Vol.2013-CVIM-186 No.17
2013/3/15

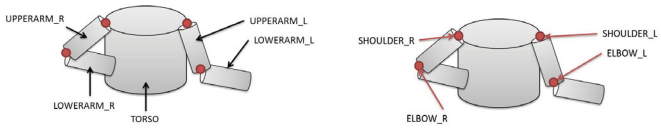
距離・画像センサと複素相関を用いた回転不変なジェスチャ認識

小林 宏章[1]　加藤 秀章[1]　尺長 健[1]

概要：距離・画像センサを用いたジェスチャ認識について述べる．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を提案している．本稿では，この方法によって時々刻々得られる人物上半身の姿勢を，重力方向まわりの回転に不変な類似度を複素相関を用いて構成することにより，胴体の回転に不変なジェスチャ認識法を提案する．また，実データに適用した結果を報告する．

1. はじめに

ジェスチャ認識は, 人間と計算機の間の自然なユーザインタフェースとして注目されている [1][2]．映像に基づくジェスチャ認識においては，従来，背景と人物の分離が難しかったことから，顕著な画像特徴に重点をおいた追跡 [3][4] を試みたり，動作に関する学習を利用したトップダウン型の認識系が提案されてきた．一方, 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, 距離情報を併用することで, 容易に人物領域を抽出でき, さらに距離情報を用いることで人物の追跡の精度も向上し, より正確なジェスチャ認識が行えると考えられる．四宮ら [5] 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．本稿では, この姿勢追跡により得られる時系列姿勢情報を用いてジェスチャ認識系を構成する．このために，まず，姿勢間の類似度，および，ジェスチャ間の類似度を定義する．本稿では，重力方向（通常の姿勢では，胴体の軸とほぼ一致する）まわりの回転に不変な姿勢類似度を複素相関を用いて定義し, これをベースとして，胴体の回転に不変なジェスチャ認識を目指す．第 2 章では, 上半身の関節物体としての取扱いと, 3 次元姿勢追跡系の概要を述べる．第 3 章では，まず，複素相関を用いて姿勢類似度，および，ジェスチャ類似度を定義する．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 4 章では本稿で提案するジェスチャ認識を用いた実動画実験の結果を述べる．カメラに対して正面を向いてジェスチャを行う動画像だけではなく, 15, 30, 45 度の各方向を向いてジェスチャを行う動画像においても実験を行い, 結果を述べる．



各部位の名称　　各関節の名称

図 1　関節物体モデル

2. 距離・画像センサによる関節物体の姿勢追跡

2.1 関節物体の姿勢

本稿で追跡の対象とするのは, 人間の上半身である．本稿では, 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．この上半身モデルの各部位と各関節を図 1 示す．なお, 左肩, 右肩の回転自由度を 2 とし, 左肘, 右肘の回転自由度もまた 2 とする．

2.2 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 つの座標系を考える．

(1) $\boldsymbol{M}_C$：カメラ (中心) 座標系
(2) $\boldsymbol{M}_O$：胴体 (中心) 座標系
(3) $\boldsymbol{M}_U$：上腕 (中心) 座標系
(4) $\boldsymbol{M}_L$：下腕 (中心) 座標系

なお, $\boldsymbol{M}_U$ と $\boldsymbol{M}_L$ については左右に別々に座標系を考える必要があるため, 実装上は 6 個の座標系を考える必要がある．本稿では, 左右の相互作用は考慮しないため, 以下で

[1] 岡山大学
Okayama University

は 4 つの座標系を用いた議論を行う.

2.3 3 次元サーフィスモデル

本稿では, 追跡対象である 3 次元物体は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. ここで, 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, 下腕座標系を用いて表す.

一方, 画像生成過程はカメラ座標で記述される. ある姿勢の物体モデルから画像を生成するためには, 胴体/上腕/下腕座標からカメラ座標 $\boldsymbol{M}_C = [X_C\, Y_C\, Z_C]^\top$ 変換する必要がある. 表記の簡単化のために,$\boldsymbol{M}_C, \boldsymbol{M}_O, \boldsymbol{M}_U, \boldsymbol{M}_L$ の同次座標表現 $\widetilde{\boldsymbol{M}}_C = [\boldsymbol{M}_C^\top\ 1]^\top$, $\widetilde{\boldsymbol{M}}_O = [\boldsymbol{M}_O^\top\ 1]^\top$, $\widetilde{\boldsymbol{M}}_U = [\boldsymbol{M}_U^\top\ 1]^\top$, $\widetilde{\mathrm{M}}_L = [\boldsymbol{M}_L^\top\ 1]^\top$ を用いる.

2.3.1 胴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 4×4 姿勢行列で取り扱う. 胴体座標系で表される 3 次元点の同次表現を $\widetilde{\boldsymbol{M}}_O = [\boldsymbol{M}_O^\top\ 1]^\top$ とし, 胴体座標系からカメラ座標系への変換 $\boldsymbol{D}_{OC}$ を胴体の姿勢行列と考えることができる.

$$\boldsymbol{D}_{OC} = \begin{bmatrix} \boldsymbol{R}_{OC} & \boldsymbol{t}_{OC} \\ \boldsymbol{0}^\top & 1 \end{bmatrix} \tag{1}$$

$$\widetilde{\boldsymbol{M}}_C = \boldsymbol{D}_{OC}\widetilde{\boldsymbol{M}}_O \tag{2}$$

ここで式 (2) における $\boldsymbol{R}_{OC}$ は胴体の回転を表す行列であり, $\boldsymbol{t}_{OC}$ は胴体の並進を表すベクトルである.

