

# カスケード姿勢推定に基づく 距離・画像センサを用いたジェスチャ認識

加藤 秀章<sup>1</sup> 小林 宏章<sup>1</sup> 山根 亮<sup>1</sup> 尺長 健<sup>1</sup>

概要：従来、我々は距離・画像センサデータを用いた人物姿勢追跡法およびジェスチャ認識を提案している。本稿では、14 自由度で表現した人体の上半身モデルの姿勢推定を行う上で、胴体と腕部の重なりを考慮した追跡を行う。また、この方法により得られる人物姿勢を時々刻々処理し、各フレームの姿勢類似度からジェスチャ認識を行う。シミュレーション実験および複数人の実動画に適用した結果を報告し、提案手法の有効性を示す。

## 1. はじめに

ジェスチャ認識は、人間と計算機の間での自然なインタフェースとして注目されている [1][2]。画像情報を用いた関節物体の追跡は古くから試みられ、尺長 [3] は線分追跡により姿勢追跡が原理的に可能であることを示している。また、佐竹・尺長 [4] は関節構造を簡単化した人物追跡系を提案しており、歩行人物追跡においては有効性が確認されている。一方、距離情報を用いた動作認識 [5] は距離センサの性能が低かったために研究があまり進展しなかった。しかしながら、近年高性能の距離・画像センサが容易に利用できるようになったのに伴い、画像情報と距離情報を併用することで、容易に人物領域を抽出でき、人物追跡の精度が向上できることから、従来と比較して正確なジェスチャ認識が行えると考えられる。四宮ら [6] は距離・画像センサを用いた追跡系を構成し、人物の上半身を 14 自由度の姿勢空間で取扱うことで 3 次元疎テンプレート追跡に基づく姿勢追跡を提案している。小林ら [7] はこの姿勢追跡により得られる時系列姿勢情報を用いて胴体の回転に不変なジェスチャ認識法を提案している。本稿では、これらの研究で用いられてきた姿勢追跡およびジェスチャ認識を改良し、高精度な姿勢追跡およびジェスチャ認識を目指した。本稿で取り上げる改良点を以下に列挙する。

### (1) 胴体から姿勢を推定するカスケード姿勢推定の改良

従来の胴体 → 上腕 → 下腕の  $6 + 2 + 2$  次元追跡としていたものを、上腕と下腕の姿勢を同時推定する方法 ( $6 + 4$  次元追跡) に変更。

### (2) オクルージョン処理を追加

従来系では胴体と腕部の重なりを十分には排除できていなかったため、姿勢推定が不安定になっていた。本稿ではマスク処理により、この問題の解決を図る。

### (3) ジェスチャ類似度の改良

開始フレーム検出後、フレームごとの姿勢類似度を求めていき、これを累積することにより、最終的なジェスチャ類似度を算出するように変更する。

以下、第 2 節では上半身の関節物体の取扱いについて説明し、第 3 節では 3 次元姿勢追跡系の概要を述べる。第 4 節では複素相関を用いた姿勢類似度およびジェスチャ類似度の定義を示す。また、姿勢追跡により得られたパラメータからジェスチャを認識する方法について述べる。第 5 節ではシミュレーション実験および実動画実験の結果を述べる。

## 2. 関節物体モデル

本稿で追跡対象とするのは、人間の上半身である。上半身は胴体、左上腕、左下腕、右上腕、右下腕の 5 つの剛体が左肩、左肘、右肩、右肘の 4 つの関節によって連結されているものとする。追跡対象である 3 次元関節物体をサーフィスモデルで取り扱う。サーフィスモデルでは、物体の表面を小さなパッチの集合として扱う。

### 2.1 座標系の定義

一般に、関節物体の姿勢は、主となる物体の姿勢と、それぞれの関節位置、関節角度により定義される。本稿では、上半身モデルのレンダリングや姿勢推定問題を考えるため、以下に示す 4 つの (同次) 座標系を考える。

(1)  $\widetilde{M}_C$  : カメラ (中心) 座標系

<sup>1</sup> 岡山大学  
Okayama University

(2)  $\widetilde{M}_O$ : 物体 O(胴体中心) 座標系

(3)  $\widetilde{M}_U$ : 物体 U(上腕中心) 座標系

(4)  $\widetilde{M}_L$ : 物体 L(下腕中心) 座標系

物体モデルの表面を構成する点を胴体座標系, 上腕座標系, 下腕座標系の 3 つの座標系を用いて記述し, 画像生成過程ではカメラ座標に変換する.

## 2.2 各剛体の表現と姿勢行列

胴体については, 6 次元姿勢 (3 次元並進 + 3 次元回転) を姿勢行列で取り扱う. 胴体座標系からカメラ座標系への変換  $D_{OC}$  を胴体の姿勢行列と考えることができる.

$$D_{OC} = \begin{bmatrix} R_{OC} & t_{OC} \\ \mathbf{0}^\top & 1 \end{bmatrix} \quad (1)$$

$$\widetilde{M}_C = D_{OC} \widetilde{M}_O \quad (2)$$

ここで式 (1) における  $R_{OC}$  は胴体の回転を表す行列であり,  $t_{OC}$  は胴体の並進を表すベクトルである.

上腕の姿勢は, 胴体の姿勢  $D_{OC}$  と肩を中心とした回転の合成により得られる. 上腕座標系から胴体座標系への変換を  $D_{UO}$  と表記する. 胴体座標系における肩関節の位置を  $t_S$  とする. ここで, 肩関節は上腕座標系の原点にあるものとする. また, 肩での回転 (2 自由度) を表す回転行列を  $R_S$  とすると, 上腕座標で表される点のカメラ座標系への変換は以下の式 (4) で表せる.

$$D_{UC} = D_{OC} D_{UO} \quad (3)$$

$$\widetilde{M}_C = D_{UC} \widetilde{M}_U \quad (4)$$

ここで,

$$D_{UO} = \begin{bmatrix} R_S & t_S \\ \mathbf{0}^\top & 1 \end{bmatrix} \quad (5)$$

$$R_S = \begin{bmatrix} c_{\theta_S} & -s_{\theta_S} & 0 \\ s_{\theta_S} & c_{\theta_S} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_{\theta_S} & 0 & s_{\theta_S} \\ 0 & 1 & 0 \\ -s_{\theta_S} & 0 & c_{\theta_S} \end{bmatrix} \quad (6)$$

なお,  $c_{\theta_S} = \cos \theta_S$ ,  $s_{\theta_S} = \sin \theta_S$  とする.

