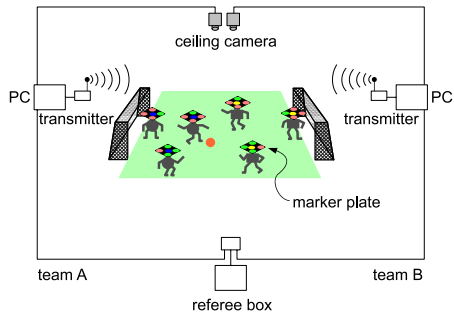


Figure 1: Initial phase of the SSL Humanoid (2009)

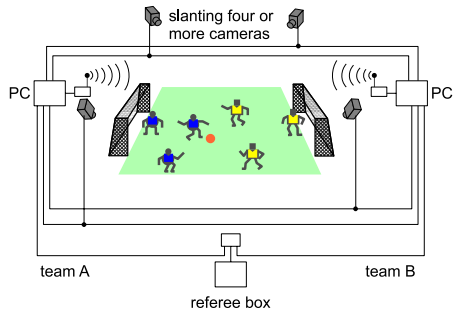


Figure 2: Final phase of the SSL Humanoid (2015)

レット画像を作成し、視体積交差法の一つである Space Carving Method(SCM) を用いて 3 次元形状をボクセルの分布として取得する．そして、そのボクセルデータをランレングス圧縮し、クライアントへ配信する．実時間での処理を行うため、SCM を並列処理している．

SSL Humanoid を想定して、対象とする空間を競技フィールド上の $3025[\text{mm}] \times 4050[\text{mm}] \times 500[\text{mm}]$ とする．

2.1 カメラ台数の追加

[木村, 2010] では、Figure 3 で黒色で示すようにフィールドの角の外側の上部から斜め下向きに撮影する 4 台のカメラを使っていたが、ロボットの位置によっては胴体と腕や両脚の分離ができない場合があった．また、オブジェクトがカメラから離れている場合、量子化誤差の影響で必要以上にボクセルが削られることがあった．そこで、精度向上のため、新たに 4 台のカメラを Figure 3 の青色のように設置した．実験に使用した PC の仕様を Table 1 のサーバ PC とクライアント PC1 である．クライアント PC ではボクセルの表示を行っている．ボクセルの 1 辺の長さを $10[\text{mm}]$ に設定している．

取得されたボクセルの一例として、4 カメラの場合と 8 カメラの場合の同じ位置の同じ形状のロボットの処理結果を Figure 4 に示す．8 台の場合の方が、余分なボクセルが除去されロボットの形状がはっきり取得できていることがわかる．一方、1 秒間の処理回数は、4 カメラの場合の約 30 回に対し、8 カメラの場合は約 23 回と低下した．

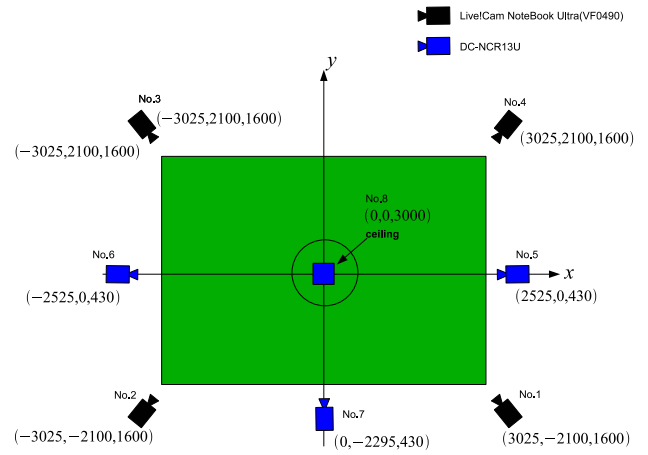


Figure 3: Camera layout

Table 1: Specifications of the experimental PCs

Server PC	
OS	Windows 7 Professional 64bit
Main Memory	4GByte
Graphic Board	NVIDIA GeForce GTX260
DirectX	DirectX 11
CPU	Intel Core i7 950(3.19GHz)
Client PC 1	
OS	Windows Vista Home Premium 32bit SP2
Main Memory	3GByte
Graphic Board	Mobile Intel 4 Series Express Chipset
DirectX	DirectX 11
CPU	Intel Core 2 Duo U9400(1.4GHz)
Client PC 2	
OS	Windows 7 Professional 64bit
Main Memory	4GByte
Graphic Board	ATI Mobile Radeon HD 5800 Series
DirectX	DirectX 11
CPU	Intel Core i7 X940(2.13GHz)
Client PC 3	
OS	Windows 7 Professional 64bit
Main Memory	4GByte
Graphic Board	ATI Radeon HD 4550
DirectX	DirectX 11
CPU	Intel Core i5 650 (3.2GHz)

