

自己増殖型 SOM を用いた 画像データベースからの手指形状の実時間推定

谷本 貴 頌[†] 星野 聖^{††}

著者らの研究グループは、手画像および角度データから得られたデータベースから類似画像を検索する手法により、高速なヒト手指推定システムを構築してきたが、データベース規模が増大した場合、処理時間の増加が予想される。そこで本研究では、自己増殖と自己消滅を含んだ自己組織化により大規模データベースを効率的に配置する方法を提案した。手画像が高速カメラを備えたシステムへ入力されると、各画像を高次局所自己相関に基づく特徴量によりデータベースのデータと比較することで推定を行うシステム構成とした。CPU クロック周波数 2.8 GHz の PC を用いた実験の結果、各画像につき 20 ms 以内の処理時間で、高精度に推定が可能であった。

Real Time Posture Estimation of Human Hand Using Database with Self-organized Map

TAKANOBU TANIMOTO[†] and KIYOSHI HOSHINO^{††}

The hand posture estimation system by searching a similar image from a vast database, such as our previous research, may cause the increase of processing time, and prevent realtime controlling of a robot. In this study, therefore, the authors proposed a new estimation method of human hand posture by rearranging the vast database with the Self-Organized Map (SOM) including self-proliferation and self-disappearance, which enables two-step searches of similar image with short period of processing time within small errors. The experimental results showed that our the system exhibited good performance with high accuracy within processing time of 20 ms or less for each image input with a 2.8 GHz CPU personal computer.

1. はじめに

人間の手指は多関節構造を持つため形が複雑に変化し、掌や各指によって隠蔽領域ができることから、形状推定が非常に難しいといわれている。これまでに行われた研究としては、上田ら¹⁾は複数カメラを用いて手指形状の Voxel モデルを作成し、それによる手指形状を推定している。島田ら²⁾は複数 PC の並列処理によって実時間推定をした。同グループは特殊な装置を利用しない単眼視カメラのみの推定も研究している³⁾。亀田ら⁴⁾は手指の三次元形状モデルを用いてトップダウンにモデルの投影を作成し、画像のシルエットとの被覆領域から推定を行った。

これらの研究では、特殊な機材を必要とするか、モデルを利用したマッチング処理によって形状推定が行われるが、その処理に時間がかかり、実時間での処理

は実現されていなかった。しかし、人間の手指形状推定を考えた場合、モデルマッチングの処理も行っているが、同時に、すでに見た手指形状がデータベースとして保存されており、そこから現在の手指画像と最も近い画像を検索する手法も行っている、とも考えられる⁵⁾。著者らの研究グループは、モデルマッチングによる推定ではなく、データベースからの類似画像の高速検索によって形状推定を行い、実時間推定を試みている。ここでは、Odo ら⁶⁾が手指形状のジェスチャ認識として高い推定結果を出している高次局所自己相関関数⁷⁾を使い、画像の局所的なベクトルの各々の総和を表現して類似画像の高速検索を行った。

ところが、画像から得られた特徴量とデータグループによって取得した角度とともにデータベースとして保存することでデータベースは構築できるが、データが増大すれば検索時間も比例して増大する。類似画像の高速検索のためにデータベースを自己組織化させる手法としては、Kubo ら⁸⁾がすでに行っている。しかし、この方法では各クラスに大量のデータが存在した場合、もしくは少量のデータしか存在しなかった場合

[†] 筑波大学

University of Tsukuba

^{††} 筑波大学/JST

University of Tsukuba/JST

などに検索時間の偏りが生じてしまい、本研究のような実時間での推定には適用が難しい。また、関連のない画像群を自己組織化したものであり、すべて手の画像という類似画像群を自己組織化するためには、2次元空間での自己組織化では各クラスのデータの類似性が不十分となる。

そこで、本研究は関節角度情報を用いて複数次元で自己組織化させることによりクラスに分類し、クラスの代表特徴量を利用した2段階検索を採用した。さらに、自己組織化時に各クラスに自己増殖消滅アルゴリズムを加えることで、クラスに属するデータ数の均一化を図った。これによりクラス数の増大が発生するが、実時間での連続した画像の推定においては、クラスどうしのつながりを利用した探索空間の制限が使用できるために、高精度な手指形状の実時間推定を実現できると考えた。