下腕の姿勢は, 上腕の姿勢  $D_{UC}$  と, 肘を中心とした回転の合成により得られる. 下腕座標系から上腕座標系への変換を  $D_{LU}$  と表記する. 上腕座標系における肘の位置を  $t_E$  とする. ここで, 肘関節は下腕座標系の原点にあるものとする. さらに肘での回転 (2 自由度) を表す回転行列を  $R_E$  とすると, 下腕の点をカメラ座標系へ変換する式 (8) は以下のように表せる.

$$D_{LC} = D_{UC} D_{LU} = D_{OC} D_{UO} D_{LU} \quad (7)$$

$$\widetilde{M}_C = D_{LC} \widetilde{M}_L \quad (8)$$

ここで,

$$D_{LU} = \begin{bmatrix} R_E & t_E \\ \mathbf{0}^\top & 1 \end{bmatrix} \quad (9)$$

$$R_E = \begin{bmatrix} c_{\theta_E} & -s_{\theta_E} & 0 \\ s_{\theta_E} & c_{\theta_E} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_{\theta_E} & 0 & s_{\theta_E} \\ 0 & 1 & 0 \\ -s_{\theta_E} & 0 & c_{\theta_E} \end{bmatrix} \quad (10)$$

なお,  $c_{\theta_E} = \cos \theta_E$ ,  $s_{\theta_E} = \sin \theta_E$  とする.

## 3. 関節物体の 3 次元姿勢追跡

本章では, 3 次元疎テンプレートによる姿勢追跡について述べる. 人体部位をカスケード姿勢推定によって追跡する. その際に, 上腕と下腕の姿勢を同時に推定し, 4 (= 2 + 2) 次元の追跡を行う. さらに, 胴体と腕による重なり部分を抽出し, マスク処理を追加した追跡を行う.

### 3.1 3 次元疎テンプレートマッチング

既知形状物体の 3 次元姿勢追跡は, 直前の物体の見えから作成された疎テンプレートによる追跡に基づいて行う [8]. テンプレート画像  $\hat{I}(u, v)$  から疎テンプレート  $\{(u_j, v_j)\}$  の各点  $(u_j, v_j)$  に対応する 3 次元座標を  $M_j$  とすると, 3 次元疎テンプレートは次のように記述できる.

$$\{(M_j; \hat{I}_j)\} = \{(X_j Y_j Z_j; \hat{I}_j)\} \quad (11)$$

ただし,  $X_j = X(u_j, v_j)$ ,  $Y_j = Y(u_j, v_j)$ ,  $Z_j = Z(u_j, v_j)$ ,  $\hat{I}_j = \hat{I}(u_j, v_j)$  である. 本稿では, 粗密追跡法 [8] を用いることでサンプル数の削減を図る. ガウスノイズの標準偏差を段階的に減らすことで, 効率よく姿勢空間を探索する.

### 3.2 カスケード姿勢推定

本稿では, 胴体, 右腕, 左腕の順に姿勢を推定するカスケード姿勢推定を提案している. 四宮ら [6] は腕部の追跡において, 上腕の 2 自由度, 下腕の 2 自由度を順に推定する 2 + 2 次元の姿勢推定を行っていた. この 2 + 2 次元姿勢推定を変更し, 上腕・下腕の姿勢を同時に推定する 4 次元の姿勢推定を提案する.

まず, 胴体の姿勢 (6 自由度) の推定を行い, 胴体姿勢を推定する. 現時刻の胴体推定姿勢と, 前時刻の腕部推定姿勢を基準姿勢とし, 右腕の姿勢 (4 自由度), 左腕の姿勢 (4 自由度) それぞれについて, 姿勢サンプルを生成し姿勢を推定する. ここで, 胴体, 上腕, 下腕それぞれの 3 次元疎テンプレートを構成する  $j$  番目の点  $M_j$  の画像上の位置  $m_{jk}$  は, 式 (12) により算出される. 各部位において括弧内を定数として取り扱うことができるため,  $k$  番目のサンプル姿勢に対応する姿勢変換行列  $\delta D_k, \delta D'_k, \delta D''_k$  を用いて, 画像座標  $m_{jk}$  が計算できる.

$$\begin{cases} m_{jk} = (A) \delta D_k (D_{OC} \widetilde{M}_j) \\ m_{jk} = (A D_{OC}) \delta D'_k (D_{UO} \widetilde{M}_j) \\ m_{jk} = (A D_{OC}) \delta D'_k (D_{UO}) \delta D''_k (D_{LU} \widetilde{M}_j) \end{cases} \quad (12)$$

同様に、距離  $Z_{jk}$  は式 (13) によって求められる。

$$\begin{cases} (X_{jk}Y_{jk}Z_{jk}1)^\top = \delta D_k(D_{OC}\tilde{M}_j) \\ (X_{jk}Y_{jk}Z_{jk}1)^\top = (D_{OC})\delta D'_k(D_{UO}\tilde{M}_j) \\ (X_{jk}Y_{jk}Z_{jk}1)^\top = (D_{OC})\delta D'_k(D_{UO})\delta D''_k(D_{LU}\tilde{M}_j) \end{cases} \quad (13)$$

ここで、 $\tilde{M}_j$  は  $M_j$  の同次座標を示す。  $\delta D_k$ ,  $\delta D'_k$ ,  $\delta D''_k$  は、サンプル姿勢の 14 パラメータ  $[\alpha_1 \ \alpha_2 \ \dots \ \alpha_{14}] = [\delta\theta_0 \ \delta\phi_0 \ \delta\psi_0 \ \delta t_1 \ \delta t_2 \ \delta t_3 \ \delta\phi_{rS} \ \delta\theta_{rS} \ \delta\phi_{rE} \ \delta\theta_{rE} \ \delta\phi_{lS} \ \delta\theta_{lS} \ \delta\phi_{lE} \ \delta\theta_{lE}]$  を式 (1), (5), (9) に代入して求める。