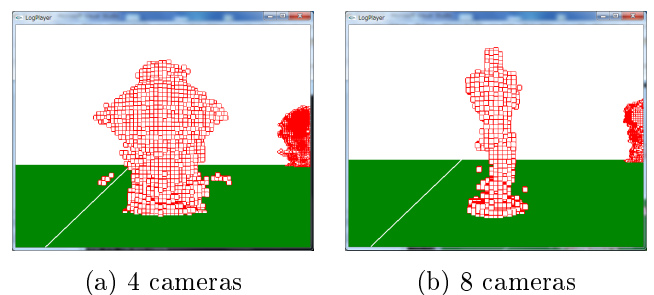


Figure 4: Result of voxel acquisition

しかし、ヒト型ロボットの応答性能が高くないので、当面はこれで十分だと考えられる．以降の節では 8 台のカメラを利用する．

3 ボクセルデータに基づくポーズ推定

ボクセルサーバより送られてきたボクセルデータからフィールド上のロボットのポーズ推定を行う．本研究では、

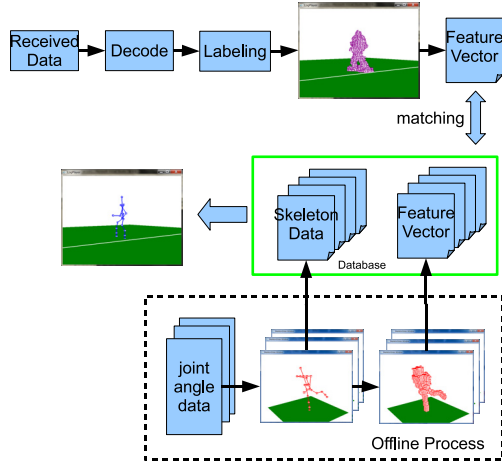


Figure 5: Data flow of the proposed system

特徴ベクトルとしてボクセル分布のヒストグラムを用いる．ロボットの様々なポーズの特徴ベクトルをデータベースとしてあらかじめ用意しておく．入力されたボクセルデータの特徴ベクトルとマッチングを行い最も類似したポーズを結果として出力する．以上の方法を考案するにあたって[佐川, 2008]を参考にした．

処理の過程を Figure 5 に示す．まず，事前にヒト型ロボットに登録されたすべてのポーズの関節角からスケルトンを生成する．そして，そのスケルトンにボクセルを肉付けし，得られたボクセルデータから特徴ベクトルを算出し，スケルトンを共にデータベースに保存しておく．次に，実データをボクセルサーバより受信すると復号し，空間全体でラベリングを行い，ボクセルをグループ化する．グループ化された個々のボクセル群について特徴ベクトルを生成し，データベースと比較する．比較した中で最も類似度が高い特徴ベクトルに対応するスケルトンデータをポーズの推定結果とする．各処理を以下の節で詳しく説明する．

3.1 ラベリング

ボクセルサーバから得られたボクセルデータには，複数のオブジェクトが含まれているので，それを分離するにはラベリングを行う必要がある．そこで，空間全体のボクセルデータに対して3次元のラベリングを行う[何, 2009]．ラベリングの処理では，全ボクセルに対して走査を行うため計算量が膨大であるため，空間を分割し複数のスレッドで処理して高速化する．

3.2 特徴ベクトル

ボクセルデータ p の重心から床面に下ろした垂線を軸とする円柱と考え，これで決まる領域で定義されるヒストグラムを特徴ベクトルとする．円柱の半径はヒト型ロボットが倒れたときのことを考慮し，ロボットの身長半分以上とする．そして，Figure 6 のように高さ，半径，円

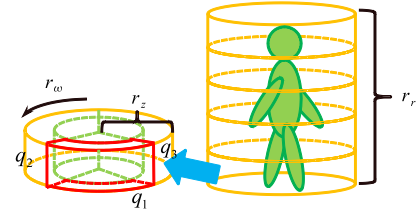
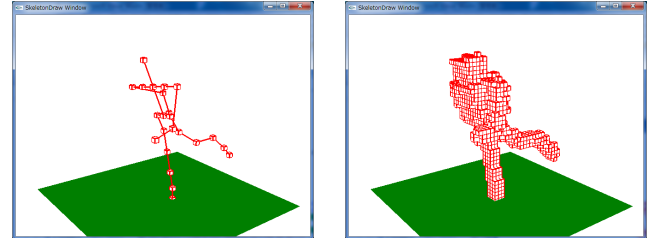


