

# 分散ビジョンシステムを用いた モデルベースドモーショントラッキング\*

森 敬充 \*<sup>1</sup> 倉爪 亮 \*<sup>2</sup> 長谷川 勉 \*<sup>2</sup> 村上 剛司 \*<sup>2</sup>

## Model-based motion tracking by distributed vision system

Takamitsu Mori \*<sup>3</sup> Ryo Kurazume\*<sup>4</sup> Tsutomu Hasegawa\*<sup>4</sup> Kouji Murakami\*<sup>4</sup>

\*<sup>3</sup>Kyushu University, Dept. of Graduate School of Information Science and Electrical Engineering  
744, motooka, nishi-ku, Fukuoka-city, Fukuoka, 819-0395, Japan

\*<sup>4</sup>Kyushu University, Dept. of Graduate Faculty of Information Science and Electrical Engineering  
744, motooka, nishi-ku, Fukuoka-city, Fukuoka, 819-0395, Japan

This paper proposes a real-time model-based motion tracking system using distributed cameras. For developing a co-existence society of human and robots, the recognition of human gestures is an indispensable technique. This paper proposes a motion tracking system using distributed network cameras which are placed in an environment such as a street, a town, etc. The model-based motion tracking is adopted and asynchronous process is invoked for updating gesture estimation in each camera. The 2D distance map created by Fast Marching Method is utilized to calculate the errors between the captured images and projected images of the model on the image plane in real-time. Experimental results including real time motion tracking using distributed cameras are successfully carried out.

**Key Words** : Motion capture, Motion tracking, Multiple cameras, Level Set Method

### 1. は じ め に

ロボットの活動の場が工場内から人間が共生する日常環境へと移行し、ロボットを取り巻く状況が複雑になるにつれて、ロボットが処理すべき情報は飛躍的に増加する。しかし、この膨大な情報処理を全てロボット単体で行わせることは、搭載するセンサや計算機の能力から非現実的である。一方、ロボットを導入する日常環境をロボットが活動しやすいように整備し、例えば環境側に予め多様なセンサを分散配置して、ロボットが自身で獲得する情報に加えて環境側からも必要な情報を得ることができれば、超高機能なロボットでなくとも実世界で使用できる実用的なシステムが構築できる。このようなロボットを取り巻く環境の整備を「環境構造化」といい、現在研究が精力的に進められている [1]。

そこで本論文では、人間とロボットが共生する環境において、安全なインタラクションの実現に必要な不可欠な基本技術として、日常環境に分散配置された多数台のカメラを用いて、環境内を移動する人物の位置や移動方向、手足の動きなどの動作を推定するシステムを開発する。

カメラ画像から人物動作を推定する手法は、モデル

ベースド・モーショントラッキングとラーニングベースド・モーショントラッキング(事例ベース)に大別される。ラーニングベースド・モーショントラッキングは、まず学習時に、事前に用意された多数の人体姿勢と画像特徴の対を学習し、実行時には、取得された画像特徴から学習結果との比較、補間により、人体の姿勢を推定するものである。一方、モデルベースド・モーショントラッキングは、人体の3次元モデルとカメラ画像上の人物像を比較することで人物の位置姿勢を推定し計算機内に取り込む手法である [2-12]。本手法の多くは、まず適当な初期姿勢で人体モデルをカメラ画像上へ投影し、その投影像とカメラ画像との間でエッジや輪郭線などの画像特徴量を比較し、それらが最も良く一致する人体モデルの姿勢を探索する。しかし画像特徴量の比較に必要な計算コストが大きく、特に多数台のカメラを用いた場合、リアルタイム処理の実現には計算量の削減が大きな課題である。一方、ロボットと人間の距離が近くなるに従い、円滑かつ安全なインタラクションを実現するには、動作推定の遅延は限りなく少なくする必要がある。そこで本研究では、カメラ画像上で高速に構築可能な2次元距離場を、個々のカメラにおける姿勢パラメータの修正量の算出に用い、求められた修正量を同一の3次元モデル上で統合することにより、複数台のカメラを用いてもリアルタイム処理が可能なモーショントラッキングシステムを開発する [13, 14]。

これまでに、C.Sminchisescu らによってシルエット

\* 原稿受付 平成 21 年 9 月 7 日

\*<sup>1</sup> 九州大学大学院システム情報科学府 (〒 819-0395 福岡県福岡市西区元岡 774 番地)

\*<sup>2</sup> 九州大学大学院システム情報科学研究院 (〒 819-0395 福岡県福岡市西区元岡 774 番地)

Email: mori@irvs.is.kyushu-u.ac.jp

ベースの2次元距離場を用いた手法 [15] が提案されているが、本研究では2次元距離場を粗密かつ輪郭線ベースで構築することで、より高速かつ高精度なモーショントラッキングを実現するものである。

本論文ではまず2台のカメラを用いたモーショントラッキングシステムを構築し、提案手法の基本動作の確認実験を行った結果を示す。次に環境内に分散配置された多数台のカメラを用い、人物の位置、姿勢に応じて動作推定に適したカメラを次々と切り替えることで、自己遮蔽や自己運動にも頑強なシステムを構築し、人物動作の推定実験により提案手法の有効性を検証する。

## 2. 2次元距離場を用いたモーショントラッキングシステム

本研究では、著者らにより提案された粗密2次元距離場を用いた高速なトラッキング手法 [13, 14, 16] を用いる。この手法は、背景差分により得られたカメラ画像上の人物の輪郭線に対して、画像内での粗密2次元距離場を高速に計算し、画像上に投影された3次元人体モデルの輪郭線上の距離値からモデルの姿勢修正量を求めることで、動作の推定を行うものである。本手法の概要を以下に示す。