### 3.3 距離・画像センサを用いた追跡

本追跡では 3 次元テンプレートを用いるため、画像情報と同時に距離情報を利用できる。距離センサから得られた距離情報と、3 次元テンプレートの各サンプルにおける推定位置を比較することで、画像情報のみを用いた追跡よりも高精度な追跡が期待できる。

#### 3.3.1 距離情報を利用したマスク処理

距離情報を用いることで背景領域の分離が可能となる。 $d_0 \sim d_1$ [mm] の範囲内で人物が観測されるならば、各画素  $I(u, v)$  に対応する距離を  $d(u, v)$  とすると、人物マスク  $m_1(u, v)$  は式 (14) により表される。

$$m_1(u, v) = \begin{cases} 1 & d_0 \leq d(u, v) \leq d_1 \\ 0 & \text{otherwise} \end{cases} \quad (14)$$

胴体姿勢推定では式 (14) で与えられるマスクを用いる。一方、腕部姿勢追跡では、現時刻の胴体姿勢がすでに推定されているため、入力距離情報と胴体推定位置との差が大きい領域を抽出し、腕部追跡では不必要な胴体領域にマスクをかける。このため、まず現時刻の距離と胴体推定位置の差（絶対値）の平均  $\bar{\epsilon}$  を式 (15) より求める。

$$\bar{\epsilon} = \frac{1}{I_t} \sum_{(u,v) \in T_t} |d_t(u, v) - Z_t(u, v)| \quad (15)$$

ここで、 $T_t$  は現時刻における胴体推定領域、 $I_t$  は領域  $T_t$  の大きさ、 $d_t(u, v)$  は現時刻の距離情報、 $Z_t(u, v)$  は現時刻の胴体推定距離である。式 (15) の  $\bar{\epsilon}$  から、腕部マスク  $a(u, v)$  を式 (16) により作成する。

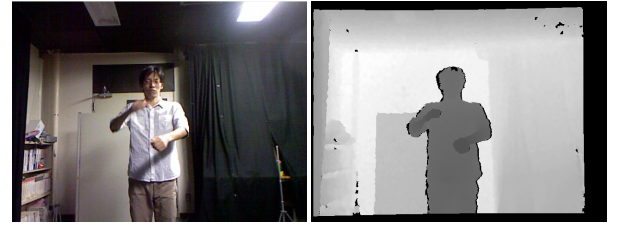
$$a(u, v) = \begin{cases} 1 & d_t(u, v) - Z_t(u, v) < -k\bar{\epsilon} \\ 0 & \text{otherwise} \end{cases} \quad (16)$$

$k$  は定数である。事前実験により  $k = 3.0$  とする。式 (14) で与えられる人物マスク  $m_1(u, v)$  と式 (16) で与えられる腕部マスク  $a(u, v)$  を用いて、腕部追跡で用いるマスク  $m_2(u, v)$  を式 (17) にて作成する。

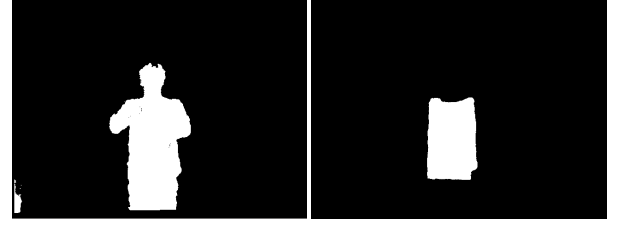
$$m_2(u, v) = m_1(u, v)(1 - t(u, v)) + a(u, v) \quad (17)$$

ここで、 $t(u, v)$  は  $T_t$  に対応する関数であり、

$$t(u, v) = \begin{cases} 1 & (u, v) \in T_t \\ 0 & \text{otherwise} \end{cases}$$



画像情報と距離情報



人物マスク  $m_1(u, v)$

胴体推定マスク  $t(u, v)$



腕部マスク  $a(u, v)$

腕部追跡のマスク  $m_2(u, v)$

図 1 入力画像とマスク領域

とする。式 (14), (17) の処理を行い作成したマスクを図 1 に示す。

#### 3.3.2 画像情報を用いたテンプレートマッチング

3.3.1 で作成したマスク  $m(u, v)$ （胴体追跡では  $m_1(u, v)$ 、腕部追跡では  $m_2(u, v)$ ）を用いて入力画像  $I(u, v)$  でテンプレートマッチングを行い、サンプル姿勢  $k$  における画像情報の評価値  $\epsilon_k$  を式 (18) にて求める。

$$\epsilon_k = \frac{1}{\sum_{j=1}^n \rho(\beta I(u_{jk}, v_{jk})m(u_{jk}, v_{jk}) - \alpha \hat{I}_j)} \quad (18)$$

ただし、

$$\alpha = \frac{1}{\sum_{j=1}^n \hat{I}_j}, \quad \beta = \frac{\alpha \sum_{j=1}^n \hat{I}_j^2}{\sum_{j=1}^n \hat{I}_j I(u_{jk}, v_{jk})},$$

$$\rho(x) = \frac{x^2}{c^2 + x^2}$$

なお、 $\alpha$  はテンプレート画像の正規化係数、 $\beta$  は入力画像の正規化係数、 $\rho(x)$  は Geman-McClure 関数、 $n$  はテンプレート点数である。また  $c$  は定数であり、ここでは、 $c = 0.5/n$  とする。

#### 3.3.3 距離情報を利用したテンプレートマッチング

サンプル姿勢  $k$  における入力距離情報の評価値  $\epsilon'_k$  は、テンプレートの  $j$  番目の点の距離情報  $Z_{jk}$  と入力距離情報  $d$  を用いて式 (19) にて計算される。マスク  $m(u, v)$  は 3.3.2 と同様である。

$$\epsilon'_k = \frac{1}{\sum_{j=1}^n \rho'(d(u_{jk}, v_{jk})m(u_{jk}, v_{jk}) - Z_{jk})} \quad (19)$$

ただし,

$$\rho'(x) = \frac{x^2}{c'^2 + x^2}$$

$\rho'(x)$  は Geman-McClure 関数である. ここで,  $c'$  は定数であり, 適切な値を設定することでノイズの影響を低減することができる. 距離・画像センサの出力が mm を単位とする整数値であることを勘案し, 予備実験により,  $c' = 5.0$  とした.

