

## 実時間手指姿勢推定のための ロバストな輪郭画像照合

今井 章博 島田 伸敬 白井 良明  
大阪大学大学院工学研究科電子制御機械工学専攻  
〒565-0871 大阪府吹田市山田丘 2-1  
E-mail:imai@cv.mech.eng.osaka-u.ac.jp

概要: 本論文では単眼カメラで撮影した手指の三次元姿勢の実時間推定方法について述べる. 本研究では形状特徴として輪郭画像を用いている. あらかじめ手指の三次元姿勢パラメータから CG モデルを作成し, その CG モデルから形状特徴を計算し, 三次元姿勢パラメータとセットで保存しておく. 推定は入力の特徴とデータベースをマッチングをすることによって行なわれる.

関節角度が量子化されたモデルと照合を行なう. そのため補正が必要となる. その後でモデルの輪郭画像との相違度を計算し, その相違度と補正の大きさの和を照合度としている. PC クラスタを用いて並列処理を行なうことにより, 実時間での推定が可能である.

また赤外線を用いたカメラを使用し複雑背景下での推定を行なった結果を示す.

## Robust Matching of Silhouette Image for Real-Time Estimation of Hand Posture

Akihiro IMAI, Nobutaka SHIMADA and Yoshiaki SHIRAI  
Dept. of Computer-Controlled Mechanical Systems, Osaka University  
2-1 Yamadaoka, Suita, Osaka 565-0871, Japan  
E-mail:imai@cv.mech.eng.osaka-u.ac.jp

Abstract: This paper proposes a method of real-time estimation of 3D hand postures from monocular images. In our research, we use contour images as the shape feature. CG models are generated from a 3D hand model, and shape feature are computed from CG models stored in database in advance. Estimation is matching an input feature to that in the database.

Because joint angle of model is quantized, contour image is compensated. After the degree of difference between input contour image and one of the models is computed, the degree of matching is evaluated by sum of the degree of difference and amount of the compensation.

The matching is parallelly processed in real-time with PC cluster.

And we estimate using a camera which uses infrared rays in complicated background.

### 1 はじめに

近年, ジェスチャを用いた人と機械の自然な対話を目指す試みが盛んである. キーボードとマウスに代わるインターフェースに手話の認識やバーチャルリアリティなどがある. 複雑なジェスチャを理解するためには詳細な手指の三次元姿勢を利用することが望ましい.

手指の姿勢を得る方法としては手指にマーキングを行なう ([1]) などの方法がある. このような手法はシンプルで実時間での処理に敵しているが, マーキングなどは人間にとって不自然さを与える場合がある.

マーキングなどを行なわない方法に [2][3] がある. この手法では姿勢候補を三次元モデルから作成し, そのシルエットを用いて探索を行なう. しかし推定時に三次元モデルから画像を作成し, 入力画像と直接マッチングを行なうため計算量が大きい.

[4] は入力画像から一次元の特徴を抽出し, データベースとして記憶しておいたモデルの形状特徴とマッ

チングを行うことによって高速な推定を行なっている. しかし情報を一次元におとすため実際の形状の段階と形状特徴の段階で類似度に差がある.

本研究では高い精度を実現するために入力画像とデータベース中のモデルの輪郭画像を照合して最もマッチしたモデルの三次元姿勢パラメータを取り出すことにより推定を行なっている. 関節角度が量子化されたモデルとの照合を行なう. そのため照合の際に補正を行なう. また PC クラスタを用いて並列処理を行い, 実時間で推定を行なっている.

赤外線カメラを用いて複雑背景下での推定も行なう.

### 2 アルゴリズムの概要

あらかじめ三次元姿勢推定パラメータから CG モデルを作成する. その CG モデルから形状特徴を計算し, 姿勢パラメータと形状特徴をセットでデータベー

スとして保存しておく。

実際の推定の際にはまず単眼カメラで人間の手指を撮影し、その入力の手画像から形状特徴を計算する。その入力の形状特徴と作成しておいたデータベース中の形状特徴とを照合し、入力に対して最もマッチしたモデルを求める。そのモデルから三次元姿勢パラメータを取り出すことによって3次元姿勢を推定し、結果をPCに出力する。

実時間で推定を行なうためにPCクラスタを用いて並列処理を行なっている。(アルゴリズムの流れを図1に示す)

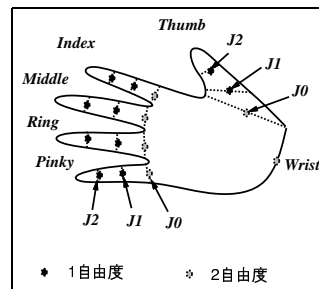


図 2: 三次元モデル

表 1: パラメータの変化

親指 ( $\theta_T$ )	0	20	50	90		
人指 ( $\theta_I$ )	0	20	50	90		
中指 ( $\theta_M$ )	0	20	50	90		
薬指 ( $\theta_R$ )	0	20	50	90		
小指 ( $\theta_P$ )	0	20	50	90		
手首 ( $\theta_{W1}$ )	-150	-120	-90	-60	-30	0
	30	60	90	120	150	180
手首 ( $\theta_{W2}$ )	-90	-75	-60	-45	-30	-15
	0	15	30	45	60	75
	90					