2.3.2 上腕の表現と姿勢行列

上腕の姿勢は, 胴体の姿勢 $\boldsymbol{D}_{OC}$ と肩を中心とした回転の合成により得られる. 上腕座標系で表される 3 次元点の同次表現を $\boldsymbol{M}_U$ とし, 上腕座標系から胴体座標系への変換を $\boldsymbol{D}_{UO}$ と表記する. 胴体座標系における肩関節の位置を $\boldsymbol{t}_S$ 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, 肩での回転を表す回転行列を $\boldsymbol{R}_S$ 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.

$$\boldsymbol{D}_{UO} = \begin{bmatrix} \boldsymbol{R}_S & \boldsymbol{t}_S \\ \boldsymbol{0}^\top & 1 \end{bmatrix} \tag{3}$$

$$\widetilde{\boldsymbol{M}}_C = \boldsymbol{D}_{UC}\widetilde{\boldsymbol{M}}_U \tag{4}$$

ここで,

$$\boldsymbol{D}_{UC} = \boldsymbol{D}_{OC}\boldsymbol{D}_{UO} \tag{5}$$

$$\boldsymbol{R}_S = \begin{bmatrix} c_{\theta_S} & 0 & s_{\theta_S} \\ 0 & 1 & 0 \\ -s_{\theta_S} & 0 & c_{\theta_S} \end{bmatrix} \begin{bmatrix} c_{\phi_S} & -s_{\phi_S} & 0 \\ s_{\phi_S} & c_{\phi_S} & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{6}$$

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

2.3.3 下腕の表現と姿勢行列

下腕の姿勢は, 上腕の姿勢 $\boldsymbol{D}_{UC}$ と, 肘を中心とした回転の合成により得られる. 下腕座標系で表される 3 次元点の同次表現を $\boldsymbol{M}_L$ とし, 下腕座標系から上腕座標系への変換を $\boldsymbol{D}_{LU}$ と表記する. 上腕座標系における肘の位置を $\boldsymbol{t}_E$ 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 $\boldsymbol{R}_E$ 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.

$$\boldsymbol{D}_{LU} = \begin{bmatrix} \boldsymbol{R}_E & \boldsymbol{t}_E \\ \boldsymbol{0}^\top & 1 \end{bmatrix} \tag{7}$$

$$\widetilde{\boldsymbol{M}}_C = \boldsymbol{D}_{LC}\widetilde{\boldsymbol{M}}_L \tag{8}$$

ここで,

$$\boldsymbol{D}_{LC} = \boldsymbol{D}_{UC}\boldsymbol{D}_{LU} \tag{9}$$

$$\boldsymbol{R}_E = \begin{bmatrix} c_{\phi_E} & -s_{\phi_E} & 0 \\ s_{\phi_E} & c_{\phi_E} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_{\theta_E} & 0 & s_{\theta_E} \\ 0 & 1 & 0 \\ -s_{\phi_E} & 0 & c_{\theta_E} \end{bmatrix} \tag{10}$$

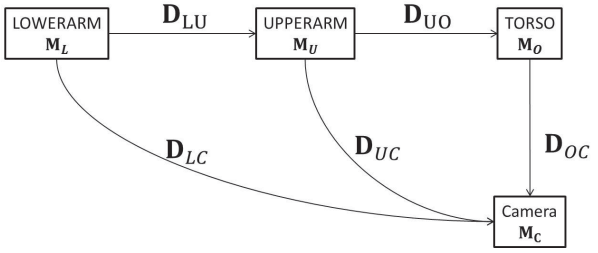


図 2 変換の関係

なお, $c_{\theta_E} = \cos\theta_E$,$s_{\theta_E} = \sin\theta_E$ とする. 4 つの座標系の相互変換をまとめて図 2 に示す. 座標系 $\boldsymbol{M}_L$, $\boldsymbol{M}_U$, $\boldsymbol{M}_O$ からカメラ座標系 $\boldsymbol{M}_C$ への座標変換は, それぞれ $\boldsymbol{D}_{LC}$, $\boldsymbol{D}_{UC}$, $\boldsymbol{D}_{OC}$ で行える. また, 図中の座標変換の逆変換は逆行列で表される. 例えば, $\boldsymbol{D}_{CL} = \boldsymbol{D}_{LC}^{-1}$ である.

2.4 基準姿勢における 3 次元疎テンプレートの作成

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. 以下に, その詳細を述べる. 前時刻 $t-1$ の推定姿勢が与えられた場合, この姿勢から姿勢行列 $\boldsymbol{D}_{t-1}$ を算出できる. 以下では, 前時刻の推定姿勢から予想される姿勢を基準姿勢とよぶ. 基準姿勢から生成される姿勢行列を基準姿勢行列と呼び, $\hat{\boldsymbol{D}}$ と表す. (簡単には, $\hat{\boldsymbol{D}} = \boldsymbol{D}_t$ としてよい.) 通常の Z-buffer 法の拡張として, 画像 $I(u,v)$, 距離 $Z(u,v)$ を更新する際に, 同時に $X(u,v)$,$Y(u,v)$ の値も更新することにより各画素 (u,v) において画素値 $I(u,v)$ に対応する 3 次元座標 $[X(u,v)Y(u,v)Z(u,v)]$ を計算する. これにより, 基準姿勢行列 $\hat{\boldsymbol{D}}$ によって作成されるテンプレート画像 $\hat{I}(u,v)$ から 3 次元疎テンプレートを作成できる. すなわち, 局所領域内で輝度値が最大・最小となる点を選択す

ることにより, 疎テンプレート $\{(u_j, v_j)\}$ を作成し, この各点 (u_j, v_j) に対応する 3 次元座標を $\boldsymbol{M}_j$ 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.

$$\{(\boldsymbol{M}_j; \hat{I}_j)\} = \{(X_j\ Y_j\ Z_j; \hat{I}_j)\} \tag{11}$$

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.