### 3.3.4 画像情報と距離情報を併用した評価値の算出

画像情報に対して算出した評価値  $\epsilon_k$  と距離情報に対して算出した評価値  $\epsilon'_k$  を用いて姿勢の推定を行うことで, より適切な姿勢を決定できると考えられる.

あるサンプル姿勢における評価値  $\epsilon$  を両者の重み付き和によって式 (20) で定義する.

$$\epsilon = (1 - \lambda)\epsilon_k + \lambda\epsilon'_k \quad (20)$$

腕部のサンプル姿勢の場合は, 式 (20) から上腕と下腕それぞれの評価値  $\epsilon_u, \epsilon_l$  を求め, 式 (21) により腕部のサンプル姿勢の評価値を求める.

$$\epsilon_{arm} = \sqrt{\epsilon_u^2 + \epsilon_l^2} \quad (21)$$

## 4. 姿勢追跡を用いたジェスチャ認識

本章では, 姿勢情報からジェスチャ認識を行う手法について述べる. 各ジェスチャをあらかじめ複数回撮影し, 各追跡結果として得られた姿勢の時系列平均を登録ジェスチャとする. 登録ジェスチャと入力姿勢情報とを現時刻で評価し, 入力動画画像中の人物が登録されているジェスチャのうちどのジェスチャを行っているかを認識する.

### 4.1 登録ジェスチャ

ジェスチャの登録には, 胴体を除いた両腕の姿勢 (肩, 肘を結んだ 3 次元ベクトルと肘, 手を結んだ 3 次元ベクトル) を用いる. 姿勢  $\mathbf{P}$  は, 右上腕, 右下腕, 左上腕, 左下腕の 3 次元ベクトルの単位ベクトル  $\mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3, \mathbf{V}_4$  を用いて以下のように表す.

$$\mathbf{P} = [\mathbf{V}_1^T \ \mathbf{V}_2^T \ \mathbf{V}_3^T \ \mathbf{V}_4^T]^T \quad (22)$$

第  $g$  登録ジェスチャを  $\mathbf{S}_g$  と表す. 登録ジェスチャの数を  $G$  とする. また, 第  $g$  登録ジェスチャの第  $f$  ( $f = 1, 2, \dots, F$ ) フレームを  $\mathbf{S}_{gf}$  で表す. 本稿では, 登録ジェスチャのフレームは  $F = 30$  とする. 登録ジェスチャは姿勢  $\mathbf{P}$  を用いて以下のように表す.

$$\mathbf{S}_g = [\mathbf{P}_{g1}^T \ \mathbf{P}_{g2}^T \ \dots \ \mathbf{P}_{gF}^T]^T \quad (23)$$

ここで,  $\mathbf{P}_{gf}$  は第  $g$  ジェスチャ  $\mathbf{S}_g$  中の第  $f$  フレームの姿勢を表す.

### 4.2 姿勢類似度

胴体の回転に不変な評価値を得るため, 小林ら [7] の手法と同様な複素相関を用いて姿勢類似度を表す. 世界座標における重力方向をカメラ座標の  $y$  軸と一致させておくことで,  $y$  軸まわりの回転 (ほぼ胴体の回転に相当) に不変な類似度を計算できる. なお,  $i$  を虚数単位とし,  $\mathbf{P}$  の  $y$  軸成分以外の 2 軸成分を用いた複素表現  $\mathbf{q}$  (4 次元複素ベクトル) を次式で表す.

$$\mathbf{q} = \mathbf{P}_x + i\mathbf{P}_z \quad (24)$$

また,  $\mathbf{q}$  の複素共役を  $\mathbf{q}^*$  で表す.

$$\mathbf{q}^* = \mathbf{P}_x - i\mathbf{P}_z \quad (25)$$

ここで,  $\mathbf{P}_x, \mathbf{P}_y, \mathbf{P}_z$  はそれぞれ  $\mathbf{P}$  の  $x, y, z$  軸成分のみで構成される 4 次元ベクトルである.

2 つの姿勢  $\mathbf{P}, \mathbf{P}'$  に対応する複素表現が  $\mathbf{q}, \mathbf{q}'$  で表せる場合,  $\mathbf{q}$  と  $\mathbf{q}'$  の内積を以下のように表す.

$$\hat{\epsilon}_P = \mathbf{q}^{*T} \mathbf{q}' = |\hat{\epsilon}_P| e^{i\theta} \quad (26)$$

式 (26) で得られる  $|\hat{\epsilon}_P|$  は,  $y$  軸まわりの回転に不変である. また,  $\theta$  は  $y$  軸まわりの回転を示す. ここに  $y$  軸方向に関する項を加えることで, 2 つの姿勢  $\mathbf{P}, \mathbf{P}'$  の姿勢類似度  $C$  を定義する.

$$C(\mathbf{P}, \mathbf{P}') = \frac{1}{4}(|\hat{\epsilon}_P| + \mathbf{P}_y^T \mathbf{P}'_y) \quad (27)$$

この値は,  $\mathbf{P}'$  を  $y$  軸まわりに  $-\theta$  だけ回転させた時の 2 姿勢間の正規化相関を示している.

### 4.3 ジェスチャ類似度

ジェスチャ類似度は, ジェスチャ全体を複素相関で評価する方法と, 各フレームの複素相関を平均して評価する方法が挙げられる. 以下にそれらのジェスチャ類似度の導出方法について述べる.