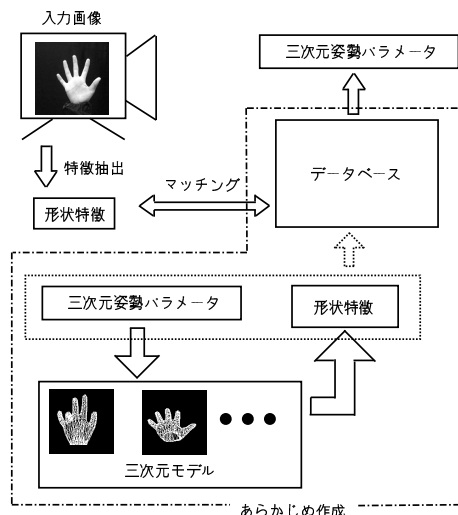


図 1: 推定アルゴリズム

### 3 データベースの作成

三次元姿勢パラメータからCGモデルを作成し、その画像からそれぞれ形状特徴を計算し、三次元姿勢パラメータとセットでデータベースとして保存しておく。以下に三次元姿勢パラメータからのCGモデルの作成について述べる。

#### 3.1 三次元姿勢パラメータからのCGの作成

図2に示すように、本研究で用いる手指の三次元形状モデルは16個のパーツからなり、合計22自由度の関節の角度を変化させることで様々な姿勢の手指を表現できる。

本研究では、簡単のため各指について全ての関節が連動して動き、親指以外のJ0の二つの内の一つの関節角度は一定であると仮定する。各指について1自由度ずつ、掌の向き2自由度、合計7自由度 ( $\theta_T, \theta_I, \theta_R, \theta_P, \theta_{W1}, \theta_{W2}$ ) の三次元姿勢パラメータで手指を表現する。

各指については4段階、 $\theta_{W1}$  は12段階、 $\theta_{W2}$  は13段階に変化させ、合計159744 (=  $4^5 \times 12 \times 13$ ) 枚のCGモデル画像を作成する。作成したCGの例を図3に示す。

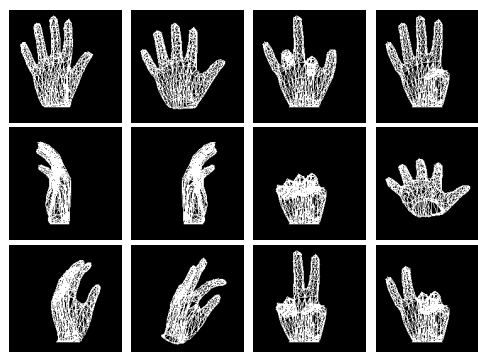


図 3: 作成したCGの例

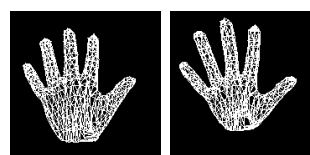
#### 3.2 透視変換に対応したCGの作成

変換式を1に示す。

$$x^* = \frac{fx}{F}, y^* = \frac{fy}{F}, z^* = 0 \quad (1)$$

( $x, y, z$ ): 変換前の座標, ( $x^*, y^*, z^*$ ): 変換後の座標,  $F = f + z$ ,  $f$ : 焦点距離

透視変換前と後のCGの例を図4に示す。



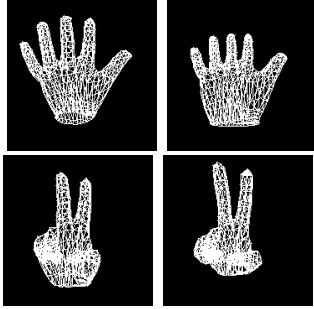


図 4: 透視変換前の CG(左) と後の CG(右)

## 4 照合

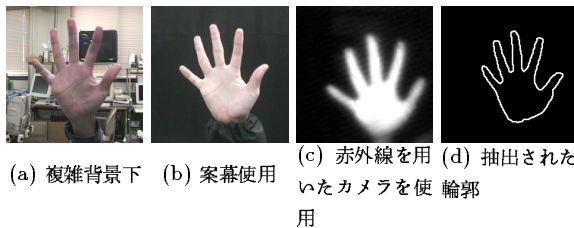
この章では、特徴量の照合について説明する。

形状特徴として手指の輪郭画像を用いる。まずスケール、画面内での回転、平行移動に対して補正を行なう。これは後でアフィン変換の補正量を計算する際に補正量の対象をせん断と二軸の比が等しくない拡大、縮小に限定したいからである。スケール、画面内回転、平行移動の補正の後でアフィン変換を用いて補正を行ない、その後で指が少し曲がっているなどの微量なずれに対して補正を行なう。補正を行なうときには輪郭の山の点や谷の点となっている特徴点の情報を用いる。補正を行なった後に相速度を計算して補正の量と相速度の和を照合度とする。

### 4.1 形状特徴の抽出

本研究では、入力画像に対しては手指の輪郭画像が抽出しやすいように暗幕を背景とし、手首は黒い衣服で覆われているとする。特殊な赤外線を用いたカメラを用いれば暗幕などを使用しなくても手指の輪郭画像を抽出することができる。図 5 に色々な環境下での手指の画像を示す。

