

モーションデータベースを用いた人の動作の高速認識

タン ジュークイ† 河野憲亮† 石川聖二†(正会員)
金 亨燮† 四宮孝史††

†九州工業大学工学部
††(株)ニコン

〈あらまし〉 本稿は、データベースの検索によって人の姿勢や動作の高速認識を行う手法を提案する。人とロボットが共存する社会では、生活上の活動支援のために、人の多様な姿勢や動作・行動をロボットが認識する必要がある。このときロボットは、観察方向によらずに、また高速に人の動作などを認識しなければならない。これを実現するために提案法は、人の姿勢・動作の多様な見え方を画像として固有空間に記録する。多様な姿勢・動作の多様な見え方を記録すれば、そのデータ量は膨大となり実用的な検索速度が得られない。これを回避するため、固有空間を B-tree で構造化する。こうして得られるモーションデータベースの検索によって、高速な姿勢・動作認識を実現する。提案法を 6 種類の人の動作認識に適用し、良好な結果を得た。

キーワード：動作認識，固有空間，B ツリー，データベース

〈Summary〉 This paper proposes a technique for high speed recognition of human motions. For an intelligent robot which will help human lives in various ways, it is indispensable to recognize human motions or behaviors irrespective of its observing direction. For this purpose, we propose a structured motion database provided by an eigenspace having a B-tree structure within it. In the motion database, every appearance of a human motion is to be registered. Another idea for speeding up the recognition is to describe a motion in a compressed form. We employ a JK method to represent a motion sequence in a single image. Performed experiment showed satisfactory results.

Key words: motion recognition, eigenspace, B-tree, database

1. ま え が き

近年、カメラとコンピュータを用いた人の姿勢や動作認識の研究が盛んに行われており、セキュリティ、ロボットビジョンなどへの応用が期待されている。人が倒れたりうずくまったりしたとき、その姿勢や動作を監視カメラやロボットが自動的に検出・認識できれば、なんらかの支援を行うことができる。

そのような動作は三次元空間内で行われる。そのため、監視カメラやロボットには三次元物体認識能力が備わっていることが求められる。三次元物体の認識問題は、観

察方向によらずに対象を認識できなければならないという難しさがある。これを解決するために、従来は、三次元の人モデルを画像にあてはめるという方法^{1,4)}や、正面向き・横向きの像のみを対象とするなど、観察方向を限定して姿勢・動作認識を行う方法^{5),15)}が主流であった。しかし前者は処理時間が膨大になること、後者は認識方向の限定が実際的でないなどの問題で、いまだに実用化には至っていない。これを解決するために、我々は、固有空間法を用いて人の姿勢・動作認識を行う方法^{4),5)}を提案している。

固有空間法²⁾は画像を点として記憶する方法で、姿勢や動作のあらゆる見え方の画像が、点の集合として記憶されデータベース化される。固有空間法を用いれば、三次元物体は二次元画像の集合と見なすことができる。よって、物体のすべての見え方を固有空間上の点の集合とし

“High Speed Human Motion Recognition Employing a Motion DataBase” by Joo Kooi TAN, Kensuke KOUNO, Seiji ISHIKAWA (Member), HyoungSoup KIM (Faculty of Engineering, Kyushu Institute of Technology), and Takashi SHINOMIYA (Nikon Corporation).

て登録すれば、観察方向によらない物体認識が可能となる。一方、人は多様な姿勢を取り多様な動作を行うため、それをすべて登録すればこのデータベースは膨大となり、検索時間が実用的でなくなるという問題が生じる。そこで、膨大なデータベースに対する高速な検索法が必要となる。我々は、固有空間を構造化することによりこの問題を解決する。

固有空間法では画像を n 次元空間に写像し、画像間の相関を距離で計算することによって、類似画像の検索を行う。そのため、固有空間をデータベース化するには、高次元検索構造が必要となる。従来、高次元検索構造を採用し、高速検索を可能とした研究として文献7)~9)がある。文献7),8)では、データ構造として木構造を採用しているが、この木構造は平衡木ではなく、また外部記憶装置を用いることを前提としたデータ検索が考慮されていない。また、構造が平衡木となっていないため、データによっては検索時間がデータ数に比例する場合もあり、木構造のもつ利点が生かせない。