#### 4.3.1 全体の複素相関を用いたジェスチャ類似度

$i$  を虚数単位とし,  $\mathbf{S}$  の  $y$  軸成分以外の 2 軸成分を用いた複素表現  $\mathbf{Q}$  ( $4 \times F$  次元複素ベクトル) を以下のように表す.

$$\mathbf{Q} = \mathbf{S}_x + i\mathbf{S}_z \quad (28)$$

また,  $\mathbf{Q}$  の複素共役を  $\mathbf{Q}^*$  で表す.

$$\mathbf{Q}^* = \mathbf{S}_x - i\mathbf{S}_z \quad (29)$$

ここで,  $\mathbf{S}_x, \mathbf{S}_y, \mathbf{S}_z$  はそれぞれ  $\mathbf{S}$  の  $x, y, z$  軸成分のみで構成される  $4 \times F$  次元ベクトルである. 2 つのジェスチャ  $\mathbf{S}, \mathbf{S}'$  に対応する複素表現が  $\mathbf{Q}, \mathbf{Q}'$  で表せる場合,  $\mathbf{Q}$  と  $\mathbf{Q}'$  の内積を以下のように表す.

$$\hat{\epsilon}_S = \mathbf{Q}^{*T} \mathbf{Q}' \quad (30)$$

$$= |\hat{\epsilon}_S| e^{i\theta} \quad (31)$$

式 (31) で得られる  $|\hat{\epsilon}_S|$  は,  $y$  軸まわりの回転に不変である. また,  $\Theta$  は  $y$  軸まわりの回転を示す. ここに  $y$  軸方向に関する項を加えることで, 2つのジェスチャ  $S, S'$  の類似度  $C'_1$  を定義できる.

$$C'_1(S, S') = \frac{1}{4F} (|\hat{\epsilon}_S| + \mathbf{S}_y^T \mathbf{S}'_y) \quad (32)$$

この値は,  $S'$  を  $y$  軸まわりに  $-\Theta$  だけ回転させた時の 2 ジェスチャ間の正規化相関を示している.

#### 4.3.2 各フレームの複素相関を用いたジェスチャ類似度

2つのジェスチャ  $S = [\mathbf{P}_1^T \dots \mathbf{P}_F^T]^T$ ,  $S' = [\mathbf{P}'_1^T \dots \mathbf{P}'_F^T]^T$  があつた場合, 各フレームの姿勢類似度を算出し, それらの平均からジェスチャ類似度を求めることができる. つまり, 式 (33) からジェスチャ類似度  $C'_2$  を求める.

$$C'_2(S, S') = \frac{1}{F} \sum_{i=1}^F C(\mathbf{P}_i, \mathbf{P}'_i) \quad (33)$$

ここで,  $C'_1$  のかわりに  $C'_2$  を使用することで, ジェスチャ動作の途中で体の向きが変化してもよいことになる. すなわち,  $C'_2$  では人物中心座標系によるジェスチャ動作の認識が可能となる.

#### 4.4 ジェスチャ認識

小林ら [7] の手法では各登録ジェスチャの開始フレームと最終フレームを検出したのちに, 登録ジェスチャと推定姿勢との対応付けを行っていた. 本稿では, 各登録ジェスチャの開始フレームが検出された時点から, フレームごとに姿勢類似度を算出し, 登録ジェスチャと推定姿勢との対応付けを行う. まず, 各登録ジェスチャの初期フレームの姿勢と, 現在の推定姿勢との類似度を求め, ジェスチャの開始検出を行う. ここで, 類似度は次式によって計算する.

$$\epsilon_g = C(\mathbf{P}_f, \mathbf{P}_{g1}) \quad (34)$$

これにより, 胴体の回転に不変な評価値を得ることができる [9] [10].  $\mathbf{P}_f$  は第  $f$  入力フレームの姿勢推定結果,  $\mathbf{P}_{g1}$  は  $S_g$  の開始姿勢である.  $\epsilon_g > \tau$  の場合にジェスチャの開始を検出する. ここで,  $\tau$  はしきい値である.

開始フレームが検出された場合, 以下に示す方法により, 最適な入力ジェスチャ  $\hat{S}$  を求める. 前時刻までに, 登録ジェスチャの第  $f (< F)$  フレームまで対応付けされているとする. ここで, 現時刻の推定姿勢と, 登録ジェスチャの第  $f$  フレームとの姿勢類似度を  $C_f$  とする. このとき, 登録ジェスチャの第  $f+1$  フレームとの姿勢類似度  $C_{f+1}$  が  $C_{f+1} > C_f$  の関係にある, もしくは  $C_{f+1}$  がしきい値  $\tau_1$  以上の場合, 登録ジェスチャの第  $f+1$  フレームが検出されたとし,  $f$  の値を  $f+1$  に更新する. こうして  $f$  を時々刻々更新していき,  $f$  が  $F$  に到達したとき, 登録ジェスチャに対応する最適な入力ジェスチャ  $\hat{S} = [\hat{\mathbf{P}}_1^T \hat{\mathbf{P}}_2^T \dots \hat{\mathbf{P}}_F^T]^T$



図 2 上半身モデル

を得る. 本稿では  $\tau_1 = 0.96$  としている.

得られた  $\hat{S}$  を用いて式 (32) あるいは式 (33) により評価を行う.

$$\epsilon_{S_g} = C'_i(\hat{S}, S_g) \quad (i = 1, 2) \quad (35)$$

式 (35) における評価値  $\epsilon_{S_g}$  がしきい値  $\tau_2$  を越えたときジェスチャ  $g$  を認識する.

### 5. 実験

まず追跡性能を評価するために, シミュレーション上での姿勢追跡実験を行う. 次に, 数種類のジェスチャを複数の人物ごとに学習し, 実動画を用いてジェスチャ認識を行う. 本稿では, 距離・画像センサとして Kinect を用いている. このセンサは赤外線パターンを前方に照射し, 赤外線カメラで取得した画像から距離を計算することで, 距離情報をリアルタイムで獲得できる. 距離情報は mm 単位で表され, 500mm 以上離れた対象について有効な距離情報が得られる.

人物モデルは, KinectFusion[11] を用いて 3 人のそれぞれ異なる 3 次元モデルを作成した. 追跡に用いたある 1 人に対する人物モデルを図 2 に示す. 人物モデルは胴体と左右上腕, 左右下腕からなる.