1. Active Contour Model (Snakes や Level Set Method [17, 18]) を用いて、2次元画像から人体のシルエット輪郭線を抽出する。
2. Fast Marching Method [17] (Appendix 参照) を用い、得られたシルエット輪郭線からの2次元距離場を構築する。
3. 3角パッチで記述された3次元人体モデルを適当な初期姿勢、位置に配置し、その像を2次元画像上に投影する。
4. 3次元人体モデルの投影像の輪郭線に対応する3角パッチを求める。
5. 得られた輪郭線上の3角パッチ投影点における2次元距離値を用いて誤差を計算する。
6. 誤差を減少させるための3次元人体モデルの各関節、重心位置、姿勢の修正量を決定する。
7. 1-6を3次元人体モデルの投影像と2次元画像が一致するまで繰り返す。

本論文では上述した手法を、姿勢変化も含む全身のモーショントラッキング、および分散カメラを用いた人間動作の推定システムに拡張する。まずこれまでに提案した粗密2次元距離場を用いた高速なトラッキング手法について概説する。

**2.1 3次元人体モデル** 本論文で使用する3次元人体モデルは、図1に示すように、14個のリンクが13個の関節によって連結され、全21自由度を有している。なお、図1において座標系の数字は各関節での自由度を示している。ただし本論文では、3次元人体モデルの位置姿勢は、胴体中心の3次元位置と胴体中心を通る床面に垂直な軸周りの回転角度、及び首、手首、足首を除く8個の関節（各腕2、各脚2、全16自由度）の回転角度によって決定されるものとする。またこの3次元人体モデルの表面形状は、ほぼ均等な大

きさの微小な多数の3角パッチにより記述されている。

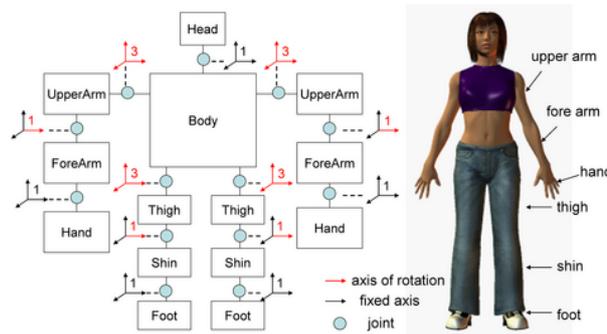


Fig. 1 3D body model

**2.2 2次元距離場の構築と姿勢推定** 提案手法において、2次元距離場  $T$  とは、画面上の人物輪郭線  $I_h$  からの最短距離を画面上の全ての点で求めたものである。画像上の点  $p = (x_p, y_p)$  での  $T(p)$  は以下を満たす。

$$T(p) = 0, p \in I_h \quad (1)$$

$$|\nabla T(p)| = 1 \quad (2)$$

具体的な2次元距離場の構築方法について述べる。まず、カメラ画像から背景差分によって人物シルエット画像を取得し、これにLevel Set Methodを適用して、人物輪郭線を抽出する。次に、抽出された輪郭線からの距離場  $T$  をFast Marching Methodによって構築する。2次元距離場の構築の様子を図2に示す。ただし、本手法では2次元距離場は輪郭線に近い領域 (Distance Band) のみ詳細に構築し、輪郭線から離れた領域では粗に計算することで、構築処理を高速化している。

次に、構築された2次元距離場を用いた3次元人体モデルの位置修正量の計算手順を示す。まず、3次元人体モデルを2次元距離場に投影する。ここで、3次元人体モデルの投影輪郭線上の点  $p_i = (x_{p_i}, y_{p_i})$ 、及びそれに対応した3角パッチを  $u_i$  とする。このとき、パッチ  $u_i$  に加えられる画面に平行な力  $f_i$ 、及び3次元人体モデルの中心に加えられる力  $F$  を以下のように求める。

$$f_{u_i} = \begin{bmatrix} T(p_i)T_x(p_i)/D(p_i) \\ T(p_i)T_y(p_i)/D(p_i) \end{bmatrix} \quad (3)$$

$$D(p_i) = \sqrt{T_x(p_i)^2 + T_y(p_i)^2} \quad (4)$$

$$F = \sum_{u_i} f_{u_i} \quad (5)$$

ただし、 $T_x$ 、 $T_y$  はそれぞれ  $T$  の  $x$ 、 $y$  方向の一次微分値である。求められた  $F$  を3次元人体モデルの中心に加えることで、位置を修正する。

次に、各関節の修正量を求める。ここでは簡単のため、図3に示す腕部分について考える。まず、3次元モデル前腕（手も含む）の投影輪郭線に対応するパッチを  $u_{fore_i}$ 、肘関節から  $u_{fore_i}$  へのベクトルを  $r_{fore_i}$ 、