図 6 のように輪郭 (モデルの輪郭を  $C$ , 入力の輪郭を  $C'$  とする) 上に  $N$  個の点を等間隔に置く (本研究では  $N = 256$  とした)。各点の位置を計算し、モデルのそれを  $\vec{x}(i)$ , 入力画像のものを  $\vec{x}'(i')$  とする。



(a) 複雑背景下 (b) 案幕使用 (c) 赤外線を用いたカメラを使 輪郭抽出 (d) 抽出された輪郭

図 5: 色々な環境下での画像

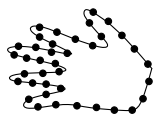


図 6: 輪郭上の点

### 4.2 スケール, 平行移動に対する補正

モデルの面積を  $A$ , 補正前の重心の位置を  $\vec{g}$  とする。また入力面積を  $A'$ , 重心の位置を  $\vec{g}'$ , スケール, 平行移動に対する補正後のモデルのこの点の位置を  $\vec{x}_{scale}$  とする。スケール, 平行移動に対する補正の変換式は以下のようになる。

$$\vec{x}_{scale} = \sqrt{\frac{A'}{A}}(\vec{x} - \vec{g}) + \vec{g}' \quad (2)$$

### 4.3 輪郭上の特徴点の抽出と対応付け

特徴点の山, 谷の点では周辺の点に比べて曲率が大きくなっている。そこで k-curvature の極値を計算するとそれが特徴点である。ただ全ての極値を特徴点としてしまうとノイズや小さい山, 谷の点を特徴点としてしまうので k-curvature の値が  $\frac{5\pi}{12}$  以下, または  $\frac{19\pi}{12}$  以上の極値を特徴点とする。実際の画像での k-curvature とそれから計算した特徴点の様子を図 7 に示す。

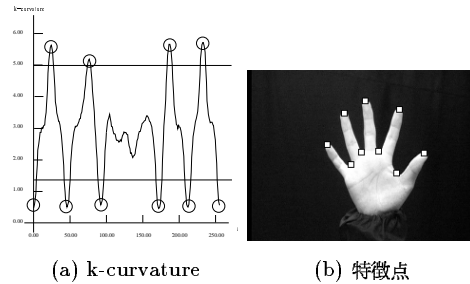


図 7: 特徴点の計算

ここで,  $k$  は指の第 1 関節の長さと同程度の値を使用した。具体的には式 (3) のようになる。

$$k = 0.3 \frac{\sqrt{AN}}{L} \quad k' = 0.3 \frac{\sqrt{A'N}}{L'} \quad (3)$$

ただし,  $L, L'$  は輪郭の長さである。

2つの輪郭画像の特徴点の対応をとるために 1組の対応の基準となる特徴点を決める。まず上記の方法で 2つの輪郭画像に対して特徴点を求め、その点を  $\{F(i)\}, \{F'(i')\}$  とする。このとき対応の基準となる  $i_m$  と  $i'_m$  は式 (4) を最小にする  $i$  と  $i'$  である。

$$\sum \|y(\vec{t}) - y'(\vec{t})\|^2 \quad (4)$$

ただし  $y(\vec{t}), y'(\vec{t})$  は輪郭  $C, C'$  上の点  $F(i), F'(i')$  を始点として右回りにサンプル点 20 点間隔ごとに粗く分割したときの  $t$  番目の位置。(図 8) ただし  $t = 0$  の点は点  $F(i), F'(i')$  とする。

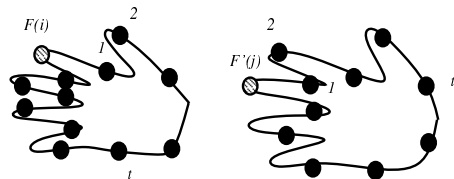


図 8: 粗い分割

入力画像の特徴点についてはモデル画像の特徴点を基準にしてそれに対応する特徴点を入力画像について求め直す。これは図9のようにわずかの差で曲率が閾値に達しなかったためにその極値が特徴点にならず特徴点同士の対応がとれない場合があるからである。

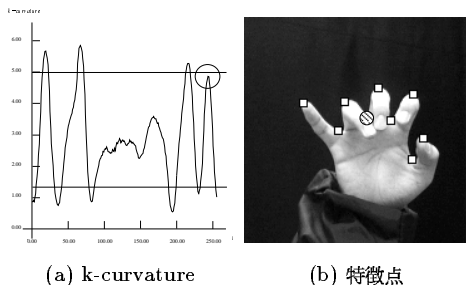


図9: k-curvature と特徴点

$i_m, i'_m$  を基準にして正規化を行なった後の特徴点の番号を  $i_n$ , 点  $F(i_n), F'(i_n)$  に対するサンプル点の中での番号を  $I(i_n), I'(i_n)$  とすると  $I'(i_n+1)$  は  $I'(i_n) + I(i_n+1) - I(i_n) - k \leq I \leq I'(i_n) + I(i_n+1) - I(i_n) + k$  の中で最も k-curvature の値が大きい ( $F(i_n+1)$  が極大値のとき), または最も値が小さい ( $F(i_n+1)$  が極小値のとき) 極値の  $I$  である。この範囲内に極値がない場合は  $k$  の値を2倍, 3倍に大きくしてそのウィンドウの中で極値を見つける。