2. システム構成

2.1 1次データベース作成

2.1.1 画像と角度情報のセットの取得

データベース作成のために、画像と角度を組にして取得する必要がある。モノクロ高速度カメラ (MEGAPLUS 社製, ES310/T) を使い、解像度を 320×240 画素として、手指が画面内に十分な大きさで映っている状態でデータベース用のヒト手画像を撮像した。また手指関節角度はデータグローブ (Virtual Technologies 社製, CyberGlove) を用いることで値を取得した。このデータグローブは 18 自由度を持ち、24 種の角度情報を出力する。

この角度情報の内訳は各指 3 関節に各指間の角度 4 種、親指の回旋、手首の角度 3 種と使用されない 2 種である。このうち各指の DIP 関節は PIP 関節より算出されるため、18 自由度となる。また、手首の 3 種についてはモーションキャプチャと組み合わせなければならないため、本実験では使用していない。実際の人の手の自由度とデータグローブの手の自由度には違いがあるが、角度推定の定量的評価のために用いた。

また、本システムは画像どうしのマッチングを行うものであり、将来別の計測機器を用いることも可能である。ただし、データセット作成時には、データグローブを装着した際にデータグローブ固有の画像特徴が出てしまうため、それを消すために白手袋を着用した。これによりヒトの手と色やテクスチャが異なり、形状も若干違うものとなるが、後述の特徴量化により一般の手画像として扱うことが可能である。また、このデータベース作成では背景画像が存在しても作成はで



図 1 データベース作成時画像の例

Fig. 1 An example of image for constructing database.

きるが、推定に影響を及ぼさないよう背景の影響を極力少なくするために、背面に黒色スクリーンを用いた。それぞれの白手袋画像と、同期させて取得したデータグローブからの手指角度を組にして保存することで 1 次データベースを作成した。データベース作成時の画像として図 1 を示す。

2.1.2 輪郭抽出処理

2.1.1 項により得られた 1 次データベースの画像情報から、推定に必要な特徴量を取り出す処理を行った。処理としては背景除去、解像度変更、輪郭抽出を行った。

まず、画像情報から背景除去を行った。あらかじめ取得しておいた背景画像を用いて、対象画像との差分を取得した。その差分が閾値以上の場合は対象画像の画素をそのまま使い、以下ならば 0 に置き換えた。次に背景を除去した画像を用いて、解像度を変更した。今回取得した画像は 320×240 の画素値を持つ画像だが、これをそのまま用いて各種画像処理を施して形状推定するには、計算時間がかかる。また人間が推定する際には、今回用いた画像のサイズより小さくても推定ができるものと考え、 64×64 のサイズに解像度を落として使用した。解像度変更後の画素値は次式により得られる。

$$gr(i, j) = \frac{1}{r} \sum_k \sum_l go(i * 320/64 + k, j * 320/64 + l) \quad (1)$$

ここで $gr(i, j)$ は解像度変更後の i 行 j 列番目の画素値。 $go(i, j)$ は解像度変更前の i 行 j 列番目の画素値。ここで、縦についても 320 画素で計算しているが、これは変更後の画素値を 64×64 にしているため縦横比を合わせるためである。また、 k と l は解像度

変更前のそれぞれの区画内の行と列にそれぞれ対応する r は $k * l$ である。

最後に、輪郭抽出を行った。推定時とデータベース作成時では、環境光や人の肌の色の違いなどがある。ましてやデータベース作成時には白手袋を使用しているため、微分処理のようなエッジ抽出を行っても、指から背景への画素の変化の領域がその時々で値や幅などが違ってしまいうため、データベースの値と推定時の値とのマッチングが困難となる。ほかにも、影と手の内部に生まれるエッジとの区別がつけにくくなる。そこで輪郭抽出を行い、曖昧性のある手の内部のエッジを使わずに輪郭部のみのエッジの幅を 1 画素にし値を二値化することで、データベースの手から得られる情報と推定時に得られる情報との差を少なくし、環境の影響などを排除することで推定時の誤差を減少させる方式を採用した。