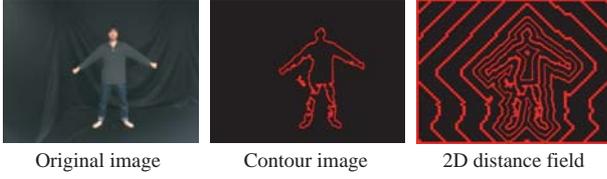


Fig. 2 Construction of 2D distance field

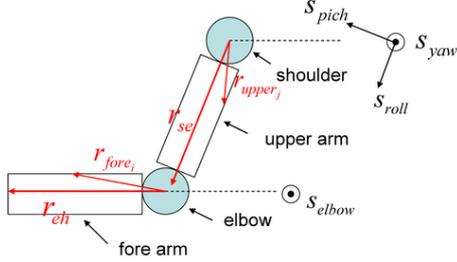


Fig. 3 Arm model

$u_{fore_i}$  に加えられる画面に平行な力を  $f_{fore_i}$  とし、肘関節周りに発生するモーメントを

$$M_{fore} = \sum_{u_{fore_i}} (r_{fore_i} \times f_{fore_i}) \quad (6)$$

として求める．これより、肘関節の回転軸方向を  $s_{elbow}$  とすると、肘関節の修正量は

$$\Delta\phi_{fore} = s_{elbow} \cdot M_{fore} \quad (7)$$

として求まる．

次に、3次元モデル上腕の投影輪郭線に対応するパッチを  $u_{upper_j}$ 、肩関節から  $u_{upper_j}$  へのベクトルを  $r_{upper_j}$ 、 $u_{upper_j}$  にかかる力を  $f_{upper_j}$ 、肩関節から肘関節へのベクトルを  $r_{se}$  とする．これより、肩関節周りに発生するモーメントを

$$M_{upper} = \sum_{u_{upper_j}} (r_{upper_j} \times f_{upper_j}) + \sum_{u_{fore_i}} ((r_{fore_i} + r_{se}) \times f_{fore_i}) \quad (8)$$

として求める．さらに、肩関節の回転軸方向を  $s_{roll}, s_{pitch}, s_{yaw}$  とすると、肩関節の修正量は

$$\Delta\phi_{roll} = s_{roll} \cdot M_{upper} \quad (9)$$

$$\Delta\phi_{pitch} = s_{pitch} \cdot M_{upper} \quad (10)$$

$$\Delta\phi_{yaw} = s_{yaw} \cdot M_{upper} \quad (11)$$

として計算できる．

これまでに提案した手法 [13, 14, 16] は、式 (7), (9) ~ (11) の処理を両腕に適用し、上体の姿勢推定のみを行うものであったが、本論文ではこれを両腕両脚、および胴体姿勢について求めることで、姿勢変化も含む全身のモーショントラッキングを実現する．

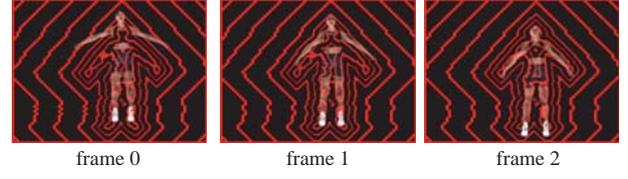


Fig. 4 Examples of motion estimation

**2.3 肌色情報の利用** 手足が胴体などに重なって撮影された場合、本手法では手足の輪郭線を抽出することができず、2次元距離場だけでは姿勢修正量を正しく求めることができない．そこで本手法では、2次元距離場に加えて手足の肌色情報も利用する．

まず、3次元モデルをカメラ画像上に投影し、3次元モデルの手先位置から画像上の肌色位置へのベクトルを  $f_{hand}$  として求め、これを3次元空間に逆投影した  $f'_{hand}$  を求める．次に、3次元モデルの肘関節から手先位置へのベクトルを  $r_{ch}$  とし、肘関節の修正量を以下のように求める．

$$M_{fore2} = r_{ch} \times f'_{hand} \quad (12)$$

$$\Delta\phi_{fore2} = s_{elbow} \cdot M_{fore2} \quad (13)$$

同様に、肩関節の修正量を以下のように求める．

$$M_{upper2} = (r_{se} + r_{ch}) \times f'_{hand} \quad (14)$$

$$\Delta\phi_{roll2} = s_{roll} \cdot M_{upper2} \quad (15)$$

$$\Delta\phi_{pitch2} = s_{pitch} \cdot M_{upper2} \quad (16)$$

$$\Delta\phi_{yaw2} = s_{yaw} \cdot M_{upper2} \quad (17)$$

以上、式 (7), (9) ~ (11), (13), (15) ~ (17) を両腕について求めることで、3次元人体モデルの姿勢を修正する．以上は腕部についての説明であるが、同様の処理を足先および脚部についても行う．2次元距離場と肌色情報を用いた姿勢推定の様子を図4に示す．

**2.4 人物の向き推定** 次に、人物の向きの推定手法について述べる．ただし、本論文では3次元モデルの向きは胴体中心を通る床面に垂直な軸周りの回転角度  $\phi_{body}$  によって決定されるものとする．また本節では人物向きの推定手法として、1) 足先の向きを用いる方法と、2) 輪郭線を用いる方法の2つについて説明し、両手法の比較実験は次章で行う．