更に、大規模なデータベースを構築する場合、その容量は膨大となる。そのためデータを外部記憶装置に保存する必要があるが、外部記憶装置は内部メモリと異なりアクセス時間が遅いという点を考慮する必要がある。文献9)では空間を分割管理するための多次元データ管理構造として kd-tree¹²⁾ の拡張型を用いているが、kd-tree はデータの次元が増加するほど探索時間が指数関数的に増大するという欠点がある¹¹⁾。これは分割領域の重なりを許容しているためである。探索点が分割領域の重なり部に存在する場合、その領域をすべて探索するため探索時間が増大する。固有空間法の性質上、データベースが膨大になるほど固有空間の次元数を高くする必要があるので、kd-tree を固有空間に適用することは難しい。

そこで本稿では、固有空間に、平衡木の一つで外部記憶装置上のデータ探索に優れている B-tree¹⁾ を適用した、構造化固有空間を提案する。B-tree は本来多次元データ管理構造ではないが、本研究では固有空間の分割により B-tree の適用を可能とした。分割の際に領域の重なりを許容しないため、上述の分割領域の重なり部を全探索するという問題は発生しない。本法により、データベースの容量の上限を大幅に引き上げることができ、知能ロボットによる人の姿勢や動作認識技術を、実用化に近づけることができる。

2. 固有空間法

2.1 データベースの構築

画像 I_p ($p = 1, 2, \dots, P$) をラスタースキャンしてベクトル $\mathbf{x}_p = (x_1, x_2, \dots, x_N)^T$ を得る。ただし、

$\|\mathbf{x}_p\| = 1$ 、また N は全画素数である。 N 行 P 列の行列 $X \equiv (\mathbf{x}_1 - \bar{\mathbf{x}}, \mathbf{x}_2 - \bar{\mathbf{x}}, \dots, \mathbf{x}_P - \bar{\mathbf{x}})$ を定義する。ここで、 $\bar{\mathbf{x}} = \sum_{p=1}^P \mathbf{x}_p / P$ である。共分散行列 $Q = XX^T$ の固有方程式を解けば、 Q に関する N 個の固有値 λ_i ($i = 1, 2, \dots, N$)、固有ベクトル \mathbf{e}_i ($i = 1, 2, \dots, N$) が得られる。固有値 λ_i を降べきの順にソートし、 N より十分小さい k 個の固有値 $\lambda_1, \lambda_2, \dots, \lambda_k$ を選び、対応する固有ベクトル $\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_k$ を基底ベクトルとして k 次元固有空間 $ES(\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_k) \equiv ES$ を構築すれば、ベクトル \mathbf{x}_p は次式によって固有空間 ES 上の点 \mathbf{g}_p に投影される。

$$\mathbf{g}_p = (\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_k)^T (\mathbf{x}_p - \bar{\mathbf{x}}) \quad (1)$$

上述の方法により学習画像は固有空間に投影され、固有空間上の点としてデータベースに記憶される。なお、固有空間法では、対象物体は背景差分によって画像から切り出され、一定のサイズに正規化されるため、対象物体の位置ずれや拡大縮小の影響はない。

2.2 未知画像の認識

未知画像 $I_{p'}$ を固有空間上に投影した点の座標を $\mathbf{g}_{p'}$ とすれば、学習画像 \mathbf{g}_p との最短距離 d_{p^*} は次式で与えられる。

$$d_{p^*} = \min \|\mathbf{g}_p - \mathbf{g}_{p'}\| \quad (2)$$

$\mathbf{g}_{p'}$ と最小距離となる座標点が存在し、かつ、ある正の値を取るしきい値 ϵ に対して不等式 $\|\mathbf{g}_{p^*}\| < \epsilon$ が成り立てば、未知の姿勢が $p' = p^*$ として認識される。ここでしきい値 ϵ は実験的に定められる値である。本研究では距離 d の定義として次式で示されるユークリッド距離を採用する。