2.5 3 次元疎テンプレートマッチング

3 次元疎テンプレートを構成する j 番目の点 $\boldsymbol{M}_j$ が胴体, 上腕, 下腕のどこに属すかによって, 式 (12) によって画像上の位置 $\boldsymbol{m}_{jk}$ を算出することになる. すなわち, 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 $\delta\boldsymbol{D}_k$,$\delta\boldsymbol{D}'_k$,$\delta\boldsymbol{D}''_k$ を用いて, 画像座標 $\boldsymbol{m}_{jk}$ が計算される.

$$\begin{cases} \boldsymbol{m}_{jk} = (\boldsymbol{A})\delta\boldsymbol{D}_k(\boldsymbol{D}_{OC}\widetilde{\boldsymbol{M}}_j) \\ \boldsymbol{m}_{jk} = (\boldsymbol{A}\boldsymbol{D}_{OC})\delta\boldsymbol{D}'_k(\boldsymbol{D}_{UO}\widetilde{\boldsymbol{M}}_j) \\ \boldsymbol{m}_{jk} = (\boldsymbol{A}\boldsymbol{D}_{UC})\delta\boldsymbol{D}''_k(\boldsymbol{D}_{LU}\widetilde{\boldsymbol{M}}_j) \end{cases} \tag{12}$$

同様に, 距離 $= \boldsymbol{Z}_{jk}$ は次式によって求められる.

$$\begin{cases} (X_{jk}Y_{jk}Z_{jk}1)^\top = \delta\boldsymbol{D}_k(\boldsymbol{D}_{OC}\widetilde{\boldsymbol{M}}_j) \\ (X_{jk}Y_{jk}Z_{jk}1)^\top = (\boldsymbol{D}_{OC})\delta\boldsymbol{D}'_k(\boldsymbol{D}_{UO}\widetilde{\boldsymbol{M}}_j) \\ (X_{jk}Y_{jk}Z_{jk}1)^\top = (\boldsymbol{D}_{UC})\delta\boldsymbol{D}''_k(\boldsymbol{D}_{LU}\widetilde{\boldsymbol{M}}_j) \end{cases} \tag{13}$$

ただし, $\widetilde{\boldsymbol{M}}_j$ は $\boldsymbol{M}_j$ の同次座標を示す. また, $\delta\boldsymbol{D}_k$, $\delta\boldsymbol{D}'_k$, $\delta\boldsymbol{D}''_k$ はサンプル姿勢の 14 パラメータ $[\alpha_1\ \alpha_2\ \ldots\alpha_{14}]^\top = [\delta\theta_0\ \delta\phi_0\ \delta\psi_0\ \delta t_1\ \delta t_2\ \delta t_3\ \delta\phi_S\ \delta\theta_S\ \delta\phi_E\ \delta\theta_E\ \delta\phi'_S\ \delta\theta'_S\ \delta\phi'_E\ \delta\theta'_E]$ を用いて次式で示される.

$$\delta\boldsymbol{D}_k = \begin{bmatrix} c_2c_3 & -c_2s_3 & s_2 & \delta t_1 \\ s_1s_2c_3 + c_1s_3 & c_1c_3 - s_1s_2s_3 & -s_1c_2 & \delta t_2 \\ s_1s_3 - c_1s_2c_3 & c_1s_2s_3 + s_1c_3 & c_1c_2 & \delta t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{14}$$

$$\delta\boldsymbol{D}'_k = \begin{bmatrix} c_jc_{j+1} & -c_js_{j+1} & s_j & 0 \\ s_{j+1} & c_{j+1} & 0 & 0 \\ s_js_{j+1} & s_js_{j+1} & c_j & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{15}$$

$$\delta\boldsymbol{D}''_k = \begin{bmatrix} c_{j+2}c_{j+3} & -s_{j+3} & s_{j+3}c_{j+3} & 0 \\ c_{j+2}s_{j+3} & c_{j+3} & s_{j+2}s_{j+3} & 0 \\ -s_{j+2} & 0 & c_{j+2} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{16}$$

ここで, $c_i = cos\alpha_i$, $s_i = sin\alpha_i$ を示す. 式 (15), (16) 中において, 右手系では $j = 7$, 左手系では $j = 11$ である.

2.6 粗密追跡法

追跡では, 姿勢サンプル集合の一部をフレームからフレームへと伝搬させる. ガウスノイズが各サンプルのすべての姿勢パラメータに付加され, その生成されたサンプルから姿勢行列を算出し, 最終的にすべてのサンプルは疎テンプレートマッチングによって評価される [6]. 本稿では, 粗密追跡法 [7] を用いることで, サンプル数の削減を図る. 粗密追跡法では, 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.

2.7 カスケード姿勢推定

本稿では, 人体 (上半身) の姿勢は胴体に 6 自由度, 肩に 2 自由度, 肘に 2 自由度をもつものとしている. したがって上半身の追跡には 14 次元の姿勢空間を取り扱う必要がある.