#### 4.4 画面内回転に対する補正

重心から見た  $F(i_n)$  の角度を  $\theta_{ext}(i_n)$ ,  $F'(i_n)$  の角度を  $\theta'_{ext}(i_n)$ , 特徴点の数を  $N_f$  とする。モデルの輪郭画像のある点の回転の補正後の位置を  $x_{rot}$  とすると変換式は次のようになる。

$$x_{rot} = \begin{bmatrix} \cos(\theta_{rot}) & -\sin(\theta_{rot}) \\ \sin(\theta_{rot}) & \cos(\theta_{rot}) \end{bmatrix} x_{scale}$$

ただし,

$$\theta_{rot} = \frac{1}{N_f} \sum (\theta'_{ext}(i_n) - \theta_{ext}(i_n)) \quad (5)$$

である。

#### 4.5 アフィン変換による補正

対応がとれた特徴点の位置を用いてアフィン変換のパラメータを推定し, そのアフィン変換によって補正を行なう。アフィン変換で起こる変換は平行移動, 回転, 拡大縮小, 反転, せん断である。これを用いてマッチングを平行移動, 回転, 拡大縮小に対して不変なものとし, せん断による補正を行なう。変換前の座標を  $(x_{rot}, y_{rot})$ , 変換後の座標を  $(x_{affin}, y_{affin})$  とするとアフィン変換は次式で表される。

$$x_{affin} = ax_{rot} + by_{rot} + c \quad (6)$$

$$y_{affin} = dx_{rot} + ey_{rot} + f \quad (7)$$

変換式の各パラメータは特徴点の位置について最小自乗法を用いることによって推定する。 $a, b, c, d, e, f$  は次の式を最小にする値である。

$$\sum_{i_n} ((x'(I'(i_n)) - (ax_{rot}(I(i_n)) + by_{rot}(I(i_n)) + c))^2 + (y'(I'(i_n)) - (dx_{rot}(I(i_n)) + ey_{rot}(I(i_n)) + f))^2) \quad (8)$$

変換によってあまり輪郭が変形し過ぎると誤推定の原因となるので最終的な照合の際にこのアフィン変換による次式の変形の大きさを考慮に入れる。

$$\sum \sqrt{|x_{rot}(I(i_n)) - x_{affin}(I(i_n))|^2} \quad (9)$$

#### 4.6 輪郭位置の微量なずれに対する補正

対応がとれた特徴点を用いて補正を行なう。補正ではアフィン変換後のモデルの輪郭画像を点  $F(i_n)$  の位置が点  $F'(i_n)$  の位置に一致するようにモデルの輪郭画像を変形させる。変換前の  $I(i_n)$  と  $I(i_n+1)$  の間の点の位置を  $x_{affin}(I)$ , 変換後の位置を  $X(I)$  とすると  $I(i_n)$  と  $I(i_n+1)$  の間の点は式(10)のように補間する。

$$\begin{aligned} \vec{X}(I) = & x_{affin}(I) + \vec{x}'(I(i_n)) - x_{affin}(I(i_n)) \\ & + \frac{I - I(i_n)}{I(i_n+1) - I(i_n)} \\ & \times (\vec{x}'(I(i_n+1)) - \vec{x}'(I(i_n))) \\ & - x_{affin}(I(i_n+1)) + x_{affin}(I(i_n)) \end{aligned}$$

照合の際に次式の変形の大きさを考慮に入れる。

$$\sum \sqrt{|x_{affin}(I(i_n)) - \vec{X}(I(i_n))|^2} \quad (11)$$

#### 4.7 手首部分の補正

入力画像では手首の部分は腕が少し見えていたり, 逆に手の平が少し隠れていたりすることがあり, 手指が同一の形状でも手首の部分の形状は異なることがある。(図10) そのため輪郭中で手首の部分の両端の点(図11)を k-curvature の極値と同様に特徴点として扱う。さらにその部分では残差を計算せず, 評価に用いない。

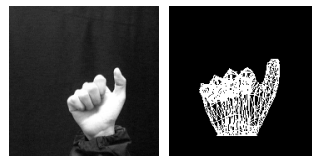


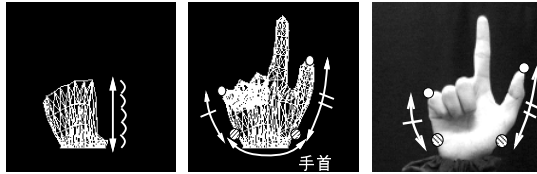
図10: 手首形状の違い

手首での特徴点を求める際にはまずモデルで特徴点を求め, その後にそのモデルの特徴点に対応する入力の特徴点を求める。

モデルでの特徴点の求め方を以下に述べる。まずモデルの手の画像の手の平の部分だけを求める。(図11(a)) 元々モデルは手の平や指の各パーツに分かれていてそれらを組み合わせることによって手のモデルが出来上がるのでそのうちの手の平の部分だけの画像を作成することは簡単である。作成した手の平の高さの5分の

1を計算し, 全体の手の画像で下のその5分の1の高さに相当する部分の輪郭を手首として扱い, その両端の点を特徴点とする.