2.1.3 特徴量化

本研究では、Otsu ら⁷⁾ が提案している高次局所自己相関関数を使用した。高次局所自己相関特徴は画像の認識や計測のために有効な特徴であり、次式で定義される高次相関関数を、参照点とその近傍について計算したものである。

$$x^N(a_1, a_2, \dots, a_N) = \int f(r)f(r+a_1) \cdots f(r+a_N)dr \quad (2)$$

ここで x^N は N 次元での r 点近傍の相関関数である。一般に自然画像を処理対象とする場合、対象点まわりの画素が重要なため、係数 N を 2 次までとした。また、平行移動による等価なものを除いて、図 2 で示す 25 種類の特徴量で表現できる。同図の 1 の部分の画素値を掛け合わせることで、各種類の特徴量 x^N が得られる。ただし No.1 から No.5 までは、No.6 以降の特徴量に比べスケールが小さいので、それを合わせる必要がある。No.2 ~ No.5 までは参照点の画素をさらに掛け合わせ、No.1 は参照点の画素の 2 乗を掛けることで他の特徴量に一致させた。

本研究では輪郭画像の画面を縦 8 分割、横 8 分割にし、計 64 分割した分割画面に対して画素すべてを参照点として、高次局所自己相関関数によって 25 の特徴量に次元低減を行った。すなわち、全体の画像で考えれば $25 * \text{画面分割数} (= 64)$ の特徴量で記述することになる。

2.2 自己増殖型 SOM による 2 次データベース作成

2.1 節で作成されたデータベースをそのまま検索に用いると、データベースの増大とともに検索時間も同

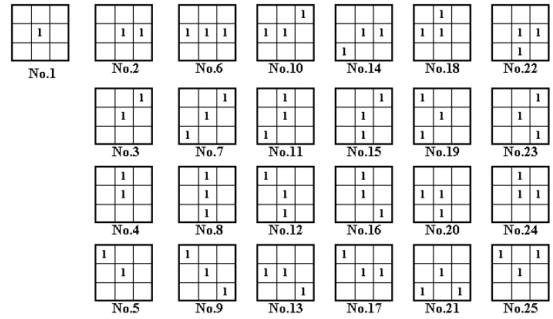


図 2 高次局所自己相関関数

Fig. 2 Higher-order autocorrelation.

時に増大してしまう。そのため、データベースのデータを関節角度が近いもので集めることで検索空間を狭められるようにした。

本研究ではデータを関節角度が近いデータどうしを集めるために、自己組織化マップを用いてクラスタリングを行った。各データは角度と特徴量を持っているが、推定時に検索に使用するものは特徴量である。しかし、特徴量は角度のうえでは離れていても特徴量で見れば近いものが存在してしまい、自己組織化時に特徴量で分類を行うと角度が離れたものが同じクラスに含まれる可能性がある。一方角度のうえでは近いデータであっても、特徴量のうえでは離れてしまうものも少数存在する。しかし、これらについては後述する近傍クラスの検索でカバーできると判断し、本研究では特徴量での自己組織化よりも角度を使用しての自己組織化を採用した。

初めに、初期クラスとして代表角度、代表特徴量、近傍クラス情報を持ったクラスを作成した。初期の角度と特徴量は 0 から 1 の範囲で乱数を用いた。近傍クラス情報は各クラスの角度間の距離をユークリッド距離で算出し、その距離の近いクラスを近傍クラスとして、そのクラス番号とした。なお、データベースの規模や PC の処理性能に左右されるが、本実験ではヒューリスティックに検討し、20 番目に近いクラスまでを近傍クラスとした。次に、1 次データベース中のデータを用い、データとクラス間の角度の距離を算出し、最も近いクラスを選出した。以下はこのクラスを最近傍クラスと呼ぶ。また、使用したデータは最近傍クラスの所属データとする。最近傍クラスの代表角度、代表特徴量をデータに近づけるように次式を用いて更新を行った。

$$\begin{aligned}
CA_{ij}(n) &= \\
&CA_{ij}(n-1) - \alpha(CA_{ij}(n-1) - DA_{rj}) \\
CF_{ij}(n) &= \\
&CF_{ij}(n-1) - \alpha(CF_{ij}(n-1) - DF_{rj})
\end{aligned} \quad (3)$$