$$d \equiv \sum_{i=1}^{n_d} (x_i - q_i)^2 \quad (3)$$

ただし、 n_d は空間の次元数、 x_i および q_i は座標成分である。

2.3 固有空間法の問題点

固有空間法による類似画像の検索では、式 (2) を用いて最小距離の点を検索する。これは線形探索法と等価であり、平均して $n/2$ 回の距離計算が行われるため、その計算量は $O(n)$ となる。ただし n は全画像数である。従って、データベースに登録される画像に比例し探索時間も増大する。このため、大規模なデータベースに対しては検索時間が実用的でなくなる。

2.4 Partial Distance Search

式 (3) では固有空間の次元数分の積和演算を行うため、計算量は膨大となる。この問題に対し、Cheng らは Partial Distance Search (PDS)¹⁰⁾ と呼ばれるアルゴリズムを提案している。これは式 (3) に対し簡単な変更を

加えたアルゴリズムである。式 (3) において、 n_{part} 次元までの部分積 d_{part} を

$$d_{part} = \sum_{i=1}^{n_{part}} (x_i - q_i)^2 \quad (4)$$

とすれば、この部分積がある正の値をもつしきい値 ϵ に対して

$$\epsilon \leq d_{part} \quad (5)$$

を満たすとき、 n_{part} 次元以上の計算をやめる。

これはパターンマッチングにおける SSDA 法¹³⁾ と同様のアルゴリズムである。Cheng らによれば、PDS により計算量は 1/4 程度まで減少できる。本法では、式 (3) を用いるときに PDS を適用する。

3. B-tree 構造

3.1 B-tree の概要

B-tree¹⁾ とは m 分木をベースとした探索木である。 m 分探索木のうち以下の条件を満たす木を m 階の B-tree という。

- (i) 根は葉であるか、または $2 \sim m$ 個の子を持つ
- (ii) 根、葉以外の節は $\lceil m/2 \rceil \sim m$ 個の子を持つ
(ただし、 $\lceil x \rceil$ は x 以上の最小の整数を表す)
- (iii) 根からすべての葉までの経路の長さが等しい

B-tree では格納するデータから作られるキー (境目を表す値) が重要な意味を持ち、キーによって根や節が構築される。また、キーは大小比較することが可能なスカラー値である。B-tree では、データを持つのは葉のみで、データは葉に格納される。それ以外の節 (これを内部節 (Page) という) はキーだけを持つ。

条件 (iii) から、B-tree は常にバランスが取れていることになる。B-tree の場合、節が可変個の子を持てるという性質を利用して木のバランスをとる。つまり、5 階の B-tree では、内部節は 3~5 個の子を持てることになり、この範囲内で子の数を調整して木のバランスが取れた状態に保つ。

n 個のデータを持つ B-tree の高さは、最悪の場合で $\log_{\lceil m/2 \rceil} n$ 程度、最良の場合で $\log_m n$ 程度となる。すべての節が $\lceil m/2 \rceil$ 個しか子を持っていないのが最悪の場合、すべての節が m 個の子を持っているのが最良の場合となる。いずれの場合においても B-tree の高さは $O(\log n)$ に収まる。

根や節は同じ大きさの内部節として表現される。図 1 に内部節を示す。この内部節において、キー x_i は昇順に格納される。ただし、節は $\lceil m/2 \rceil \leq l \leq m$ 、根は $1 \leq l \leq m$ である。更に、子を示す $l+1$ 個のポインタ p_0, p_1, \dots, p_l を保持している。図 2 に 5 階 (最大 5 個の子を持つ) の B-tree の例を示す。ただし、キーは