Figure 6: Regions of histogram for the feature vector



(a) Skeleton

(b) Voxel

Figure 7: Artificial voxel for the database

周をそれぞれ r_r , r_z , r_w に分割し3次元空間を領域を $r_q (r_q = r_r r_z r_w)$ 個設ける．そして，インデックス値をそれぞれ $i_r (0 \leq i_r < r_r)$, $i_z (0 \leq i_z < r_z)$, $i_w (0 \leq i_w < r_w)$ としたとき領域のインデックス値 i は式 (1) で得られる．

$$i = i_r + i_z r_r + i_w r_r r_z \quad (1)$$

次に，各領域を占めるボクセルのヒストグラムを求める．ボクセル群 p の重心位置を (g_x, g_y, g_z) とし，領域 b_i 内にあるボクセル $v(v_x, v_y, v_z)$ の中心軸からの距離 d_v に応じた重み付けを行う． p に対する特徴ベクトル $Q(p)$ の i 番目の要素は，式 (3) で得られる．

$$d_v = \sqrt{(g_x - v_x)^2 + (g_y - v_y)^2} \quad (2)$$

$$Q_i(p) = \frac{\sum_{v \in b_i} d_v}{\sum_v d_v} \quad (3)$$

3.3 データベース作成

本研究では，ヒト型ロボットのポーズ(各関節角度)のデータを利用しスケルトンを作り出し，ボクセルデータへ変換する．このボクセルデータの特徴ベクトルに変換し，特徴ベクトルに対応するスケルトンの関節座標位置と共にデータベースに保管する．

ポーズ数を n としたとき，1ポーズにつきロボットの方角を r_w 個の方角を変更したデータを作る必要がある．そのため，総データ数は $n r_w$ 個となる．まず，各ポーズを利用し順運動学に基づきスケルトンの生成を行う．そして，スケルトンを利用し各関節間のリンクを直方体で表現し，ボクセルの肉付けを行う．一例を Figure 7 に示す．ポーズ番号を $\alpha (\alpha = 1, 2, 3, \dots, n r_w)$ をとしたとき，特徴ベクトル q' ，回転後の関節位置の座標 S' ，角度の分解能

$e(e = 1, 2, 3, \dots, \frac{360}{r_\omega})$, スケルトンの回転角度 θ を式 (4) ~ (6) で得る .

$$\theta_\alpha = -180 + \frac{360}{r_\omega} e \frac{\pi}{180} \quad (4)$$

$$q'_{\alpha a} = Q(R_y(\theta_\alpha) p_a) \quad (5)$$

$$S'_{\alpha a} = R_z(\theta_\alpha) s_a \quad (6)$$

データベースには $\theta_\alpha, q'_{\alpha a}, S'_{\alpha a}$ を保管する .

3.4 データベースとのマッチング

3.3 節で生成したデータベースとラベリング後のボクセルとのマッチングを行う . ラベリングを行ったボクセルのグループ $p(t)$ とデータベース内の特徴ベクトルの比較のために , 類似度の指標として Bhattacharyya 係数[Kailath, 1967]を用いる . 類似度を $\phi(t)$, $l(l = 1, 2, \dots, nr_\omega)$ をデータベース内のポーズ番号をとしたとき , 式 (7) が得られる . $0.0 \leq \phi(t) \leq 1.0$ であり , 1.0 に近づくほど類似している .

$$\phi_\alpha(t) = \sum_{r=0}^{r_q} \sqrt{Q_r(p(t)) q'_{\alpha r}} \quad (7)$$

そして , 得られた $\phi(t)$ を降順に整列し j を保存する関数 sort を利用し式 (8) により $m(t)$ を得る .

$$m(t) = \text{sort}(\phi(t)) \quad (8)$$

実データとデータベースとのマッチングは , 全てのポーズに対して式 (7) の計算を行うため時間がかかる . そこで , 本研究ではこの計算を複数のスレッドに分割して処理し高速化を図っている .