ここで CA_{ij} はクラス i の代表角度 j , DA_{rj} はデータ r の角度 j , CF_{ij} はクラス i の代表特徴量 j , DF_{rj} はデータ r の特徴量 j , α は学習係数とした。なお, α は本実験では 0.1 から 0.0001 までヒューリスティックに変化させた。次に, 最近傍クラスの近傍クラス情報に含まれるクラスについても同様の更新を行った。ただし最近傍クラスの学習係数よりも低い学習係数とした。これを 1 次データベース中のデータすべてについて行った。以上の作業をクラスの代表角度, 代表特徴量の動きがほぼなくなるまで繰り返し, 自己組織化を行った。

自己組織化を行ったデータベースをそのまま用いても, 検索の絞込み, 高速化はある程度実現できる。しかし, そのまま用いた場合は各クラスに含まれるデータ数に幅があり実時間推定を行うにはクラス間で処理時間の偏りが起きてしまう。そのため自己組織化時に自己増殖, 消滅を導入することでそれを解消した。

すべてのデータの所属クラス決定後, 各クラスの所属データ数が理想データ数の 2 倍を超えたものについては同一クラスを複製した。また, 所属データ数が理想データ数の 4 分の 1 以下のものについては該当クラスを削除した。これによりクラスに属するデータ数がばらつくことなく一定範囲内で収まるため, 推定時にどのクラス内のデータを用いて検索を行っても, 処理時間が一定時間内で収まった。なお, このときの理想データ数は PC の処理性能やデータベース規模により変化するが, 本実験では 40 と設定した。この自己増殖, 消滅アルゴリズムを導入した場合, 通常の自己組織化では不変であるクラス間の関係も変化が起るため, クラス間の関係も作り直さなければならない。そこで, クラスの複製削除が行われた場合, 初期化時と同様の手法で近傍クラス情報を作り直した。

以上の操作で得られたクラスの代表角度, 代表特徴量, 近傍クラス, 所属データと 1 次データベースをまとめて 2 次データベースとした。以後, 2 次データベースを単にデータベースと記す。

2.3 手指角度推定

推定時には, 1 次データベース作成時と同様に高速カメラで画像を取得した。同画像に対し, 1 次データベース作成時と同様の画像処理を施し特徴量を求めた。この特徴量とデータベースの特徴量を後述の処理によ

り比較し, 最も近いデータと組となる角度を推定結果とした。

類似度の推定のため, データベースのすべてのクラスに対し, 特徴量による距離算出を行った。なお 2 回目以降の推定については, 前回推定データが所属するクラスの近傍クラスについてのみ距離算出を行うことで, 処理の高速化を行った。距離の算出は単純なユークリッド距離で行った。距離は次式により算出した。

$$E_r = \sum_{i=1}^{25 \times n} (x_{ri}^N - x_{ii}^N)^2 \quad (4)$$

ここで, x_{ri}^N はクラス r の高次局所自己相関関数による代表特徴量 i , x_{ii}^N は時刻 t での高次局所自己相関関数による特徴量 i であり, 2 次の高次局所自己相関関数であるので N は 2 とする。 E_r を最小にするクラス r を時刻 t の最近傍クラスとした。この最近傍クラスの所属データについて画像から得られた特徴量との距離を式 (4) により算出した。最も距離の小さいデータの角度を現在の推定角度とした。ただし, 角度の大幅に違うものの中に特徴量の近いデータが紛れ込むことがあるので, 前時刻の推定角度を用いて角度制限を行った。

前時刻の推定角度との距離が近い推定候補について, 次式で示す評価関数 A_p が規定値を超える場合, 推定候補から除外した。

$$A_p = \sum_{i=1}^{24} (ang_{i(t)} - ang_{i(t-1)})^2 \quad (5)$$

ここで $ang_{i(t)}$ は時刻 t の i 番目の角度データである。本実験では A_p は各指 30° を超えた場合に機能するように, 実験的に決定した。以上の操作により現在の推定角度を求めた。