#### 5.1 ジェスチャの種類

4 章で述べた追跡・認識手法について 7 種類のジェスチャを登録して実験を行った. ジェスチャ 1 は右手を人物から向かって左上から右上へ動かすジェスチャであり, 上腕と下腕が重なるオクルージョンが一部発生する. ジェスチャ 2 は胴体の前で両手をあわせるジェスチャであり, 胴体と下腕でオクルージョンが発生する. ジェスチャ 3 は右腕を人物から向かって右から左に動かすジェスチャであり, 比較的短いジェスチャである. ジェスチャ 4 は胴体の横で両腕を下から上下へ動かすジェスチャであり, 比較的長いジェスチャである. ジェスチャ 5, 6 は右腕を胴体の前で回

表 1 実験条件

| 項目                          | 設定値    |
|-----------------------------|--------|
| サンプル数                       | 1500   |
| 粗密探索の段数                     | 3      |
| 初段のサンプル生成範囲 ( $\sigma$ )    | 表 2 参照 |
| 疎点の数                        | 128    |
| 見え画像と奥行き情報の比率 ( $\lambda$ ) | 0.90   |

表 2 初段のサンプル生成範囲 ( $\sigma$ )

| 胴体    |       |       |            |          |          | 肩        |            | 肘        |            |
|-------|-------|-------|------------|----------|----------|----------|------------|----------|------------|
| $t_1$ | $t_2$ | $t_3$ | $\theta_0$ | $\phi_0$ | $\psi_0$ | $\phi_S$ | $\theta_S$ | $\phi_E$ | $\theta_E$ |
| 5.0   | 0.5   | 5.0   | 1.0        | 1.0      | 1.0      | 8.0      | 8.0        | 15.0     | 15.0       |

すジェスチャであり、それぞれ回転する方向が逆になっている。ジェスチャ 7 は胴体の前で両腕を回すジェスチャであり、オクルージョンが複雑なジェスチャである。なお、学習ジェスチャにおいては、ジェスチャの開始、終了を手動で与えている。姿勢推定結果から各ジェスチャの初期姿勢を検出し、各ジェスチャが独立に認識を開始する。各ジェスチャ認識は並列に実行される。

## 5.2 シミュレーション追跡実験

5.1 で述べた 7 つのジェスチャの登録ジェスチャを正解姿勢としシミュレーション実験を行った。シミュレーション実験では、Kinect の入力情報の擬似表現として距離情報の分解能を粗くし、入力距離値を 5[mm] 単位としている。

3.2 で述べた 4 自由度を同時に推定する腕部追跡と、上腕・下腕を順に推定する腕部追跡の比較実験を行う。実験において、14 次元姿勢空間を取り扱っているパラメータ設定の詳細を表 1, 2 に表す。精度評価の尺度として、ジェスチャ認識に用いる 4 つの姿勢ベクトルと正解姿勢ベクトルがなす角度の誤差を用いる。なお、この実験では追跡におけるマスク処理は用いていない。

各登録ジェスチャに対して乱数を変えて 5 回試行を行い、各姿勢ベクトルに対する誤差平均を表 3 と表 4 に示す。これらの結果から、どちらの追跡においてもすべての部位で平均 1[deg] 以下の高精度な推定ができている。表 4 では表 3 と比較して下腕の追跡精度が良い。上腕と下腕を順に推定する追跡では、下腕追跡におけるサンプル生成範囲のパラメータが 2 自由度と少ないため、4 自由度のパラメータを一度に推定する追跡より精度が向上したと考えられる。上腕推定の比較では、ジェスチャ 5、ジェスチャ 6 の右腕や、ジェスチャ 7 といった激しい動作では、4 自由度を同時に推定の方が良い精度を示した。上腕・下腕がともに動くジェスチャに対して、4 次元での追跡が有効だといえる。また、上腕・下腕を順に推定する追跡では、胴体から上腕、上腕から下腕と 2 段階の誤差の伝播を伴うため、ノイズを伴う実動画実験では姿勢推定に不安が残る。4 自由度を同時に推定する追跡は胴体から腕部への 1 段階の誤差

表 3 腕部を 4 次元として扱った場合の追跡による誤差 [deg]

|         | 右上腕   | 左上腕   | 右下腕   | 左下腕   |
|---------|-------|-------|-------|-------|
| ジェスチャ 1 | 0.971 | 0.842 | 0.623 | 0.651 |
| ジェスチャ 2 | 0.720 | 0.628 | 0.658 | 0.673 |
| ジェスチャ 3 | 0.662 | 0.740 | 0.466 | 0.621 |
| ジェスチャ 4 | 0.945 | 0.935 | 0.617 | 0.647 |
| ジェスチャ 5 | 0.584 | 0.867 | 0.644 | 0.658 |
| ジェスチャ 6 | 0.581 | 0.825 | 0.546 | 0.638 |
| ジェスチャ 7 | 0.531 | 0.606 | 0.632 | 0.622 |

表 4 腕部を 2 + 2 次元として扱った場合の追跡による誤差 [deg]

|         | 右上腕   | 左上腕   | 右下腕   | 左下腕   |
|---------|-------|-------|-------|-------|
| ジェスチャ 1 | 0.846 | 0.839 | 0.390 | 0.472 |
| ジェスチャ 2 | 0.687 | 0.660 | 0.470 | 0.517 |
| ジェスチャ 3 | 0.676 | 0.662 | 0.387 | 0.381 |
| ジェスチャ 4 | 0.922 | 0.811 | 0.408 | 0.378 |
| ジェスチャ 5 | 0.737 | 0.885 | 0.730 | 0.468 |
| ジェスチャ 6 | 0.693 | 0.773 | 0.535 | 0.403 |
| ジェスチャ 7 | 0.878 | 0.907 | 0.766 | 0.671 |

の伝播ですみ、複雑な動作でも追跡が安定すると考えられるため、実動画実験においてはこちらを利用する。

## 5.3 マスク処理を含む実動画実験

3.3.1 で述べたマスク処理を含む姿勢推定と、従来の姿勢推定とを比較した。腕部追跡は 4 自由度を同時に推定する追跡を用い、条件は表 1, 2 と同様である。この実験では、最も複雑なジェスチャ 7 について安定的に追跡可能であるかを確認した。

図 3 にそれぞれの追跡結果の例を示す。この図に示したように、マスク処理を含まない姿勢推定では、胴体の推定位置に腕部の推定位置が重なってしまい、追跡が大きく外れ、そのまま追跡が復帰できないことがあった。一方、マスク処理を加えると、腕部推定では胴体の推定位置が排除されるため、追跡を外さずに推定できることがわかる。