3.5 連続性を考慮したマッチングの改善

ロボットの前後方向の判断が難しいポーズ (直立など) では , 1 フレームのデータのマッチングだけでは , 正しい結果が得られない場合がある . そこで , 前フレームとの連続性を考慮したマッチングを行う . 前フレームの類似度をソートしたポーズ番号 $M(t-1)$, 現在の類似していると判断したポーズ番号を $M(t)$, スケルトンの向いている方向の許容範囲 $\pm\theta$, 関節角の前フレームからの差分の許容範囲 $\pm\omega$ とし , すべての関節角内で前フレーム関節角の最も差分が大きい角度を返す関数 diff とし , 式 (9) ~ (11) でポーズ番号を推定する .

$$M(t) = \begin{cases} m_1(t) & (t = 0) \\ M'(m_j(t)) & (t \neq 0) \end{cases} \quad (9)$$

$$M'(m_j(t)) = \begin{cases} M''(m_j(t)) \\ (-\theta \leq l(M(m(t-1))) - l(m_1(t)) \leq \theta) \\ M(m_{j+1}(t)) \quad (\text{otherwise}) \end{cases} \quad (10)$$

$$M''(m_j(t)) = \begin{cases} m_j(t) \\ (-\omega \leq \text{diff}(m_j(t), M(t-1)) \leq \omega) \\ M'(m_{j+1}(t)) \quad (\text{otherwise}) \end{cases} \quad (11)$$

4 ポーズ推定の評価実験

4.1 実験の環境と装置

実験に用いたサーバ PC とクライアント PC2 の仕様を Table 1 に示す . クライアント PC は , 受信したデータを復号しラベリングを行い , ポーズ推定を行い , その結果を表示する .

4.2 評価方法

ヒト型ロボットとして京商の MANOI AT01(身長 378[mm]) を用い , 遠隔操作で動作させて , 実時間でボクセルデータを生成し , ポーズを推定する実験を行う . MANOI に登録されているポーズは 230 種類で , 各ポーズに対して 2000 から 3000 個のボクセルから成るボクセルデータが生成された . 次に , 各ポーズのボクセルデータに対して特徴ベクトルを算出する . その際のパラメータは , $r_r = 8$, $r_z = 3$, $r_\omega = 24$ で , 領域数は $r_q = 576$ となり , これが特徴ベクトルの次元数となる .

4.3 結果

推定の例として四つの入力データによるポーズ推定の結果を示す . Figure 8 の左側にボクセルサーバからの受信したボクセルデータを示す . これに対して , 推定されたポーズを右側に示す . サーバから送られてきたデータを復号しポーズを推定するまでの所要時間は 80[mm] 程度であった . 実時間と呼べるようになるには , さらに高速化が必要であるが , 対象としているヒト型ロボットは応答性が高くないため , この程度でも制御に使えると考えられる .

5 推定されたポーズに基づく行動決定

本節では , ボクセルデータから推定したポーズに基づいてロボットの行動決定を行う . ロボットには予めモーションが登録されており , 外部の PC で行動決定を行い実行するモーション番号を選択し , それを無線でロボットへ送信する . 本研究では , Table 2 に示す 9 種類のモーションを用いる .

5.1 オブジェクトの識別

ボクセルサーバより得られるボクセルデータには色情報がないため , 色やパターンによってオブジェクトを識別することができない . そこで , 初期状態におけるボクセル群とロボット番号の対応は与えることにする . また , ボールについては , 実際のボールの体積と近いボクセル群をボールと認識させる . その後は前回のデータと重心位置が最も近いボクセル群を同じオブジェクトとして認識させる .