3. 推定実験

3.1 実験方法

本研究の有効性を確かめるため, 実画像に対し推定を試みた。被験者はカメラ前方 1 m 程度の位置に手をかざし, 掌をカメラ側に向けた条件で, 自由に指を動かした。本研究では実時間で推定が最低 30 fps での動作と考えたが, 本システムはそれ以上の速度を見込めたため, カメラからの画像の取得は 60 fps で取得し, システムを動作させた。カメラ画角の範囲内に収まるなら, 手が上下左右ならびに前後に多少移動することは許容した。なお, 画面中の手の位置検出を行い, 手領域を抽出後に本システムを適応することで上記の条件は取り払うことが可能である。また, 理想環境下で

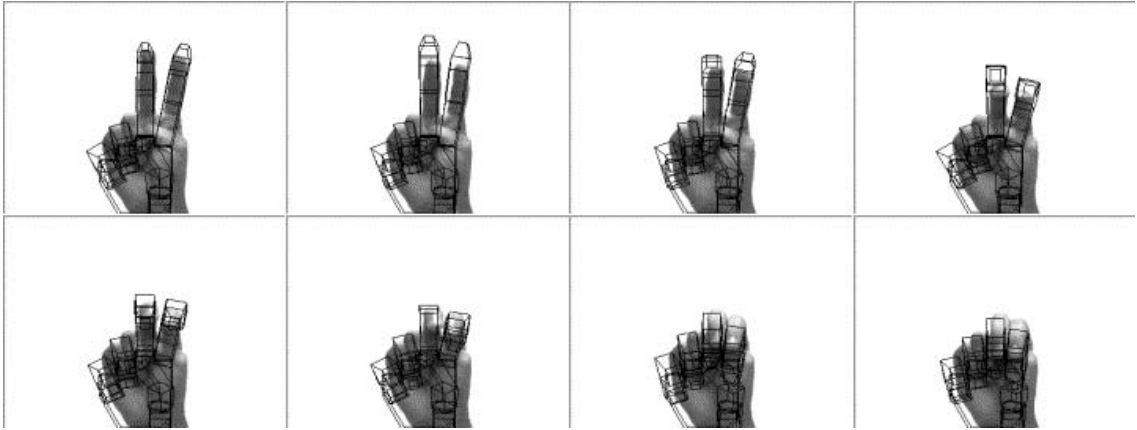


図 3 推定結果の例

Fig. 3 Examples of results.

の推定実験として、データベース内のデータを被推定画像としての推定実験も行った。用いたデータベースは、特徴量と角度の組が 4,000 組保存されたものであり、自己組織化した結果、クラス数は約 150 個のクラスが生成され、各クラスの保有データの範囲は各指の角度がそれぞれ平均 15 度程度となった。データベース作成時には約 30 fps でカメラからの画像を取得し、それと同期させて角度情報を取得した。想定する多数パターン手の形状をカメラの前で再現し、そのときの速度は 2 秒程度で手の開閉を行うといった程度の速度で行った。

本実験では、PC (CPU : Pentium IV 2.8 GHz, 主記憶 : 512 M), カメラ (MEGAPLUS 社製, ES310/T) とデータグローブ (Virtual Technologies 社製, CyberGlove) を使用した。

3.2 結果および考察

図 3 に推定結果の例を示す。実画像の上に、ワイヤフレームで描画した推定結果が重ね書きしてある。手指を連続して動かした場合に、手指角度が高精度で推定ができていることが直感的に理解できる。また、データベース構築時には角度データ取得のためのデータグローブを装着していたが、推定時には非装着で推定を行っていることが分かる。さらに、同様に照明環境が変わっても、手画像が背景とほぼ同じにならない限りは推定が行えることが確かめられた。実画像を用いた実験では、推定角度の定量的評価のためには真値と推定値とを比較する必要があるため、データベース構築時と同様にデータグローブを装着し、その上から白手袋をつけて推定実験を行った。実験結果を図 4 に示す。図中では中指第 2 関節の角度を示している。ここでの角度は全指を伸ばし、閉じている状態での角度を