## 5.4 実動画におけるジェスチャ認識実験

実動画において、3 人の人物に対しジェスチャ認識実験を行った結果を述べる。5.1 で示した各ジェスチャをそれぞれの人物について撮影し、各人物に対応したモデルを用いて追跡実験を行った。各ジェスチャにつき 20 回の撮影を行い、5 シーケンスを登録ジェスチャとし、残りの 15 シーケンスをテストシーケンスとした。よって、登録ジェスチャは各人物に対応した学習データとなっている。なお、ジェスチャの開始検出に用いるしきい値  $\tau$  およびジェスチャ認識に用いる評価値のしきい値  $\tau_2$  はともに 0.96 とする。

各ジェスチャを評価式  $C'_1$  および  $C'_2$  で評価した際のジェスチャ認識率を表 5 に示す。表 5 内の P1 ~ P3 はそれぞれ異なる人物を示す。なお、実験条件は表 1, 表 2 と同様である。



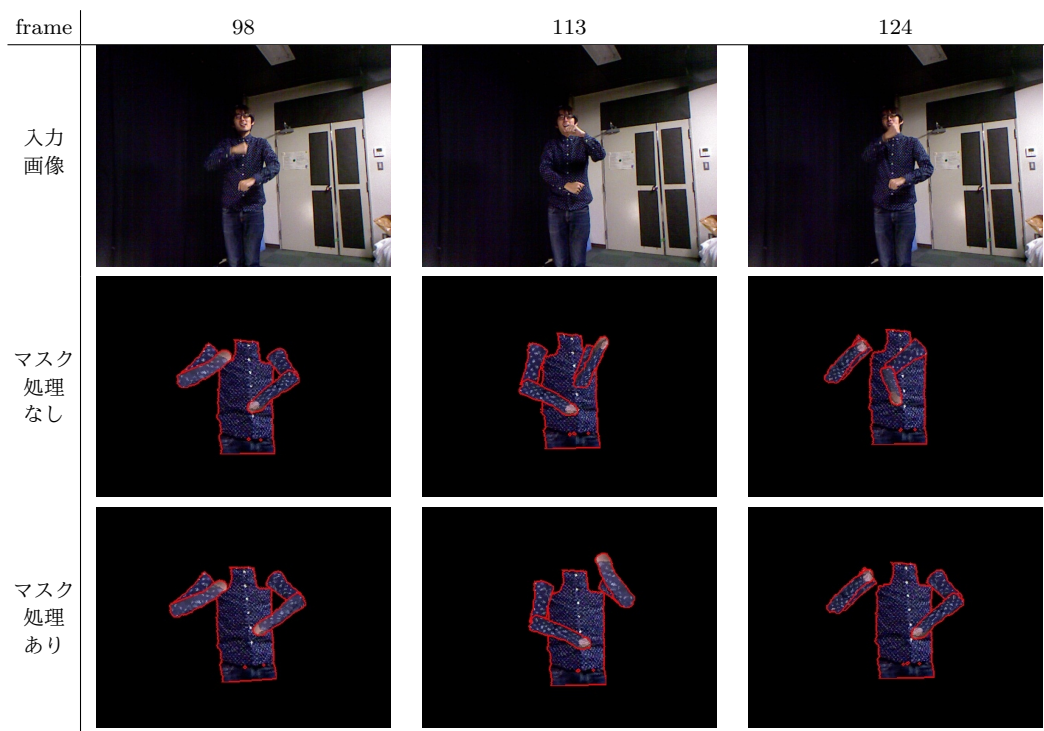


図 3 マスク処理の有無による姿勢追跡結果の比較

表 5 評価式の違いによる認識率の比較

| Person  | $C'_1$ での認識率 |      |      | $C'_2$ での認識率 |      |      |
|---------|--------------|------|------|--------------|------|------|
|         | P1           | P2   | P3   | P1           | P2   | P3   |
| ジェスチャ 1 | 33%          | 87%  | 100% | 100%         | 100% | 100% |
| ジェスチャ 2 | 94%          | 100% | 100% | 94%          | 100% | 100% |
| ジェスチャ 3 | 100%         | 100% | 100% | 100%         | 100% | 100% |
| ジェスチャ 4 | 94%          | 87%  | 100% | 94%          | 100% | 100% |
| ジェスチャ 5 | 100%         | 100% | 100% | 100%         | 100% | 100% |
| ジェスチャ 6 | 100%         | 100% | 100% | 100%         | 100% | 100% |
| ジェスチャ 7 | 87%          | 87%  | 80%  | 94%          | 87%  | 80%  |

表 5 から、全ての人物に対して  $C'_1$  と  $C'_2$  のどちらを用いてもほぼ全てのジェスチャで高い認識率を達成できた。また、一部のジェスチャで  $C'_2$  を用いたジェスチャ認識の方がより高い認識率となった。これは、認識が行われているジェスチャを追跡している間に、いくらか胴体の回転が起こることがあり、 $C'_1$  では各フレームごとの胴体の回転を考慮できていないためだと考えられる。そのため、平均としてはジェスチャ認識の評価値は  $C'_2$  の方が高くなった。認識できなかったジェスチャについてその原因を分析してみると、推定姿勢が外れているものが含まれていた。一方、見た目上正しく姿勢を追跡できていても、ジェスチャの評価値がしきい値を越えない場合もあった。ジェスチャ 7 の姿勢追跡では、特に P3 の人物において服の皺の影響が多分にあった。動作中に腕の動きに従って服の表面が変化し、胴体の評価姿勢がブレやすくなっている。これは、現在の疎テンプレート追跡では補えない問題だと考えられる。

## 5.5 回転動作を伴うジェスチャの追跡実験

人物が回転しながらジェスチャを行う場合に同様なジェスチャ認識ができるかを実験した。胴体方向を軸として回転しながらジェスチャを行うシーケンスを撮影し、ジェスチャ追跡、および認識ができるかを確認した。ジェスチャの認識には  $C'_1$  および  $C'_2$  を用いてジェスチャの評価を行った。なお、この実験においては表 2 の  $\phi_0$  の値を 3.0 に変更している。回転しながらジェスチャ 7 の動作を行った場合の追跡結果を図 4 に示す。正面から外れても比較的良好に追跡することができた。このジェスチャにおいて、 $C'_1$  で評価した場合ジェスチャ全体の評価値が 0.90 程度だったのに対し、 $C'_2$  を用いることでジェスチャ全体の評価値が 0.96 となり、ジェスチャの認識に成功することが確認できた。