モデルの手首の特徴点に対して入力画像では  $I = I'(i_n) + I(i_n + 1) - I(i_n)$  の点を特徴点とする. ただし  $I'(i_n) + I(i_n + 1) - I(i_n) - k \leq I \leq I'(i_n) + I(i_n + 1) - I(i_n)$  の範囲に入力画像の k-curvature が極大値を持つ場合はその中で最も k-curvature の値が大きい極値をもつ点を入力画像の特徴点とする.



(a) CG の手の平 (b) CG (c) 入力

図 11: 手首の領域

これまで述べてきた補正を行なって輪郭画像を重ね合わせた例を図 12 に示す.

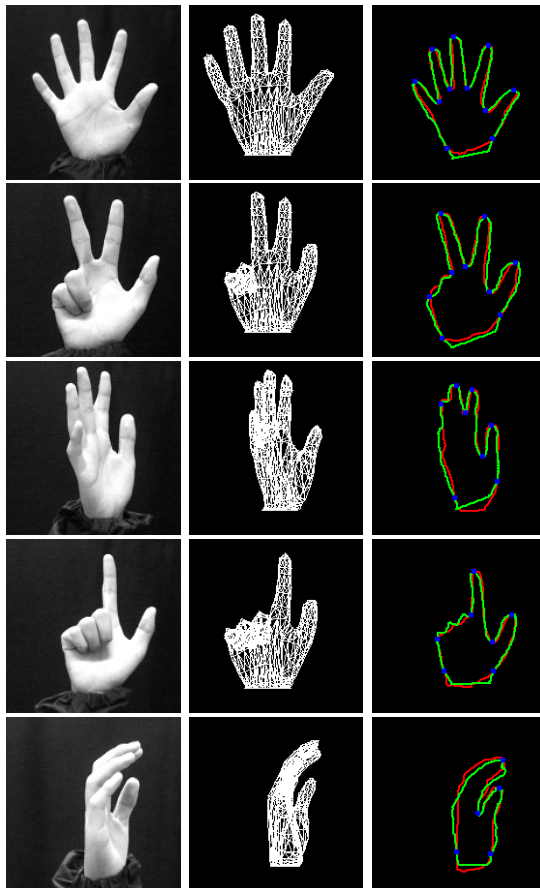


図 12: 補正後の輪郭画像の重ね合わせ

#### 4.8 相違度

入力の輪郭画像と補正後のモデルの輪郭画像から残差を計算し, その残差から補正後の二つの輪郭画像の相違度を計算する. 図 13 のようにモデルの輪郭画像 (輪郭線を  $C$  とする) と入力の輪郭画像 (輪郭線を  $C'$  とする) を重ね合わせる.

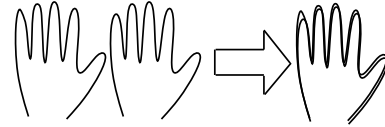


図 13: 輪郭画像の重ね合わせ

図 14 のように  $C$  上に等間隔に置かれた点を  $P_i (i = 1, \dots, N)$  とする. 点  $P_i$  を通り  $C$  と垂直に交わる直線を  $l$ ,  $l$  と  $C'$  との交点を  $P'_i$  その位置を  $X'(i)$  とすると点  $P_i$  と点  $P'_i$  の距離  $r(i)$  は次式のようになる.

$$r(i) = \sqrt{\|X(i) - X'(i)\|^2} \quad (12)$$

$r(i)$  を手の大きさで正規化したものを  $\rho(i)$  とすると,

$$\rho(i) = \frac{r(i)}{\sqrt{A'}} \quad (13)$$

2 つの輪郭画像の残差は次式のようになる.

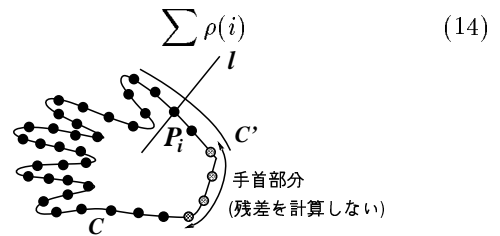


図 14: 距離

4.7 で述べたように手首の部分では残差を計算しない. 図 15 に手首で残差を計算した場合としない場合の例を示す. この例では入力画像の手首のあたりで腕が少し見えているために手首の部分 ( $i=100$  前後) で残差を計算すると残差の値が大きくなってしまふ. そのため手首の部分で残差を計算しない. この補正によって手首での形状の変化に対して柔軟に対応できる照合を行なうことができる.