本稿では, 胴体, 右腕, 左腕の順に姿勢を推定することで, 姿勢空間を小さく抑えながら高次元の姿勢空間を探索する. まず, 胴体の姿勢の推定を行い, 胴体姿勢が推定された後, 右腕の姿勢 (4 自由度) ついて姿勢サンプルを生成し, 姿勢を推定する. 最後に, 胴体と右腕の姿勢を用いて左腕のサンプル姿勢を 4 次元姿勢空間で作成し, 姿勢を推定する. 各段の計算では, 基準姿勢のまわりでサンプル姿勢に対応する疎テンプレートの位置を効率よく計算する.

2.8 距離・画像センサを用いたテンプレートマッチング

本稿の追跡では, 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. これを利用し, 距離・画像センサから得られた距離情報と物体モデルの各サンプルにおける奥行きを比較することで, 画像情報のみを用いた追跡よりも高性能な追跡を期待できる.

2.8.1 距離情報を用いた人物追跡と背景領域の分離

画像情報では人物領域と背景領域の分離を厳密に行うことは一般には容易でない. しかし, 距離情報が追跡時に利用できる場合には, 距離のギャップを利用して簡単に両者を分離できる. すなわち人物領域と背景領域では, 明らかに距離情報にギャップがあり, また, 背景の方がカメラから遠いことを利用し, 人物領域のみを抽出し, マスクをかけることができる. これにより見え画像によるテンプレートマッチングの精度が向上すると考えられる.

ここでは $d_0 \sim d_1$[mm] の範囲内で人物が観測されるとする. 各画素 $I(x, y)$ に対応する距離を $d(x, y)$ とすると, 人物領域のマスク $m(x, y)$ は以下の式 (17) により表される.

$$m(x, y) = \begin{cases} 1 & d_0 \le d(x, y) \le d_1 \\ 0 & otherwise \end{cases} \tag{17}$$

なお, 実装では $d_0 = 500$, $d_1 = 2400$ としている. 見え画像とこの処理により抽出した人物領域の一例を図 3 に示す.

2.8.2 マスクを用いたテンプレートマッチング

前節で作成したマスク $m(u, v)$ を用いて, 入力画像 I でテンプレートマッチングを行い, 画像情報の評価値 ϵ_k を求める.


見え画像　　人物マスク領域

図 3　見え画像と人物マスク領域

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

ただし,

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

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

なお，α はテンプレート画像の正規化係数，β は入力画像の正規化係数，$\rho(x)$ は Geman-McClure 関数である．また c は定数であり, ここでは，$c = 0.5/n$ とする．(ただし n はテンプレートの点数)

2.8.3　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 $\delta \boldsymbol{D}_k$ が与えられたとき, 入力距離情報の評価値 ϵ'_k は, テンプレートの j 番め点の距離情報 Z_{jk} と入力距離情報 $d(j)$ を用いて次式で計算される．

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

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 $1mm$ 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．

2.8.4　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 ϵ_k と距離情報に対して算出した評価値 ϵ'_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 ϵ を両者の重み付き和によって次式で定義する．

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

3.　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, 得られる姿勢の時系列平均を登録ジェスチャとする．これにより得られる登録ジェスチャと入力における姿勢情報とを評価し, 入力動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．

3.1　ジェスチャ認識の流れ

まず, 姿勢追跡によって得られた現在の姿勢と, 各登録ジェスチャの開始フレームの平均姿勢とを照合し, ジェスチャの開始を検出する．次に, 同様の方法でジェスチャ終了の検出を行う．開始と終了が検出さた時点で, 登録ジェスチャと入力ジェスチャの対応付けを行う．これは, 人物がジェスチャを行うときロバストに認識するためである．最後に, 対応付けにより得られた入力ジェスチャと登録ジェスチャとを照合し, そのジェスチャであるかどうかを判定する．

3.2　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．第 g 登録ジェスチャを $\boldsymbol{S}_g$ と表す．$\boldsymbol{S}_g$ は 30 フレームの姿勢 (12 次元) で表される 360 次元である．また, 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 $f (f = 1, 2, \ldots, F)$ フレームを $\boldsymbol{S}_{gf}$ で表す．本稿では, 登録ジェスチャのフレーム $F = 30$ とする．

以下に, 登録ジェスチャの要素として扱うベクトルについて説明する．なお, $\boldsymbol{D}_{UC}$ は上腕座標からカメラ座標への変換行列であり, $\boldsymbol{D}_{OC}$ は胴体座標からカメラ座標への変換行列である．詳しくは論文 [5] で定義されている．上腕座標における肘の座標 $\widetilde{\boldsymbol{M}}_{UE}$ を以下の式でカメラ座標 $\widetilde{\boldsymbol{M}}_{CE}$ にする．

$$\widetilde{\boldsymbol{M}}_{CE} = \boldsymbol{D}_{UC} \widetilde{\boldsymbol{M}}_{UE} \quad (21)$$

同様に, 胴体座標における肩の座標 $\widetilde{\boldsymbol{M}}_{OS}$ をカメラ座標 $\widetilde{\boldsymbol{M}}_{CS}$ にする．

$$\widetilde{\boldsymbol{M}}_{CS} = \boldsymbol{D}_{OC} \widetilde{\boldsymbol{M}}_{OS} \quad (22)$$

$\boldsymbol{M}_{CS}$ と $\boldsymbol{M}_{CE}$ より上腕の 3 次元ベクトル $\boldsymbol{V}_1$ を以下の式で求める．

$$\boldsymbol{V}_1 = \frac{\boldsymbol{M}_{CE} - \boldsymbol{M}_{CS}}{|\boldsymbol{M}_{CE} - \boldsymbol{M}_{CS}|} \quad (23)$$