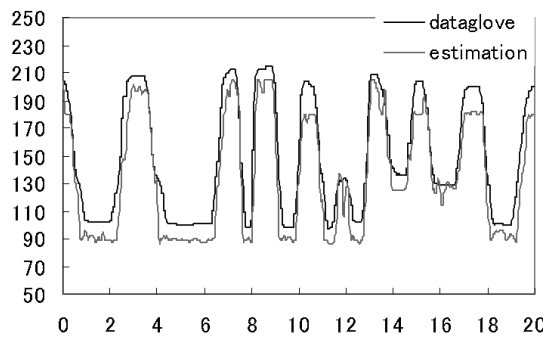


図 4 入力画像の手指角度と推定結果の例

Fig. 4 Changes of joint angles and estimated time-series.

第 2 関節については 180 度、指の間の角度については 0 度としている。このときのシステムは 50 fps 以上で動作し、実時間推定が可能となっている。同図は、データグローブの角度データ、および推定結果の移動平均のグラフである。ここで、本実験での推定結果はそのままの値では振動が大きい。そこで推定結果の前数フレームの角度を用い移動平均をとることで動き平滑化した。

同図から分かるように、データグローブで得られた角度データを真値とした場合の推定角度の誤差の標準偏差は 7.8 度であり、高精度とはいえない結果となっている。特に角度変化に対し画像変化の乏しい部分（たとえば、ジャンケンのグーの形）の推定がうまくいっていない傾向が見られた。これは人の推定を考えれば当然の推定結果であるといえる。すなわち、各指の角度が 10 度違う CG 画像を図 5 に示すが、内部角度の差 10 度には視覚上の差異はほとんど認められないため、システムの誤差出力は許容範囲内といえる。反対に、視覚上の差異が認められる区間、つまり角度

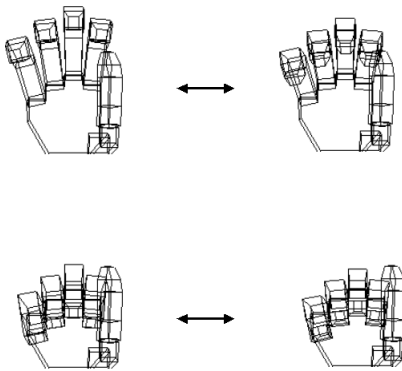


図 5 手指角度が 10 度違う CG 画像の例

Fig. 5 Examples of CG images with difference of 10 degrees of joint angles.

の変化に対し画像上の変化が激しい部分であり、指を屈曲する中間部分での推定はうまくいっていることが図から読み取れる。なお、人の指の動きは各関節が連動して動くため、生理学上の知見をもとにして、第 2 関節の角度の動きを 10 度とした場合、第 1 関節と第 3 関節をとともに 7 度動かした。また、解像度を下げたことによる精度の劣化や、画面分割数を増やすことによる精度の向上なども確認できた。しかし、視覚上の差異がない部分については現画像でもほぼ違いが見られないため、同程度の精度しか得られず、逆に処理時間の増大が顕著に出たため、現在の解像度と分割数に決定した。また、本研究での目的であるロボットの見真似による学習においては、実時間で手の動作情報が得られるため、この動作情報をもとにしたデモ動作や学習も並行して行えるので、見真似による動作教授は十分に可能だと考えられる。

次に、比較実験の結果を図 6 に示す。同図は、中指第 2 関節の角度である。前述した本実験との違いは、ここでの比較実験ではデータベース内から入力画像を選んで推定対象とするため、手の位置がデータベース画像と必ず一致、もしくはきわめて類似している点である。したがって、この結果から入力画像の位置を求める処理を追加した場合に、どの程度推定精度が改善できるかを知ることができる。データベースから入力画像を選んで推定対象とした場合の誤差の標準偏差は 2.86 度であり、きわめて高精度の推定が実現できる。将来、位置を補正する処理を本システムに組み込むことにより、この程度にまで推定誤差を減少させることが予想できる。また、画像処理部を除いた検索時間のみを見れば 1 枚あたりの処理時間は 1.61 msec となっており、自己組織化データベースを利用した 2 段階検索の有効性が証明されている。