**2.4.1 足先の向きからの推定** 画像中の人物の足先の方向から体の向きを推定する手法について述べる．ただし本手法では、足首より下の足先が、前述の肌色抽出処理によりあらかじめ正確に切り出されていると仮定する．

まず、足の肌色領域を囲む最小矩形を求める．次に、その長辺を足の向きとし、両足の向きの合成ベクトルを画像中の人物の向きとして推定する．推定の様子を図5に示す．

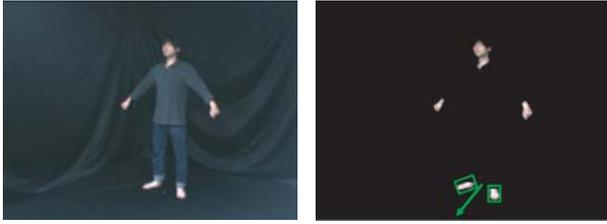


Fig. 5 Examples of body direction estimation

2.4.2 輪郭線からの推定 前項の足先の向きから体の向きを推定する方法では、肌色抽出などにより足先全体が鮮明に切り出されている必要がある。そこで腕部や脚部の姿勢推定と同様に、人物輪郭線と3次元モデルの投影輪郭線を用いて体全体の向きを推定する手法を考える。本手法では、2.2章と同様に2次元距離場から3次元モデルの胴体、上腕、前腕の各3角パッチに加えられる力を求め、その合成モーメントを計算して人物の向きを推定する。

まず、3次元モデルの胴体、上腕、前腕(手も含む)の投影輪郭線に対応するパッチを  $u_k$ 、胴体中心から  $u_k$  へのベクトルを  $r_k$ 、式(3)から求められる  $u_k$  にかかる力を  $f_k$  とし、胴体中心周りに発生するモーメントを

$$M_{body} = \sum_{u_k} (r_k \times f_k) \quad (18)$$

として求める。これより、胴体中心の床面に垂直な回転軸方向を  $s_{body}$  とすると、向きの修正量は

$$\Delta\phi_k = s_{body} \cdot M_{body} \quad (19)$$

として求まる。

2.5 分散カメラを用いたモーショントラッキングシステム 上述した処理は、単一のカメラ画像に対するものであり、これを各カメラで並列処理することで、多数台のカメラを接続した分散カメラシステムへ拡張する。システムの構成を図6に示す。

本システムでは、1台のカメラはそれぞれ別々の  $PC_i$  ( $i = 1 \sim N$ ,  $N$  はカメラ台数) に接続され、各  $PC$  は内部にそれぞれ独立に3次元人体モデルを保持している。また各  $PC$  はインターネットで相互に接続され、これらの  $PC$  とは別に、統合された3次元人体モデルの姿勢推定結果を格納するサーバ  $PC_0$  が接続されている。

処理の手順を以下に示す。まず各カメラで独立、非同期に上述の処理を行い、それぞれのカメラ座標における関節の姿勢推定量を計算する。ただし、各カメラに画像が入力された直後のそれぞれの  $PC_i$  における3次元人体モデルの初期姿勢は、その瞬間のサーバ  $PC_0$  に格納されている3次元人体モデルの姿勢を用いる。次に各  $PC_i$  で得られた姿勢推定量をサーバ  $PC_0$  に非同期で送り、サーバの3次元人体モデルの姿勢を変更する。この非同期並列処理により、カメラの台数が増減しても、1台あたりの姿勢推定のための計算量は変化しないため、本手法は分散カメラシステムに適した手法といえる。

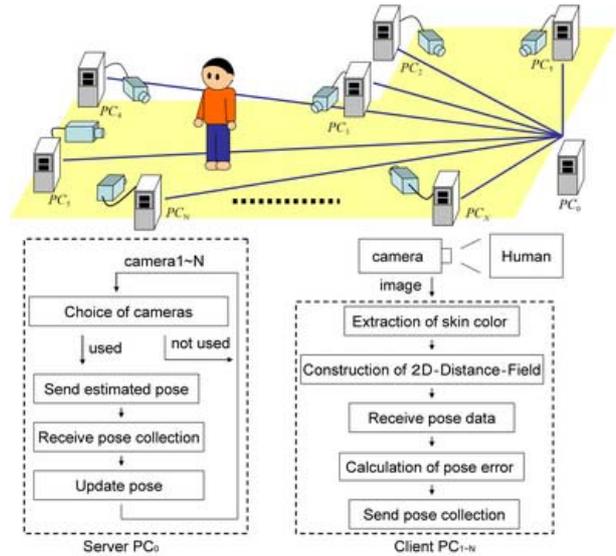


Fig. 6 Distributed camera system

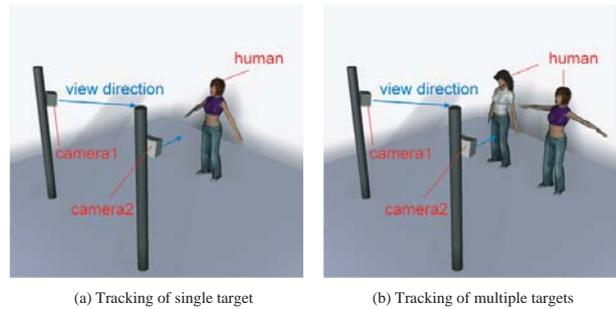


Fig. 7 Experimental setup

### 3. トラッキング実験

3.1 多視点画像によるリアルタイム・モーショントラッキング まず2台のカメラを用いた動作推定実験を行った。実験で使用したカメラはDragonfly2(Point Grey Research)であり、画像の取得から動作推定まで全てオンラインで行った。画像の大きさが  $320 \times 240$  の時の処理時間は、2次元距離場の構築が  $8.7\text{ms}$ 、姿勢修正量計算が  $3.0\text{ms}$  である。実験環境を図7(a)に、椅子に座る動作に対する推定結果を図8に示す。