下腕座標における手の座標 $\widetilde{\boldsymbol{M}}_{UH}$ を以下の式で上腕座標 $\widetilde{\boldsymbol{M}}_{CH}$ にする．

$$\widetilde{\boldsymbol{M}}_{CH} = \boldsymbol{D}_{UC} \widetilde{\boldsymbol{M}}_{UH} \quad (24)$$

式 (21) で求めた肘の座標 $\boldsymbol{M}_{CE}$ と式 (24) で求めた手の座標 $\boldsymbol{M}_{CH}$ より, 下腕の 3 次元ベクトル $\boldsymbol{V}_2$ を以下の式で求める．

$$\boldsymbol{V}_2 = \frac{\boldsymbol{M}_{CH} - \boldsymbol{M}_{CE}}{|\boldsymbol{M}_{CH} - \boldsymbol{M}_{CE}|} \tag{25}$$

また, $\boldsymbol{V}_1$ は右上腕, $\boldsymbol{V}_2$ は右下腕のベクトルとし, 同様にして求めた, $\boldsymbol{V}_3$,$\boldsymbol{V}_4$ をそれぞれ左上腕, 左下腕のベクトルとする. 姿勢 $\boldsymbol{P}$ は, 単位方向ベクトル $\boldsymbol{V}_1$,$\boldsymbol{V}_2$,$\boldsymbol{V}_3$,$\boldsymbol{V}_4$ を用いて以下のように表す.

$$\boldsymbol{P} = [\boldsymbol{V}_1^T\ \boldsymbol{V}_2^T\ \boldsymbol{V}_3^T\ \boldsymbol{V}_4^T]^T \tag{26}$$

登録ジェスチャは式 (26) を用いて以下のように表す.

$$\boldsymbol{S}_g = [\boldsymbol{P}_{g1}{}^T\ \boldsymbol{P}_{g2}{}^T\ \ldots\ \boldsymbol{P}_{gF}{}^T]^T \tag{27}$$

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

3.3.1 姿勢類似度

胴体の回転に不変な評価値を得るため, 本稿では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, $\boldsymbol{P}$ の y 軸成分以外の 2 軸成分を用いた複素表現 $\boldsymbol{q}$ (4 次元複素ベクトル) を次式で表す.

$$\boldsymbol{q} = \boldsymbol{P}_x + i\,\boldsymbol{P}_z \tag{28}$$

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

$$\boldsymbol{q}^* = \boldsymbol{P}_x - i\,\boldsymbol{P}_z \tag{29}$$

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

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

$$\hat{\epsilon}_P = \boldsymbol{q}^{*T}\boldsymbol{q}' \tag{30}$$
$$= |\hat{\epsilon}_P|e^{i\theta} \tag{31}$$

式 (31) 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, θ 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 $\boldsymbol{P}$, $\boldsymbol{P}'$ の姿勢類似度 C を定義できる.

$$C(\boldsymbol{P}, \boldsymbol{P}') = \frac{1}{4}(|\hat{\epsilon}_P| + \boldsymbol{P}_y{}^T\boldsymbol{P}'_y) \tag{32}$$

この値は, $\boldsymbol{P}'$ 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.

3.3.2 ジェスチャ類似度

i を虚数単位とし, $\boldsymbol{S}$ の y 軸成分以外の 2 軸成分を用いた複素表現 $\boldsymbol{Q}$ (120 次元複素ベクトル) を以下のように表す.

$$\boldsymbol{Q} = \boldsymbol{S}_x + i\,\boldsymbol{S}_z \tag{33}$$

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

$$\boldsymbol{Q}^* = \boldsymbol{S}_x - i\,\boldsymbol{S}_z \tag{34}$$

ここで, $\boldsymbol{S}_x$, $\boldsymbol{S}_y$, $\boldsymbol{S}_z$ はそれぞれ $\boldsymbol{S}$ の x, y, z 軸成分のみで構成される 120 次元ベクトルである. 2 つのジェスチャ $\boldsymbol{S}$, $\boldsymbol{S}'$ に対応する複素表現が $\boldsymbol{Q}$, $\boldsymbol{Q}'$ で表せる場合, $\boldsymbol{Q}$ と $\boldsymbol{Q}'$ の内積を以下のように表す.

$$\hat{\epsilon}_S = \boldsymbol{Q}^{*T}\boldsymbol{Q}' \tag{35}$$
$$= |\hat{\epsilon}_S|e^{i\Theta} \tag{36}$$

式 (36) 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, Θ 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 $\boldsymbol{S}$, $\boldsymbol{S}'$ の類似度 C' を定義できる.

$$C'(\boldsymbol{S}, \boldsymbol{S}') = \frac{1}{4F}(|\hat{\epsilon}_S| + \boldsymbol{S}_y{}^T\boldsymbol{S}'_y) \tag{37}$$

この値は, $\boldsymbol{S}'$ 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.

3.4 ジェスチャの認識

3.4.1 ジェスチャ開始, 終了検出

本稿のジェスチャ認識では, 現在推定されている姿勢と各登録ジェスチャの初期フレームの姿勢と現在推定されている姿勢との類似度を求め, ジェスチャの開始を検出する. ここで, 類似度は次式によって計算する.

$$\epsilon_g = C(\boldsymbol{P}_f, \boldsymbol{P}_{g1}) \tag{38}$$

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 [8] [9] . また, $\boldsymbol{P}_f$ は, 現在の姿勢推定結果, $\boldsymbol{P}_{g1}$ は $\boldsymbol{S}_g$ の第 1 フレームの姿勢である.