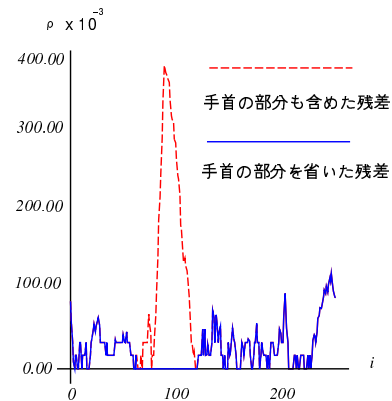
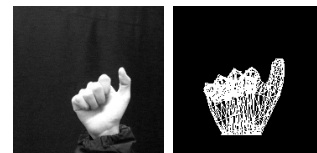


図 15: 手首での残差

図 16 のような画像を用いて残差の評価について調べる。

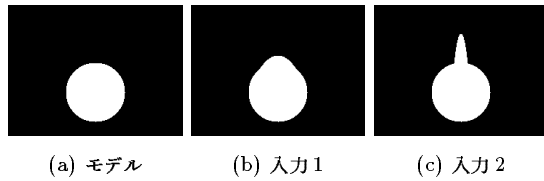


図 16: テスト画像

モデルと入力 1, 入力 2 を比較したときの残差はそれぞれ 5.89, 5.57 とほぼ同じである。しかし人が判断した場合、モデルに対して形として近い画像は入力 1 である。そこで残差の差分の絶対値を計算し、それを相違度としてマッチングを行なうことを考える。残差の差分は式 (15) のようになる。

$$r_{dif}(i) = ||r(i+1) - r(i)|| \quad (15)$$

差分を用いて入力 1, 入力 2 を比較するとその値はそれぞれ 0.69, 2.10 となる。差分を用いた方が人の判断に近い評価ができています。グラフを図 17 に示す。

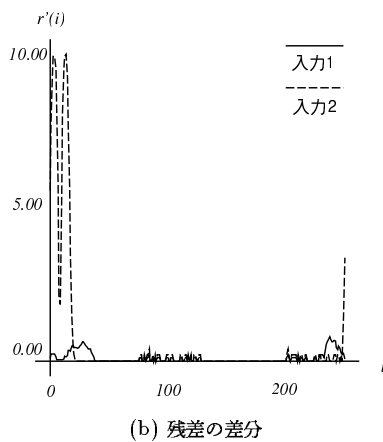
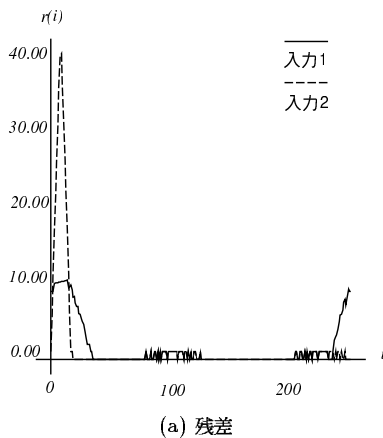


図 17: 残差と残差の差分

## 4.9 照合度

これまでに計算してきた補正量と補正後の相違度との和を特徴量としてマッチングを行なう。照合度は次の式の値となる。

$$\begin{aligned} & \sum (\sqrt{|x_{rot}(I(i_n)) - x_{affin}(I(i_n))|^2} \\ & \quad + \sqrt{|x_{affin}(I(i_n)) - \bar{X}(I(i_n))|^2}) \\ & + 0.05 \sum \rho_{dif}(i) \end{aligned} \quad (16)$$

相違度は差分をとっているので補正量とそのままを合わせることはできない。本研究では補正量に重み (0.05) を掛け合わせてからたし合わせて特徴量としている。

## 5 実験

### 5.1 実時間での推定

PC クラスタを用いて並列処理を行なって実時間で推定する。

### 5.2 システム

PC クラスタを用いた並列処理のシステムを図 18 に示す。PC0 で入力画像をキャプチャし、特徴抽出までを行なっている。計算した形状特徴を PC1 ~ PC10 に送信し、ここでデータベースとのマッチングを行なっている。マッチング作業が終了すると推定結果を PC0 に送信し、PC0 で結果の出力を行なう。これを実時間で繰り返す。

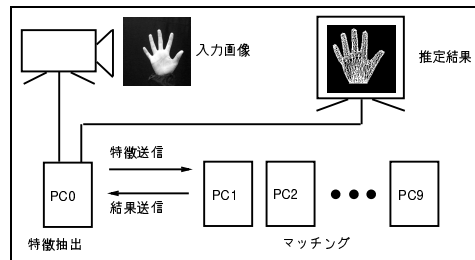


図 18: PC クラスタを用いた並列システム

むだ時間を減らすために前フレームのマッチング処理中に次フレームの画像の取り込みと特徴抽出を行なっている。タイムチャートを図 19 に示す。

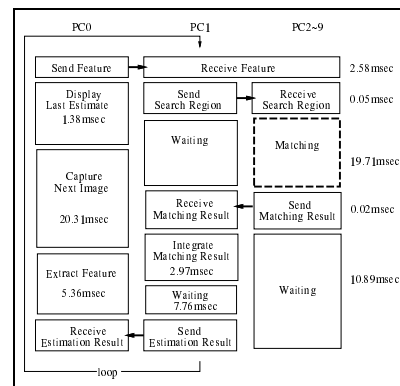


図 19: タイムチャート

### 5.3 推定結果

動画像を用いて実験し, 正しく推定されたときの画像の一部を図 20 に示す. 左が入力で, 右が出力である.