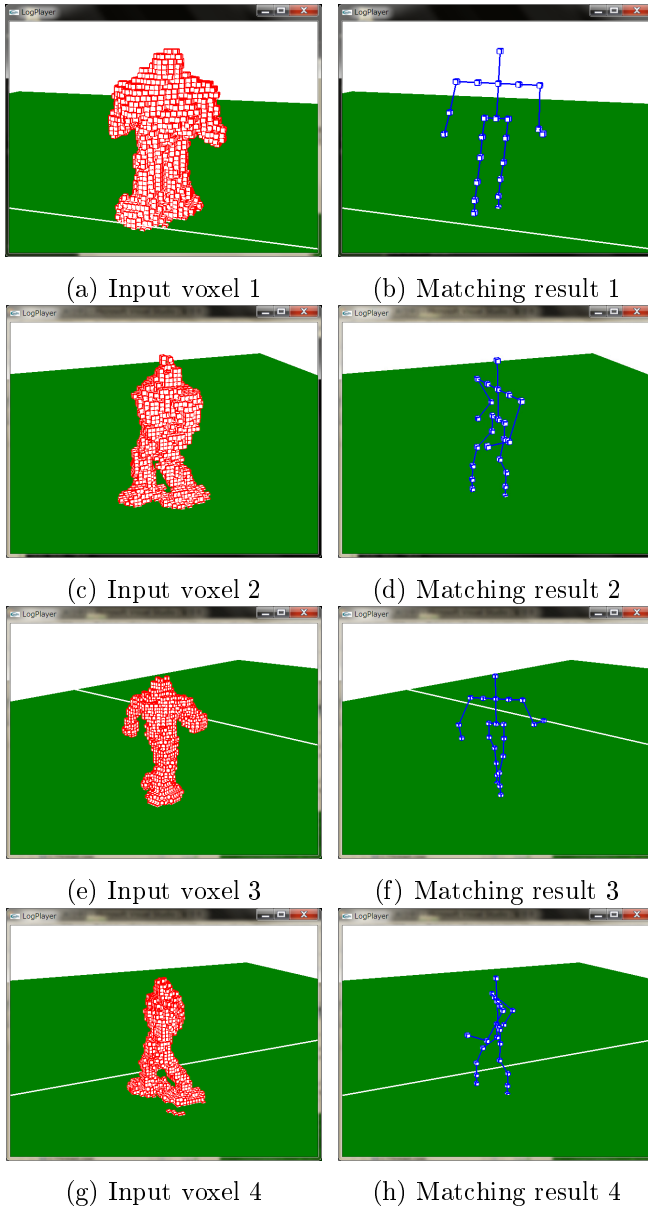


Figure 8: Examples of the pose estimation

5.2 キック行動 (パス)

ロボットのポーズを考慮した行動決定の例として、相手ロボットへボールをキックするパスについて述べる。相手のロボットの状況からキックするかどうかを判断する。タスクの流れは以下の通りである (Figure 9)。

1. キックの目的地 (x_t, y_t) からボールへ伸ばした線分から d_k [mm] 延長した点 (x, y) へ移動する
2. ボール (x_b, y_b) の方向を向く
3. ボールに距離 d_b まで小前歩行で近づく
4. 次の条件にあてはまるとキック
 - ボールのキック先に相手機が到着している
 - 相手機がボールの方向を向いている
 - 両足を揃えている
 - 相手機が倒れていない

Table 2: Kind of the motion

Number	Motion
0	Stop
1	Walk
5	Right spin
6	Small right spin
9	Left spin
10	Small left spin
11	Small walk
80	Right leg kick
144	Left leg kick

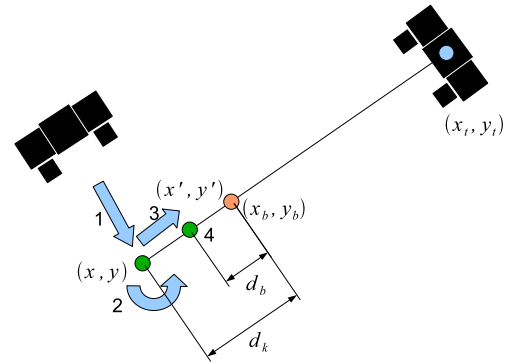


Figure 9: Steps in the pass task

6 行動決定の評価実験

ボクセルサーバから提供されたデータからポーズ推定を行い、それに基づき行動決定を行う一連の処理を評価する実験を行う。

6.1 実験の環境と装置

4.1 節と同じ環境に加え新たに Table 1 のクライアント PC3 を使用する。クライアント PC ではポーズ推定の結果により行動決定を行いロボットへコマンドを送信する。

6.2 方法

相手機の行動を考慮し相手機がボールを受け取れる体勢ならばパスを出す判断が可能かどうか実験する。ボールは $(x_t, y_t) = (1000, 500)$ へ向けてボールをキックするものとし、キックを開始する条件は 5.2 節の通りとする。

6.3 結果

フィールド座標系におけるロボットとボールの軌跡を Figure 10 に示す。また、行動中のロボットの位置や命令などの時間変化を Figure 11 に示す。Figure 11 の各線の意味は以下の通りである。

- command 自機に送っている命令番号 (Table 2)
- Leg status 両脚が揃っているかどうかの判定
- robot2(y) 相手機のロボット y 座標 [mm]
- robot2(dir) 相手機のロボットの方向 [deg]
- ball(x) ボールの x 座標 [mm]