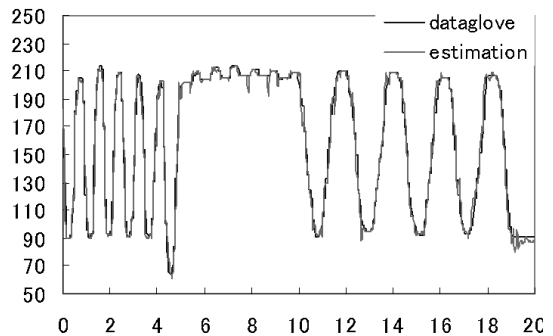


図 6 データベース情報を入力とした場合の推定結果例

Fig. 6 An example of estimation to input of database image.

また、前述したように、未知の入力画像に対する推定誤差は、標準偏差で 7.8 度であった。本件は画像処理システムであるため、たしかにジャンケンのグー状態での指関節の微小な変化は、見えのうえではほとんど差違が現れず、数値的には大きな推定誤差となって現れる。ところが、この誤差の中には、データグローブ使用時のキャリブレーションに起因する誤差も含まれる可能性がある。つまり、定量的評価のために真値としているデータグローブ出力は、手袋に装着された歪ゲージから関節角度を計算するため、装着のたびごとに歪ゲージ出力と指関節値との間のキャリブレーション操作を行う必要がある。未知の入力画像に対する推定では、データベース作成時とは別に新たにキャリブレーション操作を行う必要がある。しかし、キャリブレーション時に、用いる指を伸展した状態では基準となるものがないため、その角度が提示されたものからずれやすくなる。そのため、推定時に新たにキャリブレーションを行っても、データベース構築時のキャリブレーションの状態と異なる可能性がある。反対に、データベースの中から推定画像を選ぶ 2 番目の実験では、新たなキャリブレーションを行う必要がない。この影響が誤差の標準偏差 7.8 度の中に混入している可能性はおおいにあり、本システムが有する本質的な推定能力は、もう少し良好なものと結論づけても差し支えない。

なお、本実験ではデータベースとして、特徴量と角度の組が 4,000 組保存されているデータベースを使用したが、手指形状すべてを表すには、データセットが不足しているデータベースである。また、現在では手首の回旋といった掌を回転させる動作にはデータグローブが対応しておらずデータベースが構築できないため、それらに対応した推定はできない。しかし、本システムは手のモデル情報などを使わないで画像の

マッチングを行うものであり、原理的には回旋などにも対応できると考えた。将来的には、磁気式モーションキャプチャなどのシステムで手首の回旋情報を取得してデータベース構築を行えば、カメラ前でデータグローブを装着して動かすことのみでデータベース構築を行えるため、多様な手指形状をシステムを理解しなくても簡単にデータベースに追加可能となる。また、データベースのクラスの自己増殖、消滅アルゴリズムにより新規のデータが複数登録された場合にも組織化を新規に始める必要がなく、継続してデータベースの自己組織化が行われる。

また、将来的にはデータベース量が現在よりも増大していく。しかし、各クラスは近傍クラスの情報を持っているため、現在のクラスから角度的に近い周囲のクラスを検索することが可能となる。これにより実時間での推定を行っているため角度変化が大きく変わらないと仮定できるので周囲の検索のみで高精度に推定することが可能となる。

4. おわりに

従来の手指形状推定では、特殊な機器を必要とする、あるいはリアルタイムでの推定が行えない、といった問題があった。著者らの研究グループでは、データベースから類似画像を高速検索する手法により、単眼カメラによる手指形状の実時間推定方法について提案してきた。ところが、データベース規模が増大すると、処理時間の増加が予想された。そこで本研究では、自己増殖と自己消滅を含んだ自己組織化によりデータベースを効率的に配置することにより、大規模データベース使用時でも高速検索を可能にする方法を提案した。