## 6. まとめ

本稿では距離・画像センサを用いた人物姿勢追跡およびジェスチャ認識について議論した。現時刻の胴体推定位置と距離情報からマスク情報を生成し姿勢推定に利用した。この処理により、実動画実験では安定的な追跡を行えることを示した。ジェスチャ認識では、ジェスチャ認識の対象人物を 3 人に拡張した場合でも、従来と同様な認識率を得られた。フレームごとに登録ジェスチャとの姿勢類似度を算出し、その結果をジェスチャ認識に用いることで、回転しながらジェスチャを行った場合でもジェスチャを認識することができた。

上腕と下腕を同時に推定することで、相互の安定な姿勢推定が行えたが、上腕と下腕の重なり合いや、両腕の関係

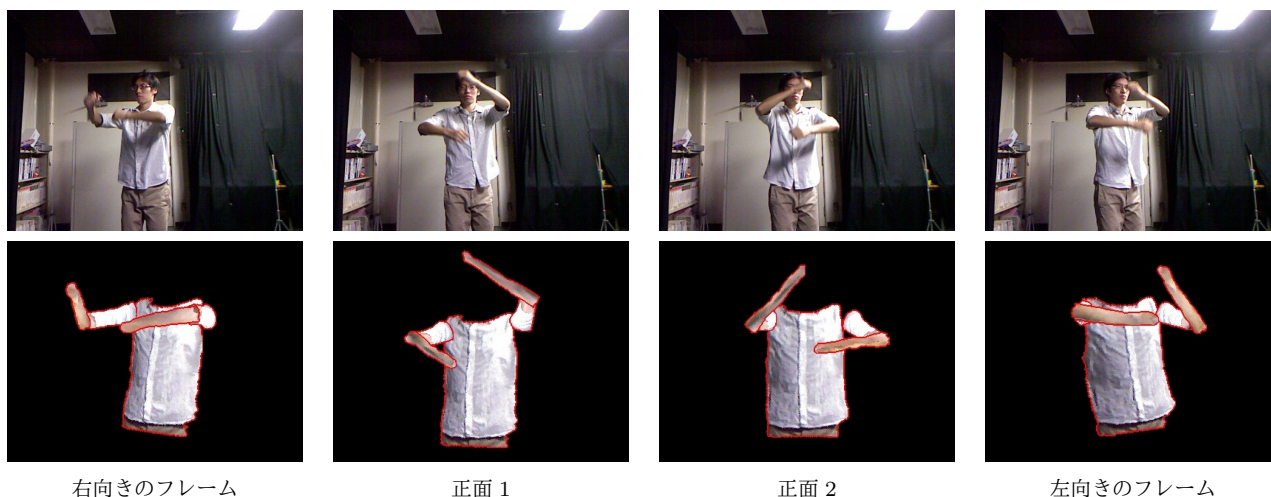


図 4 胴体の回転を伴うジェスチャ（上段）と姿勢追跡結果（下段）

については議論できていない。そのため上腕と下腕が重なる姿勢や、両腕が交差した姿勢では追跡が外れてしまう場合があった。現状のカスケード推定では、腕部同士のオクルージョンに対応できないため、下腕からの姿勢推定といった改良や、手の位置検出による腕部の姿勢推定を考えていく必要がある。また、現在の実装ではプログラムの最適化が十分でないため、追跡 1 ステップにおいて約 400ms を要している。実時間処理を実現するためには、プログラムの最適化を行うとともに GPU 実装などによる高速化が必要と考える。また、現在の実験では初期姿勢や胴体の関節位置を手動で与えている。これらのパラメータを変化させたとき、姿勢推定やジェスチャ認識に影響があることがわかっている。初期姿勢の検出や関節位置検出の自動化が今後の課題である。

#### 参考文献

- [1] Y. Song, D. Demirdjian and R. Davis: Tracking Body and Hands For Gesture Recognition: NATOPS Aircraft Handling Signals Database, *Proc. 9th IEEE Conference on Automatic Face and Gesture Recognition (FG 2011)*, pp. 500–506 (2011).
- [2] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman and A. Blake: Real-time human pose recognition in parts from single depth images, *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR'11)*, pp. 1297–1304 (2011).
- [3] T. Shakunaga: Pose estimation of jointed structures, *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR'91)*, pp. 566–572 (1991).
- [4] 佐竹純二, 尺長健: 階層的注視点制御による動画像上での複数人物追跡, 電子情報通信学会論文誌 (D-II), Vol. J86-D-II, No. 8, pp. 1212–1221 (2003).
- [5] S. Knoop, S. Vacek and R. Dillmann: Sensor fusion for 3D human body tracking with an articulated 3D body model, *Proc. IEEE International Conference on Robotics and Automation (ICRA 2006)*, pp. 1686–1691 (2006).
- [6] 四宮洋平, 加藤秀章, 尺長健: Kinect を用いた 3 次元疎テンプレートによる人物姿勢追跡, 電子情報通信学会技

術研究報告 PRMU2011-264 (2012).

- [7] 小林宏章, 加藤秀章, 尺長健: 距離・画像センサと複素相関を用いた回転不変なジェスチャ認識, 情報処理学会研究報告 2013-CVIM-186, pp. 1–8 (2013).
- [8] Y. Oka, T. Kuroda, T. Migita and T. Shakunaga: Tracking 3d Pose of Rigid Object by Sparse Template Matching, *Proc. International Conference on Image and Graphics*, pp. 390–397 (2009).
- [9] 山根亮, 川嶋幸治, 戸高千智, 尺長健: 動作データの時系列相関行列による舞踊動作解析, 電子情報通信学会論文誌 (D-II), Vol. J88-D-II, No. 8, pp. 1652–1661 (2005).
- [10] R. Yamane and T. Shakunaga: Dance Motion Analysis by Correlation Matrix Between Pose Sequences, *Proc. International Conference on Image and Vision Computing, New Zealand (IVCNZ2010)*, pp. 1–6 (2010).
- [11] R. A. Newcombe, A. J. Davison, S. Izadi, P. Kohli, O. Hilliges, J. Shotton, D. Molyneaux, S. Hodges, D. Kim and A. Fitzgibbon: KinectFusion: Real-Time Dense Surface Mapping and Tracking, *ISMAR2011*, pp. 127–136 (2011).