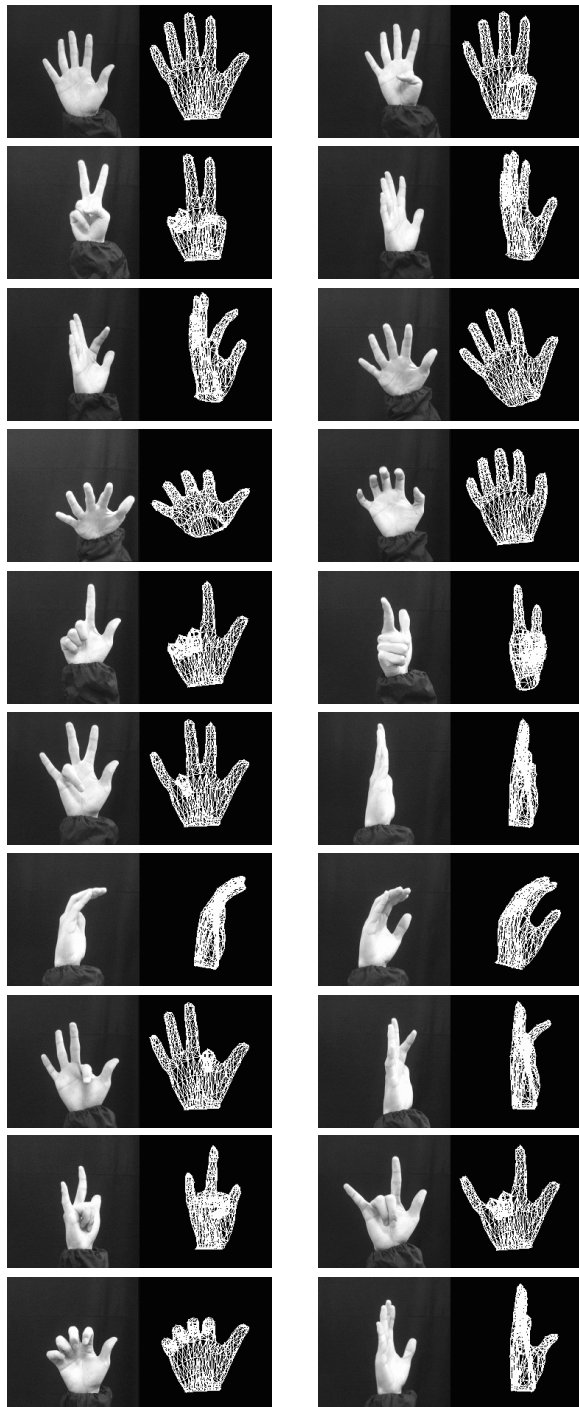


図 20: 正しい推定の例 1

照合の際に用いたモデルは関節角度が量子化されているので照合の結果得られたモデルの関節角度に対してスムージングを行なう. スムージングは 5 フレーム間の照合結果の関節角度の平均を計算する. 入力時系列画像とスムージングの前と後の推定結果, 関節角度のグラフを図 21 に示す.

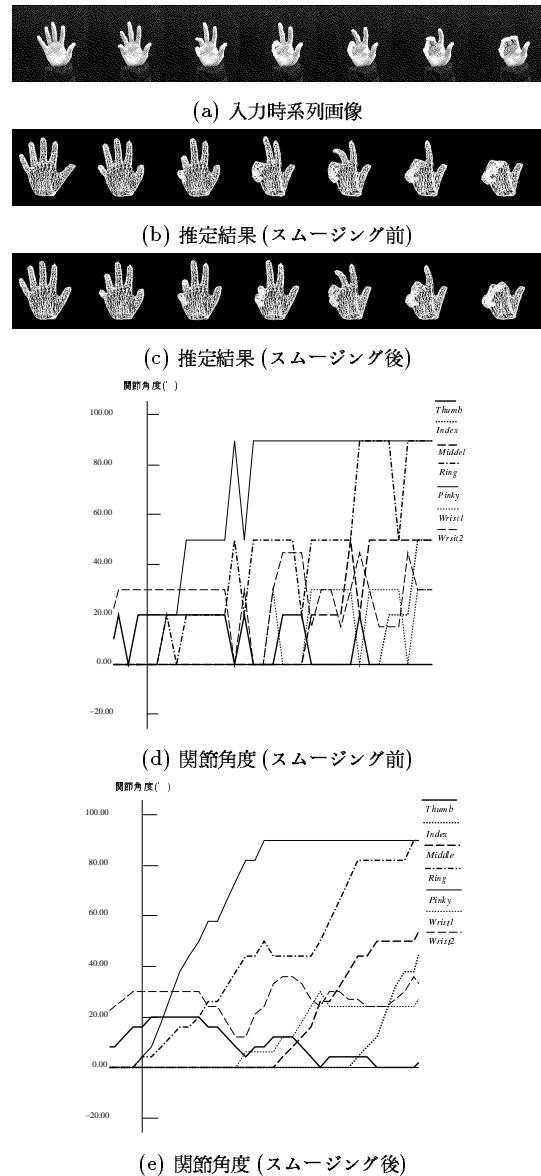


図 21: スムージング

### 5.4 誤推定の例と問題点

実験中, いくつか誤推定を起こすことがあった. その例を示す.