3.2 複数人物動作の同時推定 次に2人の人物動作を同時に推定する実験を行った。実験環境を図7(b)に、実験結果を図9に示す。これより、2人の人物に対しても、1人の場合と同様に動作推定が可能であることを確認した。なお、処理時間は、2次元距離場の構築が  $6.5\text{ms}$ 、姿勢修正量計算が  $4.2\text{ms}$  である。実験環境が違いため単純な比較はできないが、対象が1人の場合と比較して処理時間に大きな差が無いことがわかる。提案手法は、一度2次元距離場を構築すれば、追跡する輪郭数が増えた場合でも、姿勢修正量の計算時間はほとんど変化しないという特徴があり、複数人物動作の同時推定に特に適していると考えられる。

3.3 向きの推定実験 これまでの実験では、対象が方向転換しないと仮定し、人物の向きの推定は行っ

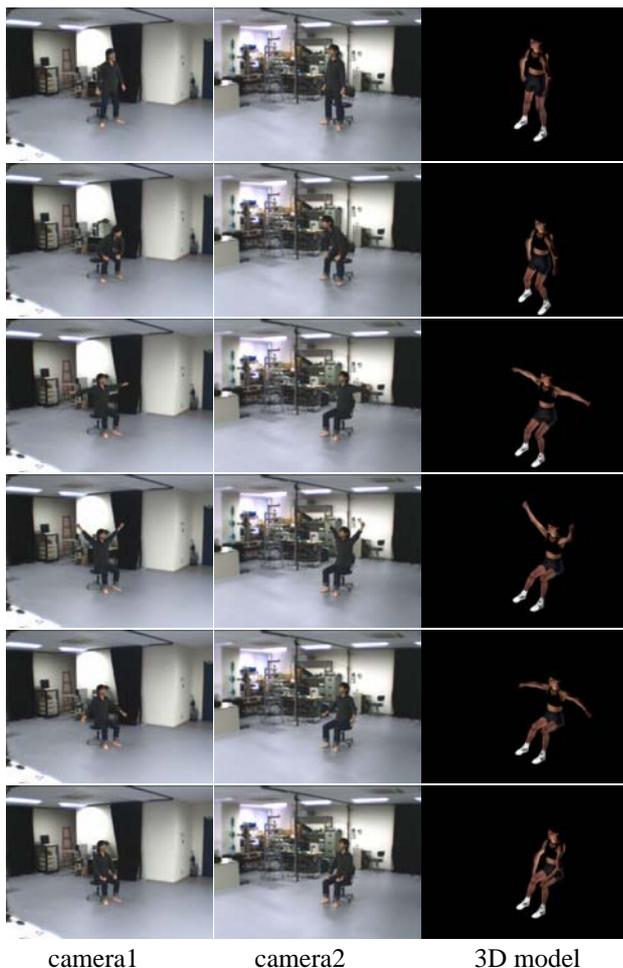


Fig. 8 Tracking results for sitting motion by two cameras

ていない．そこで，方向転換を含む歩行動作に対する人物方向の推定実験を行った．実験環境は図 7(a) と同様に，人物 1 人に対してカメラ 2 台を配置した．足の向きを用いた推定結果を図 10 に，輪郭線を用いた推定結果を図 11 に示す．ただし，足先の白い直線は推定された胴体の向きを示している．これより，両手法とも静止，移動時の方向転換に応じて，胴体の向きの推定が可能であることがわかる．しかし，足の向きを用いた推定は，足先領域が正確に抽出される必要があり，実用上は問題が多いと考えられる．

3-4 分散カメラシステムを用いた動作推定実験環境内に分散配置された 8 台のカメラを用い，人物の位置，姿勢に応じて動作推定に適したカメラを自動的に切り替えて人物の動作推定を行うシステムを開発し，実験を行った．

実験環境を図 12 に，動作推定結果を図 13 に示す．ただし実験では，動作推定に適したカメラとして，推定された人物位置から 5 m 以内のカメラを選択した．また，足先からの人物向きの推定に関しては，安定した肌色抽出が可能ないように人物の正面から  $\pm 90$  度以内に設置されたカメラを用いている．なお， $PC_i$  の処理時間は，2 次元距離場の構築が 10ms，姿勢修正量