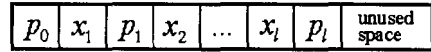


図 1 内部節 (Page) の構造
Fig. 1 Organization of a page

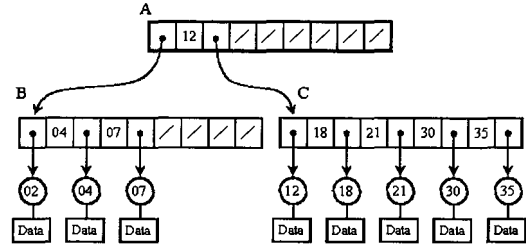


図 2 5 階 (最大 5 個の子を持つ) の B-tree の例
Fig. 2 An example of a B-tree having 5 descendants

データと同じ 10 進数で記述された数値とする。図 2 においてポインタ p_i は矢印で表現されている。大きな長方形は内部節を表し、内部にある数字はキーである。内部節の中にある斜線はキーまたはポインタが入っていないことを意味する。円は葉を表し、データが関連付けられる。また、内部節の保持するキーには、隣接した右の部分木の最小値が代入される。

3.2 探索

B-tree の内部節では、キー x_i が昇順に格納され、キーは隣接した右の部分木の最小値を保持する構造を持つ。このため、データを探索する際は、目的データの持つキーを探索キーとし、内部節の持つキーと比較することによって、どの部分木をたどればよいのかを判断する。図 2 において探索キーを 04 とおくと、まず根のキー 12 と比較し、探索キーのほうが小さいため、キー 12 の左側にあるポインタをたどり、左の節に移動する。そして、探索キー 04 は節のキー 04 以上であるため、キー 04 の右側のポインタをたどりキー 04 を持つ葉に移動する。探索キー 04 と葉の持つキー 04 は等しいため探索が成功し、葉の持つデータを引き出すことができる。

次に、B-tree における探索の計算量を求める。B-tree における探索では、根から始めて葉に到達するまでに $O(\log n)$ 個の節をたどる必要がある。また、各内部節でどの部分木をたどるかを判断する必要があるが、この計算量は内部節のキーを順に比較すれば $O(m)$ の線形探索となるため、キーが昇順に並んでいることを利用し、2 分探索を行えば $O(\log m)$ となる。一般に n の方が m よりもはるかに大きいため、 m は定数と見なすことができる。従って、B-tree における探索は常に $O(\log n)$ で実行できることになる。

4. 構造化固有空間

4.1 手法

提案法は、固有空間を超立方体で表現されるピンで区分し、このピンを B-tree によって管理することにより固有空間を構造化する。

固有空間上の座標値 e_k ($k = 1, 2, \dots, K$) を、ある幅 L を持った V 個のセクションに分割し、各セクションを固有の番号 $s_{k,v}$ ($v = 1, 2, \dots, V$) でラベル付けする。画像 I_p が式 (1) によって固有空間の点 $g_p = (g_1, g_2, \dots, g_k)$ に投影されると、 g_k ($k = 1, 2, \dots, K$) はいずれかのセクションに含まれるから、各軸において、セクションの固有の番号 $s_{k,v}$ ($v = 1, 2, \dots, V$) が与えられる。この結果、座標点 g_p は次式によって V 進数 K けたのセクション番号 S_p に変換される。

$$S_p = s_{1,v_1} s_{2,v_2} s_{3,v_3} \dots s_{K,v_K} \quad (6)$$

S_p は単なる数値であるため、B-tree のキーの条件である“比較可能なスカラー値”を満たしている。従って、 S_p を用いて画像を B-tree に格納することができる。この操作は空間充密曲線を求めることに相当する。固有空間内における第一次元は最も固有値の大きい固有ベクトルが与えられる。これは、最も分散が大きいことを意味し、他の次元と比べて第一次元はデータの分離性が大きい。そのため、第一次元の座標値は他の次元の座標値よりも重要視されることが望ましい。式 (6) では第一次元の座標値 g_1 が S_p の中で最上位の桁 (MSB) s_{1,v_1} として与えられる。このため、第一次元の座標値 g_1 は S_p の中で支配的となり、第一次元の座標値は他の次元の座標値よりも重要であるという固有空間法の性質を反映している。