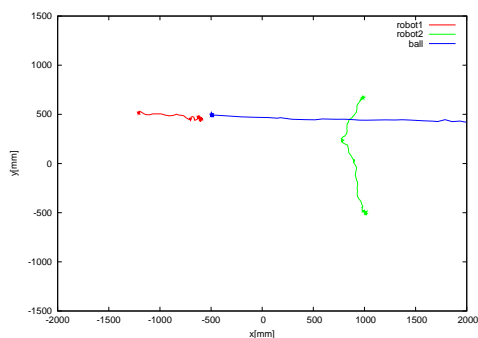


Figure 10: Trajectories of the pass task

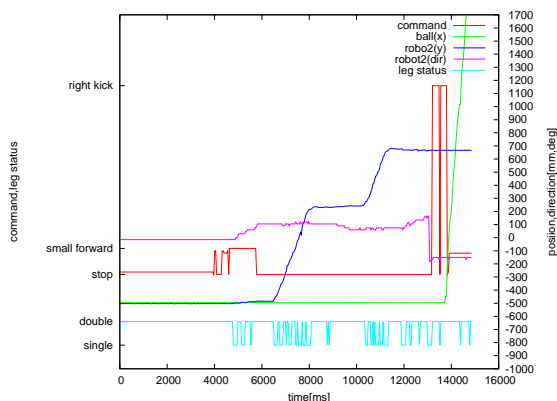


Figure 11: Command and position of the robot and the ball in the pass task

6.4 考察

13500[ms]あたりでボールを受け取ることが可能な条件が揃いキックコマンドを送出していることがわかる．これにより，相手のポーズを考慮した行動決定が可能であることを示せた．この方法を利用すると，相手の足の位置がわかるため，インターセプトを行うことなど様々なサッカーでの行動の実現が期待できる．

7 おわりに

本稿では，RoboCup SSL Humanoid の 3次元化に向けて，「ボクセルデータをチームで共有する」という提案の要となる「ボクセルサーバ」のカメラ台数を4台から8台に増やし性能を評価した．次に，ボクセルサーバから提供されるデータを利用する側の処理として，ボクセルデータからポーズ推定をする方法，推定されたポーズに基づきロボットの行動決定（行動命令の選択）する方法を提案し，実際のPCに実装し，ボクセルサーバと組み合わせて評価実験を行った．評価の一例として，ボールをパスする実験の結果を示した．提案システムでは，パスする相手ロボットのポーズを考慮して行動を決定することができた．

本研究の今後の課題や展望について述べる．ボクセル

データには色情報が含まれていないため，ロボットにボールが触れた場合，同一のボクセル群と判断されボールが認識できない場面がある．そのため，ボクセルデータに色情報を付加することを検討する．本稿の実験では，登録されたポーズを再生する方式のロボットを使い，その登録されたポーズに基づいてデータベースを作成した．したがって，登録されていないポーズに対しては，適切な推定ができない可能性がある．しかし，提案する方法は，細かな差異を吸収する方法であり，ある程度は対応可能と思われる．今後，特徴ベクトルの作り方やデータベースのマッチングの方法を改良し，より多くのポーズに対して高速にマッチングが行えるようになれば，この問題は解決できると思われる．

参考文献

- [何, 2009] 何立風, 巢宇燕, 鈴木賢治, 中村剛士, 伊藤英則: 三次元2値画像における高速ラベル付けアルゴリズム, 電子情報通信学会論文誌 D, Vol. J92-D, No.12, pp.2261–2269, 2009.
- [Kailath, 1967] T.Kailath: The divergence and bhat-tacharyya distance measures in signal selection, IEEE Trans. on Comm. Technology, Vol.15, pp.52–60, 1967.
- [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.
- [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.
- [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.
- [佐川, 2008] 佐川裕一, 下坂正倫, 森武俊, 佐藤知正: 三次元ボクセルに基づく高速オンライン人体姿勢推定, 日本ロボット学会誌, Vol. 26 No.8, pp.913–924, 2008
- [Zickler, 2009] S.Zickler, T. Laue, O. Birbach, M. Wongphati, and M. Veloso: SSL-Vision: The Shared Vision System for the RoboCup Small Size League, RoboCup 2009, Robot Soccer World Cup XIII, Springer-Verlag, pp.425–436, 2009.