各ジェスチャの開始が確認されると, ジェスチャの終了検出を行う. このとき用いられる類似度は次式で示される.

$$\epsilon'_g = C(\boldsymbol{P}_f, \boldsymbol{P}_{gF}) \tag{39}$$

ここで, $\boldsymbol{P}_{gF}$ は第 g ジェスチャ $\boldsymbol{S}_g$ の第 $F(=30)$ フレームの姿勢である. $\epsilon_g > \tau_1$, $\epsilon'_g > \tau_1$ のときに開始, 終了を検出する. ここで, τ_1 は閾値である.

3.4.2 ジェスチャの対応付け

ジェスチャを行う場合, そのときどきで, 全体, または部分的な速さが登録されているジェスチャと異なることがある. そこで, Algorithm 1 で示す動的計画法を用いて, 最適な入力ジェスチャ $\hat{\boldsymbol{S}}$ を求める (ジェスチャの開始フレームから終了フレームまでの推定姿勢が登録姿勢のどのフレームと対応するか求める). ここで, $C_{i,j}$ は, 登録ジェスチャの第 i フレームと入力画像の第 j フレームの類似度を示す.

3.4.3 評価

得られた $\hat{\boldsymbol{S}}$ を用いて式 (40) により評価を行う.

$$\epsilon_{S_g} = C'(\hat{\boldsymbol{S}}, \boldsymbol{S}_g) \tag{40}$$

$\epsilon_{S_g} > \tau_2$, のときジェスチャ g を認識したとする. ここで, τ_2 は閾値である.

Algorithm 1 for $\hat{S}$

$i \leftarrow I, j \leftarrow J$
while $i \neq 1 \vee j \neq 1$ **do**
  if $\max\{\boldsymbol{C}_{i-1,j}, \boldsymbol{C}_{i,j-1}, \boldsymbol{C}_{i-1,j-1}\} = \boldsymbol{C}_{i-1,j}$ **then**
    $(i,j) \leftarrow (i-1,j)$
  else if $\max\{\boldsymbol{C}_{i-1,j}, \boldsymbol{C}_{i,j-1}, \boldsymbol{C}_{i-1,j-1}\} = \boldsymbol{C}_{i,j-1}$ **then**
    $(i,j) \leftarrow (i,j-1)$
  else
    $(i,j) \leftarrow (i-1,j-1)$
  end if
end while

3.5 多段階認識による最適化

3.4 節で述べた手法について, 登録ジェスチャを 30 フレームとして扱う場合だけでなく, L 段階に分け, $30/L$ フレームとして扱った場合を考える. ここで, 第 l $(l = 2, 3, \ldots, L)$ 段階においての開始検出は, 第 $l-1$ 段階の終了検出とする. 第 L 段階目の式 (40) における ϵ'_g が閾値を越えた場合そのジェスチャを認識したとする. これに伴い, 第 1 段階目の終了検出は式 (39) でなく, 次の式による.

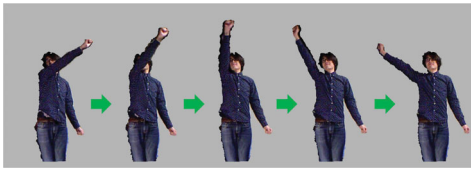
$$\epsilon_{gl} = C(\boldsymbol{P}_f, \boldsymbol{P}_{g(Fl/L)}) \tag{41}$$

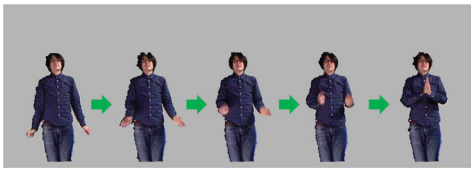
4. 実験

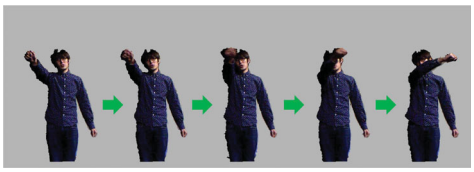
4.1 ジェスチャの種類

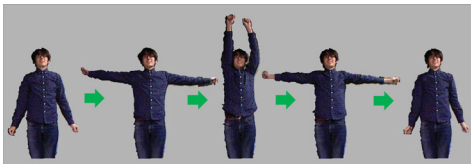
述べた追跡, 認識手法について 7 種類のジェスチャを登録して実験を行った. 各登録ジェスチャの初期フレームから最終フレームの一部を図 4 に図示する. 各ジェスチャにつき 30 個のシーケンスを用意した. ジェスチャ 1 は右手を右上から左上へ動かすジェスチャである. ジェスチャ 2 は胴体の前で両手をあわせるジェスチャであり, 胴体と下腕でオクルージョンが発生する. ジェスチャ 3 は右腕を右から左に動かすジェスチャであり, 比較的短いジェスチャである. ジェスチャ 4 は胴体の横で両腕を下から上へ動かし, さらに下へ動かすジェスチャであり, 比較的長いジェスチャである. ジェスチャ 5, 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で, 難しく設定したジェスチャである. 学習では, 各ジェスチャについて 15 シーケンスの追跡結果から平均ジェスチャを作成した. なお, 学習においては, ジェスチャの開始, 終了は手動で与えている. 学習に用いなかった残り 15 シーケンスをテストシーケンスとし, ジェスチャ認識の実験を行った.