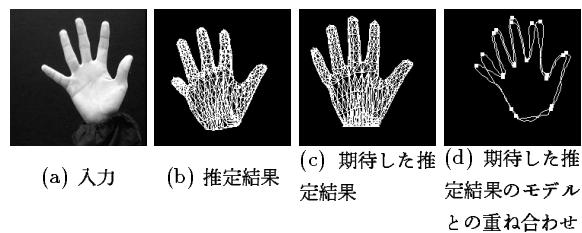


図 22: 誤推定の例 1



(a) 入力 (b) 推定結果 (c) 正しい推定

図 23: 誤推定の例 2

例 1 については入力画像に対して小指が少し曲がったモデルが推定結果として得られた。三次元姿勢パラメータの中の指の曲がる自由度が一方しかないため入力画像の形状と完全に一致するモデルが存在しない。関節と垂直な方向へ曲がった指の変化に対しては微量なずれとして補正を行なったが、その補正の量が大き過ぎるため小指が曲がっていないモデル(図 22(c))とは照合度が大きくなってしまった。

例 2 については入力では中指と薬指の間にエッジが存在するが、正しいモデルでは隠蔽が生じエッジが存在しない。そのため入力と正しいモデルが異なる輪郭となりこのような誤推定が生じた。

## 6 赤外線カメラを使用した推定システム

複雑背景化で推定するために赤外線を利用したカメラを用いて推定を行なう。

### 6.1 カメラ

使用するカメラは図 24 のカメラである。これは東芝のモーショントラッカーというカメラで、赤外線を用いてカメラとの距離が小さい物体を強調して撮影することができる。(図 5) このカメラを用いて複雑背景下での推定を行なう。

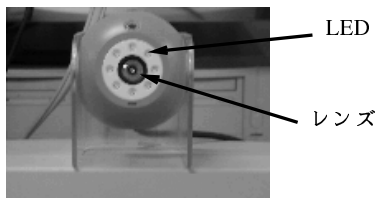


図 24: カメラ

### 6.2 透視変換

使用するカメラはその性能上被写体との距離をあまり大きくとることができない。そのため入力の輪郭画像には透視変換の影響が生じる。よって CG モデルは 3.2 で作成した透視変換に対応したものをを用いる。本来はカメラと被写体との距離によって変換は異なるが、このカメラを用いた場合カメラの焦点と被写体との距離はほぼ一定(0.19m)である。よって CG モデルを作成する際は  $F = 0.19$  とした。

### 6.3 推定結果

このカメラを使用して推定した結果を図 25 に示す。

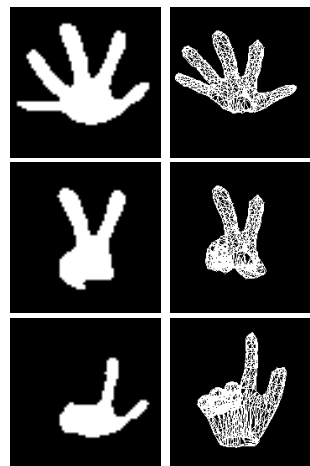


図 25: 推定結果

## 7 おわりに

本研究では単眼カメラで撮られた手指の画像からその三次元姿勢を推定する手法について述べてきた。マッチングは輪郭画像の照合によって行なった。また種々の補正を行なうことによって微量のずれや手首形状の違いに対してより強固な推定を行なうことができた。しかし誤推定も 5 節のように生じている。

この問題に対しては次の二つのことを今後の課題として考えている。

一つは個々の CG に対して許容できるずれの大きさをあらかじめ決めておいて、そのモデルを学習させておいてそれを用いて実際のマッチングの際に評価するというものである。

もう一つは形状特徴として輪郭画像のみを用いるのではなく、輪郭の内部のエッジを計算しその情報を用いて輪郭画像だけでは推定しにくい姿勢を推定するというものである。これらのことを用いればさらに精度は高くなると思われる。

## 謝辞

本研究で用いている CG モデルを提供してくださった大阪大学大学院工学研究科ヒューマンインタフェース講座の岸野教授、北村助教授に深く感謝致します。

## 参考文献

- [1] 吉野和芳, 川嶋稔夫, 青木由直: “手形状遷移に基づく手話単語推定”, in MIRU'96, Vol.II, pp.43-48, 1996.
- [2] 亀田能成, 美濃導彦, 池田克夫: “シルエット画像からの間接物体の姿勢推定法,” 電子情報通信学会論文誌, D-II Vol.J79-D-II, No.1, pp.26-35, 1996.
- [3] 島田伸敬, 白井良明, 久野義徳: “確率に基づく探索と照合を用いた画像からの手指の 3 次元姿勢推定,” 電子情報通信学会論文誌, D-II Vol.J79-D-II, No.7, pp.1210-1217, 1996.
- [4] 木村光佑, 白井良明, 島田伸敬: “CG 検索に基づく単眼シルエット画像を用いた三次元手指姿勢の推定”, MIRU2000, Vol.II, pp.145-150.2000.