画像から背景を除去し、画像サイズを 64×64 に変更した。この画像に対し輪郭抽出を行い、画像の特徴を環境光や肌の色にかかわらず、手の形においてのみ変化する特徴として取り出した。画面を複数画面に分割し、高次局所自己相関関数を用いて特徴量に変換し、それとともにデータグローブから取得した角度データを特徴データとして保存した。データベースを角度により自己組織化させることで、クラスの代表量を用いた2段階検索を可能にした。また、自己増殖消滅アルゴリズムの追加によって、クラスのデータ量の均一化を図り、クラス内の検索時間を各クラスで一定にした。

推定時の入力画像は、データベース構築時と同様に各種画像処理後、特徴量に変換した。クラスの代表特徴量を用い、画面から得られた特徴量との距離を算出した。距離の近いクラスを推定角度候補クラスとし、推定角度候補クラスとその近傍クラスに所属する各

データについて再度特徴量間の距離を算出した。特徴量間の距離が最も近いものの角度データを推定結果とした。

本手法を用いて連続的な手指形状画像列の推定を行った結果、比較的良好な推定精度を保ちながら、CPUクロック周波数 2.8 GHz の PC により 50 fps 以上の処理を実現した。

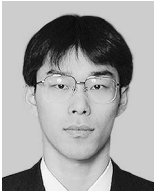
参考文献

- 1) 上田悦子, 松本吉央, 今井正和, 小笠原司: 多視点シルエット画像を用いた手の形状推定, 情報処理学会研究報告, 2001-CVIM-128, pp.25-31 (2001).
- 2) 島田伸敬, 今井章博, 白井良明: 単眼画像入力による非接触ビデオレート手指形状推定システム, 第8回画像センシングシンポジウム, pp.313-318 (2002).
- 3) 島田伸敬, 白井良明, 久野義徳, 三浦 純: 緩やかな制約知識を利用した単眼視動画像からの関節物体の形状と姿勢の同時推定, 信学論 D-II, Vol.J81-D-II, No.1, pp.45-53 (1998).
- 4) 亀田能成, 美濃導彦, 池田克夫: シルエットを利用した手指の三次元形状推定法, *Proc. MIRU'92*, II, pp.239-246 (1992).
- 5) 星野 聖, 植山祐樹, 谷本貴頌, 佐藤智子, 川淵一郎, 廣池 敦: 見まねするロボットハンドの試作, 信学技報 HIP, Vol.103, No.455, pp.97-102 (2003).
- 6) Odo, S. and Hoshino, K.: Hand shape identification using higher-order local autocorrelation features of log polar coordinate space, *Journal of Robotics and Mechatronics*, Vol.15, No.3, pp.534-540 (2003).
- 7) Otsu, N. and Kurita, T.: A new scheme for practical, flexible and intelligent vision systems, *Proc. IAPR Workshop on Computer Vision*, pp.431-435 (1998).
- 8) Kubo, M., Aghbari, Z., Oh, K. and Maki-nouchi, A.: Image Retrieval by Edge Features using Higher Order Autocorrelation in a SOM Environment, *IEICE Trans. Inf. Syst.*, Vol.E86-D, No.8, pp.1406-1415 (2003).

(平成 17 年 5 月 20 日受付)

(平成 17 年 11 月 18 日採録)

(担当編集委員 岩井 儀雄)



谷本 貴頌

2000 年筑波大学工学システム学
類卒業。現在，同大学院システム情
報工学研究科在学中。パターン認識，
人工知能の研究に従事。電子情報通
信学会学生会員。



星野 聖（正会員）

1993 年東京大学医学系大学院博
士課程修了。同年東京医科歯科大学
医学部助手。1995 年琉球大学工学
部助教授。2002 年筑波大学機能工
学系助教授となる。1998 年より科
学技術振興事業団さきがけ研究 21，2002 年より同事
業団（現・同機構）基礎的研究発展推進事業の研究者
を兼任，現在に至る。1993 年医学博士。1996 年工学
博士。ヒト運動解析，計算論的脳科学，ヒューマノイ
ドロボットの研究に従事。電子情報通信学会，ME 学
会，ロボット学会等各会員。