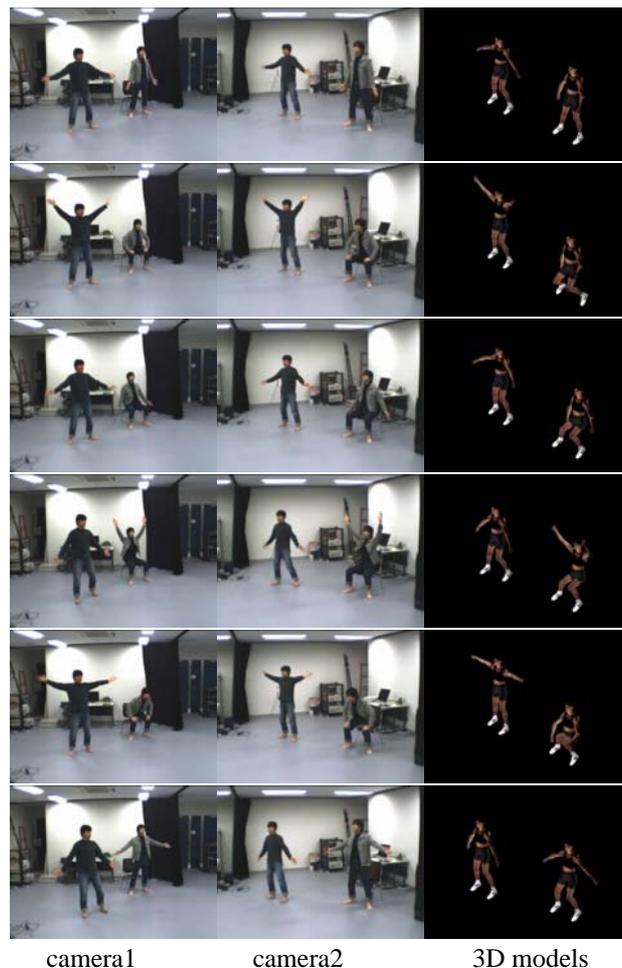


Fig. 9 Tracking results of two people by two cameras

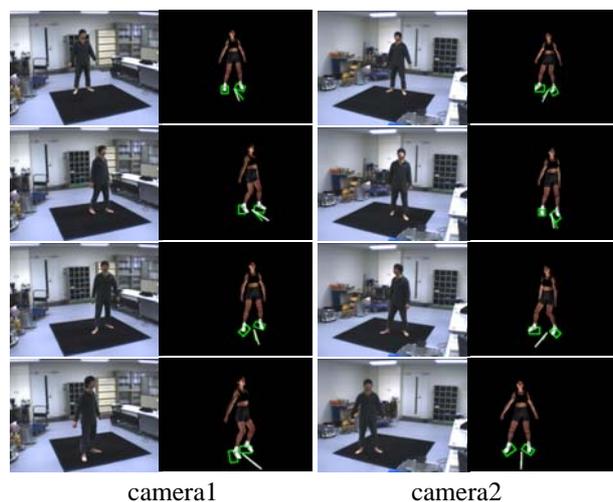


Fig. 10 Estimation of body direction by foot direction

計算が 7ms であり， $PC_0$  の 1 周期の処理時間は 125ms である．図 13 に示すように，本システムによる人物の位置，姿勢に応じて自動的に適切なカメラに切り替え，次々に動作推定を行っていることがわかる．人物はまず反時計回りに半周して椅子に座り，さらに反時計回りに半周し，向きを変えて時計回りに半周して最

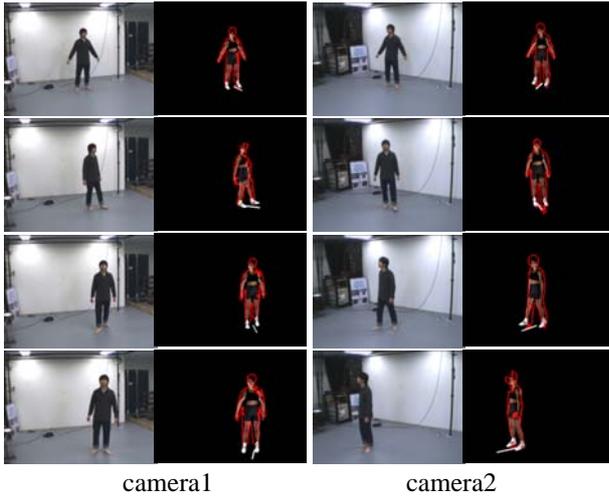


Fig. 11 Estimation of body direction by contour lines

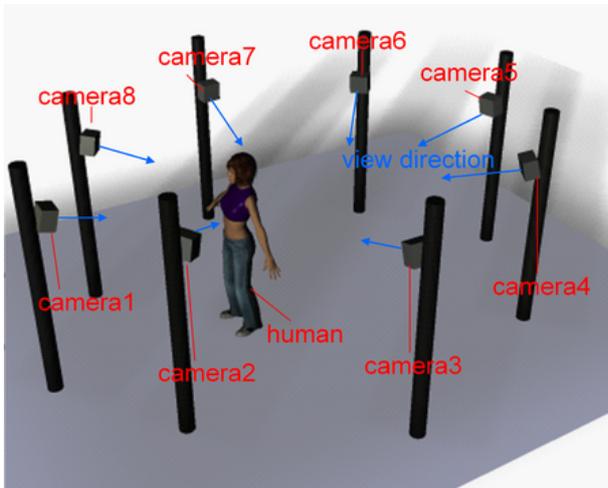


Fig. 12 Motion tracking using distributed camera system

後に椅子の上に立つといった動作を行っている。

#### 4. まとめ

本論文では、輪郭線ベースの粗密2次元距離場を用いたリアルタイム・モーショントラッキング法を提案した。また実験により、歩行動作や椅子に座る動作などが推定可能であること、複数人の動作推定にも適用可能であること、また肌色情報及び2次元距離場を用いることにより、人物の向きの推定が可能であることを確認した。さらに環境内に分散配置された多数台のカメラを用い、人物の位置、姿勢に応じて動作推定に適したカメラを次々と切り替えることで、自己遮蔽や自己運動にも頑強なシステムを構築した。今後の課題として、現在使用している3次元人体モデルでは、椅子に座るときの腰の屈伸など細かい動作を再現できない場合があり、人体モデルの胴体の自由度を増やすことが挙げられる。