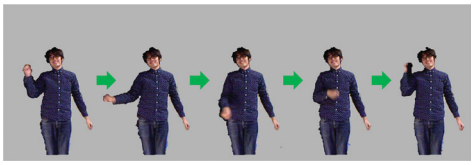
姿勢推定により, 各ジェスチャの初期姿勢を検出したときに, それぞれのジェスチャで認識を開始する. 開始フレームがほぼ同じ姿勢であるジェスチャが存在する場合, ジェスチャ認識は並列に実行されることになる.

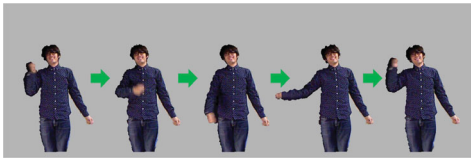

ジェスチャ 1


ジェスチャ 2


ジェスチャ 3


ジェスチャ 4


ジェスチャ 5


ジェスチャ 6

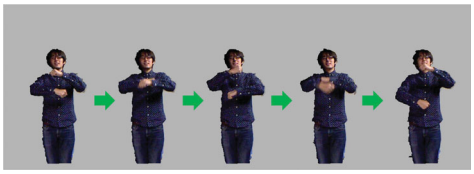

ジェスチャ 7

図 4 登録ジェスチャ

4.2 実験条件

本稿では, 距離・画像センサとして, Kinect for Xbox360 を用いた. 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, 50cm 以上離れた対象について有効な情報が得られる.

追跡に用いた人物モデルを図 5 に示す. これらのモデルは胴体と左右上腕, 左右下腕からなり, それぞれ Kinect で取得したデータから作成した.

図 5 上半身モデル

表 1 追跡条件

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 (σ)	表 2 参照
疎点の数	128
見え画像と奥行き情報の比率 (λ)	0.90

表 2 粗密追跡の初段のサンプル生成に用いた標準偏差
($t_1 \sim t_3$ は mm, その他は deg 単位である)

胴体						肩		肘	
θ_0	ϕ_0	ψ_0	t_1	t_2	t_3	ϕ_S	θ_S	ϕ_E	θ_E
1.0	1.0	1.0	10.0	0.5	7.0	8.0	8.0	12.0	12.0

追跡系は 3 段階からなる粗密追跡法 [7] をベースとして作成し, 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 に示す. なお, ジェスチャの開始, 終了に用いる閾値 τ_1 , ジェスチャ認識に用いる閾値 τ_2 はともに 0.96 と設定した.

4.3 複素相関の有効性の検証

本稿で示した手法は, カメラ座標のベクトルで表現したジェスチャを複素相関により評価を行うものである. これに対し, 胴体座標で表現したベクトルを実相関により評価する比較実験も行い, その認識精度を確認する. なお, 実相関では, 式 (40) のかわりに 式 (42) で得られる相関が閾値を越えた時ジェスチャ g を認識する.

$$C'_R(\boldsymbol{S}, \boldsymbol{S}') = \frac{\boldsymbol{S}^T \boldsymbol{S}'}{|\boldsymbol{S}||\boldsymbol{S}'|} \tag{42}$$

胴体座標で表現することにより, 胴体の姿勢推定が正確に行える場合には, 実相関によってもジェスチャ認識を行える. 複素相関を用いた場合との比較を表 3 に示す. 実相関を用いた場合, 認識率が低くなっているのは, 胴体, 上腕の姿勢推定誤差の影響と考えられる. これに対し, 複素相関を用いる場合には, 姿勢推定誤差の影響をあまり受けず, いずれのジェスチャにおいても高い認識率が得られている.

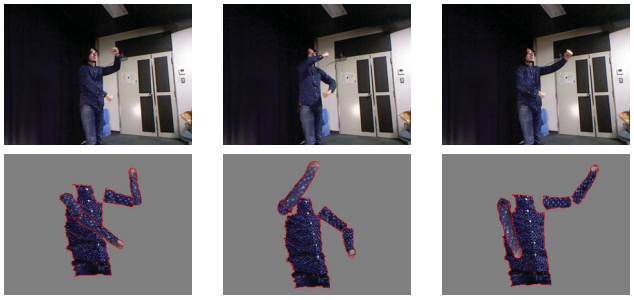


図 6 向きの違う入力画像

表 3 実相関と複素相関 (多段階) による認識率 (%)

	実相関	複素相関			
	1 段階	1 段階	2 段階	5 段階	10 段階
ジェスチャ 1	80%	94%	94%	94%	94%
ジェスチャ 2	100%	100%	100%	100%	60%
ジェスチャ 3	87%	100%	100%	100%	100%
ジェスチャ 4	94%	100%	100%	94%	100%
ジェスチャ 5	100%	100%	100%	100%	94%
ジェスチャ 6	100%	100%	100%	100%	100%
ジェスチャ 7	80%	100%	100%	100%	100%