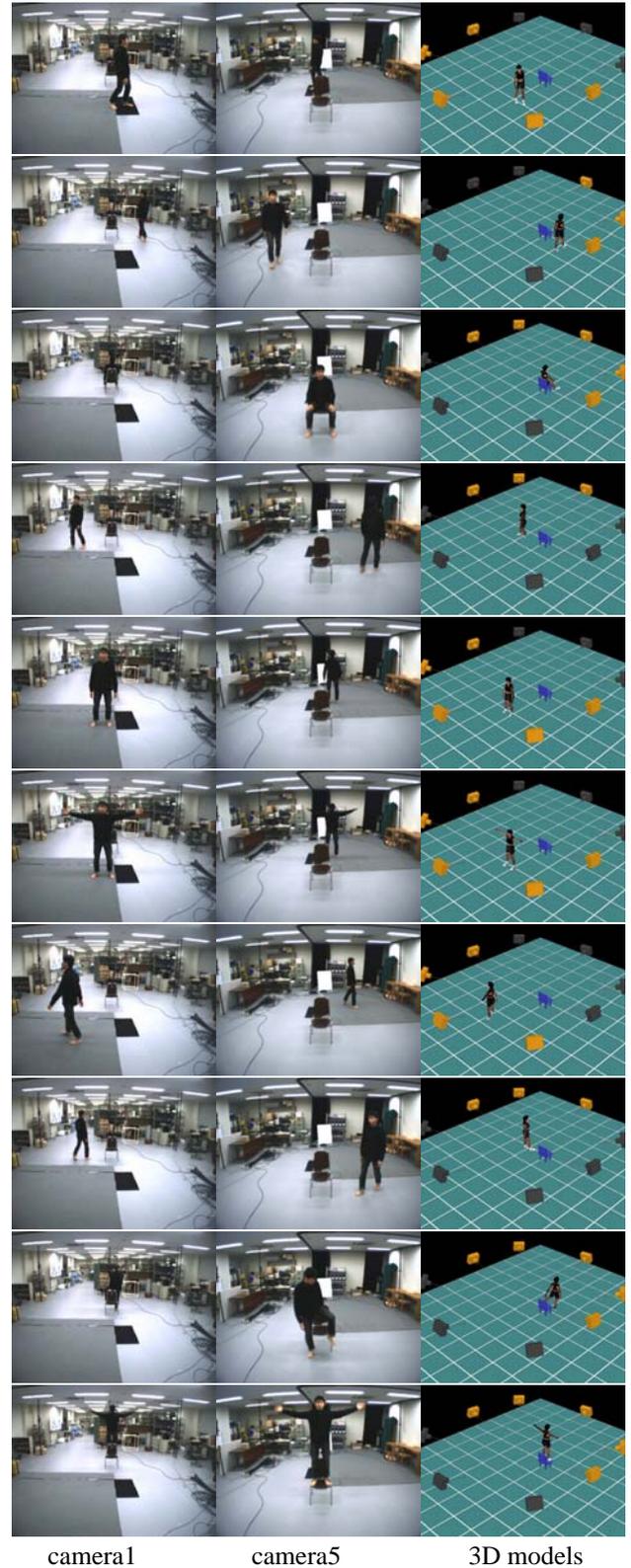


Fig. 13 Motion tracking experiments using distributed camera system

#### A Fast Marching Method

本論文で2次元距離場の構築に用いられるFast Marching Methodは、移動体追跡や3次元幾何モデル

ングなどの様々な分野で用いられる Level Set Method の高速化手法として Sethian らによって提案された手法 [17, 19] である。

Fast Marching Method は, Eikonal 方程式 ( $|\nabla T(p)| F = 1$ ) の高速な数値解法として提案された。ただし  $T(p)$  は境界が点  $p$  に到達した時刻である。この方程式は通常, 収束計算により解かれるため, 膨大な計算時間が必要である。しかし, Fast Marching Method では, 成長速度  $F$  の符号が固定という条件を加え, 到達時刻の小さい点から大きい点へ一方向に到達時刻を確定していくことで, 収束計算を行うことなく高速に Eikonal 方程式の解を導くことができる。

Fast Marching Method では, まず Eikonal 方程式を次の差分方程式に置き換える。

$$\begin{aligned} & (\max(D_{ij}^{-x}T, -D_{ij}^{+x}T, 0)^2 + \\ & \max(D_{ij}^{-y}T, -D_{ij}^{+y}T, 0)^2)^{\frac{1}{2}} = 1/F_{ij} \quad (20) \end{aligned}$$

次に, 境界は到達時間が小さい場所から大きい場所へと一方向に伝播することから, 到達時間の小さい場所から順に式 (20) を解き, 各点の到達時間を確定する。

具体的には以下ようになる。

**Step.1** (初期化) まず, 次の処理によって全てのグリッドを 3 つのリスト (*known*, *trial*, *far*) のいずれかに追加する。

1. 境界に属するグリッドを *known* のリストに追加し, 到達時間を 0 にセットする ( $T=0$ ) .
2. *known* の 4 近傍のグリッドのうち, *known* でないグリッドを *trial* に追加し, そのグリッドの到達時間を  $T_{ij} = 1/F_{ij}$  により計算して仮登録する。また, これらのグリッドを到達時間に関する昇順の Heap 形式で保存する。
3. 上記以外のピクセルを *far* に追加し, 到達時間を無限大とする ( $T=\infty$ ) .