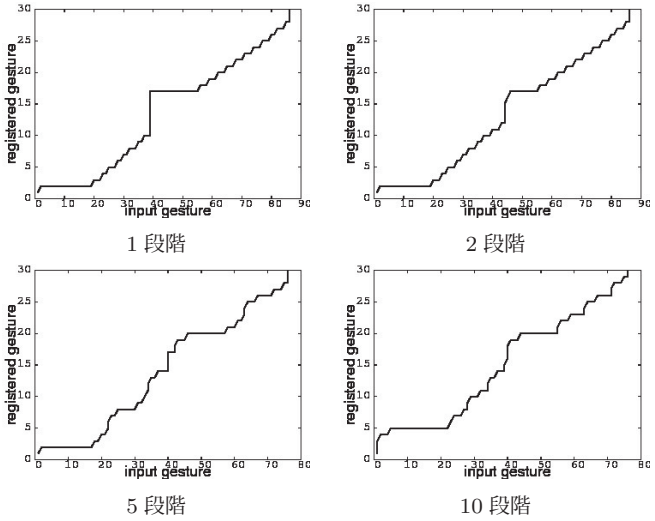
次に, 正面から約 15 度, 30 度, 45 度の各方向を向いて同じジェスチャを行った場合についても実験を行った. この結果, 実相関を用いた場合には, 約 15 度までの簡単なジェスチャしか認識できなかったのに対し, 複素相関を用いた場合には, 45 度までのいずれのジェスチャにおいて, 正面向き同様に精度の良い識別ができた. 図 6 6 に入力動画像の一部と姿勢推定結果を示す. なお, この例は正面から約 45 度の方向を向いており, ジェスチャ 7 を行っているシーケンスである. 以上の結果から, 本稿で示した複素相関によるジェスチャ認識の有効性を確認することができた.

4.4 多段階認識による比較

複素相関を用いる方法について, 3.5 節で述べた多段階認識に関する比較実験を行った. なお, 多段階認識では, 第 L 段階目の認識が確認できた場合にジェスチャ認識が成功したものとする. 実験結果を表 3 に示す.

1 , 2 段階認識では, すべてのジェスチャにおいて高い認識率となった. それに対し, 10 段階認識の場合, ジェスチャ 2 の認識率が 60% となった. ジェスチャ 2 は短いジェスチャであり, 常にオクルージョンが発生しているため推定姿勢に多くのノイズが加わっている. 10 段階認識の場合, 3 フレーム単位で段階的に認識していくため, 短時間の姿勢推定誤差によって段階の進行に乱れが生じ, 認識率が低くなったと考えられる.

また, 提案手法では, ジェスチャの開始, 終了が検出された時点でジェスチャ類似度の評価を行う. そのため, ジェスチャの数が多く, 段階数が少ない場合, たまたま似た姿勢が数フレーム入力されると誤認識を行う可能性がある. こ



1 段階　2 段階　5 段階　10 段階

図 7　入力ジェスチャと登録ジェスチャ 4 の対応

認識開始　中間　認識終了

図 8　入力と推定結果　ジェスチャ 7

のため, 適切に部分ジェスチャへの分割を行うことが望ましい.

図 7 に, ジェスチャ 4 について, 異なる段階数における登録ジェスチャと入力ジェスチャとの対応付け結果を示す. この図で, 縦に一直線になっている場所は入力における 1 フレームが登録の複数フレームに対応していることになる. 逆に, 横に一直線になっている場所は, 複数の入力が登録の 1 フレームに対応している. この場合, ジェスチャ認識には複数ある入力の内もっとも評価値が大きい入力フレームのみが用いられることになる. ジェスチャ 4 には, ジェスチャの途中で動きが止まるところが存在するため, 登録ジェスチャと入力ジェスチャがうまく対応していないように見えるが, この間は姿勢が変わっていないためジェスチャ認識としては高い評価値を得ることができ, 正しくジェスチャを認識できている. また, 実験で使用したジェスチャ 7 の認識開始フレームから終了フレームまでの一部を, それらに対する姿勢推定結果とともに図 8 に示す.

4.5　考察

本稿で提案した手法では, 人物の胴体座標基準にしたジェスチャ認識を取り扱ったが, 人間のジェスチャには, 本手法では特定できないものもある. 例えば, 「指をさす」という動作は, 世界座標を用いないと解析できない. また, より直感的なジェスチャ認識を目指す場合, 左右の腕でジェスチャを独立させる必要があると考えられる. また, 今回示した追跡・認識系はリアルタイム処理で実現できていない. 認識ステップでは, ジェスチャの認識が開始されると 1 つにつき約 0.1 msec である. 一方, 現在の 1 コア CPU 上での実装では, 追跡ステップにおいて 1 フレームにつき約 400 msec を要する. 今後, プログラムの最適化や GPU 実装により実時間処理の実現を目指す.

5.　まとめ

本稿では距離・画像センサを用いた人物追跡によるジェスチャ認識について議論した. 追跡時に得られる姿勢空間パラメータを時系列として保持し, ジェスチャとして登録し, 重力方向まわりの回転に不変な類似度を用いて評価を行った. これにより, 人物の向きが変わっても, 正面を向いている時同様, ジェスチャの認識が行えることを示した. また, 登録ジェスチャを分割し, それぞれで入力を段階的に評価していくことによる認識への影響についても述べた. 今後の課題として, ジェスチャの登録数の増加, 初期姿勢の検出, 複雑なオクルージョンへの処理, 高速化が挙げられる.

参考文献

[1] A.F.Bobick and J.W.Davis, “Real-time recognition of activity using temporal templates,” Proc. 3rd IEEE Workshop on Applications of Computer Vision (WACV), pp.39–42, 1996.
[2] 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, March 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)，no.8，pp.1212-1221，2003.
[5] 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.
[6] 松原康晴，尺長健，“疎テンプレートマッチングとその実時間物体追跡への応用，” 情報処理学会論文誌，vol.46, no.SIG(CVIM11)，pp.60-71，2005.
[7] 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.
[8] 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)，no.8，pp.1652-1661，2005.
[9] 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), 2010.