**Step.2** Heap の先頭に置かれた, *trial* のリスト中で到達時間  $T$  が最も小さいグリッド ( $i_{min}, j_{min}$ ) を選択し, そのグリッドを *trial* のリストと Heap から除外し, *known* のリストに追加する。Heap から除外する際, 同時に Heap を再構築 (downheap) する。

**Step.3** 現在選択されたグリッド ( $i_{min}, j_{min}$ ) の 4 近傍 ( $(i_{min}-1, j_{min}), (i_{min}+1, j_{min}), (i_{min}, j_{min}-1), (i_{min}, j_{min}+1)$ ) のうち, *far* のリストに属しているグリッドを *trial* のリストに追加する。

**Step.4** 現在選択されたグリッド ( $i_{min}, j_{min}$ ) の 4 近傍のうち, *trial* に属している近傍点の到達時間を式 (20) を用いて計算して仮登録し, Heap を再構築 (upheap) する。

**Step.5** *trial* に属しているグリッドが存在すれば Step.2 へ戻る。それ以外の時は処理を終了する。

この Fast Marching Method を用いると, ある与えられた境界からある地点までの距離を高速に計算するこ

とができ, これにより高速に境界からの距離場を構築できる。これは, まず式 (20) において成長速度  $F_{ij}$  を全空間で 1 とし, 上記の手順によって全領域での境界の到達時刻  $T$  を求める。ここで, 成長速度  $F$  が 1 であることから, 求められた  $T$  は境界からの距離に等しくなる。

## 謝辞

本研究の一部は, 文部科学省科研費補助金基盤 B 「ロボット群と環境固定センサ群の相互支援による人間・ロボット共生環境の情報構造化 (18360124)」および「5 感センサネットワークによる被災建造物内 RT システムの開発 (19360119)」の助成を受けて行われました。

## 文 献

- (1) Tsutomu Hasegawa. Robot Town Project : Implementation of a common platform. *The 25th Annual Conference of the Robotics Society of Japan* 1136 , 2007
- (2) Masanobu Yamamoto, Akitsugu Sato, Satoshi Kawada, Takuya Kondo, and Yoshihiko Osaki. Incremental tracking of human actions from multiple views,. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'98)*, 1998.
- (3) Z. Chen and H. J. Lee. Knowledge-guided visual perception of 3-d human gait from a single image sequence. *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 22, No. 2, pp. 336–342, 1992.
- (4) C. Orrite-Urunuela, J.M. del Rincon, J.E. Herrero-Jaraba, and G. Rogez. 2d silhouette and 3d skeletal models for human detection and tracking. In *Proceedings of the 17th International Conference on Pattern Recognition*, pp. 244–247, 2004.
- (5) L. Campbell and A. Bobick. Recognition of human body motion using phase space constraints. In *Proceedings of IEEE International Conference on Computer Vision*, pp. 624–630, 1995.
- (6) D. M. Gavrila and L. S. Davis. Towards 3-d model-based tracking and recognition of human movement: a multi-view approach. In *International Conference on Automatic Face and Gesture Recognition*, pp. 272–277, 1995.
- (7) Quentin Delamarre and Olivier Faugeras. 3d articulated models and multi-view tracking with silhouettes,. In *Proc. IEEE International Conference on Computer Vision*, pp. pp.716–721, 1999.
- (8) J. Deutscher, A. Blake, and I.Reid. Articulated body motion capture by annealed particle filtering. In *Proceedings of Computer Vision and Pattern Recognition*, pp. 2126–2133, 2000.
- (9) H. Sidenbladh, M. J. Black, and D. Fleet. Stochastic tracking of 3d human figures using 2d image motion. In *Proceedings of European Conference on Computer Vision*, pp. 702–718, 2000.
- (10) J. Carranza, C. Theobalt, M. Magnor, and H.P. Seidel. Free-viewpoint of human actors. In *Proceedings of SIGGRAPH 200*, pp. 569–577, 2003.
- (11) C. Sminchisescu and B. Triggs. Kinematic Jump Processes for Monocular 3D Human Tracking, In

*Proceedings of International Conference on Computer Vision and Pattern Recognition*, 2004.

- (12) C. Taylor. Reconstruction of Articulated Objects from Point Correspondences in a Single Uncalibrated Image, In *Proceedings of International Conference on Computer Vision and Pattern Recognition*, 2000.
- (13) Ryo Kurazume, Hiroaki Omasa, Seiichi Uchida, Rinichiro Taniguchi, and Tsutomu Hasegawa. Embodied proactive human interface "pico-2". In *International Conference on Pattern Recognition*, pp. B04–0206, 2006.
- (14) Ryo Kurazume, Seiichi Uchida, Rin-ichiro Taniguchi, Tsutomu Hasegawa. Embodied Proactive Human Interface "PICO-2 ". In *Proc. 9th Robotics Symposia*, p. RS036, 2006.
- (15) C. Sminchisescu and A. Telea. Human pose estimation from silhouettes: A consistent approach using distance level sets. In *WSCG International Conference on Computer Graphics, Visualization and Computer Vision*, 2002.
- (16) Y. Iwashita, R. Kurazume, K. Konishi, M. Nakamoto, M. Hashizume, and T. Hasegawa. Fast 2d-3d registration for navigation system of surgical robot. In *Proc. IEEE International Conference on on Robotics and Automation*, pp. pp.909–915, 2005.
- (17) J. Sethian. Level Set Methods and Fast Marching Methods, second edition. *Cambridge University Press*, UK, 1999.
- (18) Yumi Iwashita, Ryo Kurazume, Tokuo Tsuji, Kenji Hara, and Tsutomu. Fast implementation of level set method and its realtime applications. In *Proc. IEEE International Conference on Systems, Man and Cybernetics 2004*, pp. pp.6302–6307, 2004.
- (19) J. Sethian. A fast marching level set method for monotonically advancing fronts. In *Proceedings of the National Academy of Science*, Vol. 93, pp. 1591–1595, 1996.