次に、構造化固有空間を用いた未知画像認識法について述べる。未知画像を $I_{p'}$ とすると、まず $I_{p'}$ を式 (1) によって固有空間に投影し、固有空間上の座標値 $g_{p'}$ を得る。そして式 (6) によって $g_{p'}$ をセクション番号 $S_{p'}$ に変換する。この $S_{p'}$ を探索キーとして B-tree を探索することにより、 R 個の候補画像 g_{p_r} ($r = 1, 2, \dots, R$) が得られる。そして、最後に次式を適用すれば、最も近い姿勢 $p' = p^*$ を得ることができる。

$$d_{p^*} = \min \| g_{p_r} - g_{p'} \| \quad (7)$$

データベースに登録されている全画像数を P とすれば、 $R \ll P$ となることが期待されるため、探索速度は速くなる。B-tree 構造を持つ二次元固有空間の例を図 3 に示す。図中における e_1 、 e_2 は固有空間における座標軸である。また、黒丸の点は固有空間に投影された学習画像群を表している。

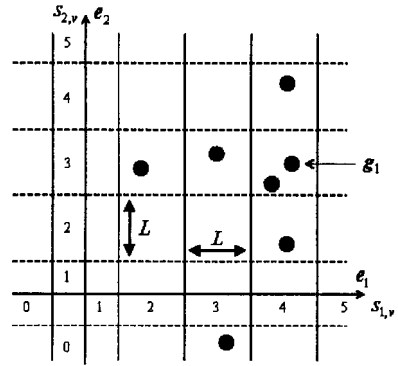


図 3 B-tree 構造を持つ二次元固有空間
Fig. 3 A 2-D eigenspace with B-tree structure

座標軸 e_1 は同一の幅 L で 6 分割されており、各セクションに 0~5 の番号がラベル付けされている。この番号が $s_{k,v}$ ($v = 1, 2, \dots, V$) である。座標軸 e_2 も同様に分割され、ラベル付けされている。

ここで、座標点 g_p をセクション番号に変換することを考える。図 3 で、図中右上の座標点 g_1 は座標軸 e_1 上の 4 番目のセクションに属しており、また座標軸 e_2 上で 3 番目のセクションに属している。従って、座標点 g_1 は 6 進数 2 けたとなるセクション番号 $S_1 = 43$ が与えられる。このようにして座標点 g_1 はセクション番号 S_1 に変換される。

セクション番号 S_1 は、座標軸 e_1 における 4 番目のセクションと、座標軸 e_2 上における 3 番目のセクションで囲まれた領域 (ピン) を意味している。固有空間における探索範囲をこのピンの中に制限すれば、その中に存在する画像 (候補画像) はデータベースに登録されている全画像数よりはるかに少ないため、高速検索が可能となる。

このように、セクション番号の導入により、 n 次元固有空間を超立方体で表現されたピンに分割することができる。そして、探索範囲をその中に制限することにより高速検索が可能となる。このセクション番号をキーとして B-tree に登録することは、各ピンを木構造によって関連付け、管理することに相当する。そして、セクション番号を探索キーとして B-tree を探索することにより、目的のピンに高速にアクセスし、その中に含まれる候補画像を得ることが可能となる。

4.2 境界問題

固有空間を分割するために生じる問題について述べる。固有空間はセクションによって分割されるため、図 4 に示すような境界問題が発生する。図 4 は B-tree 構造を持った二次元固有空間内に、二つの学習画像 A、B とテ

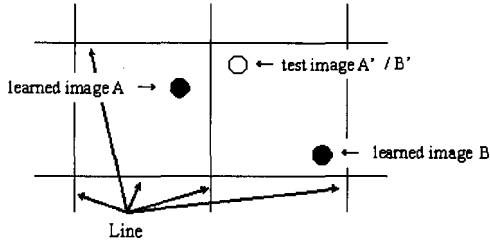


図 4 境界問題

Fig. 4 Difficulty near the border of a bin

スト画像 A' が投影されている様子である。Line はそれぞれセクションの境界を表している。

この図において、テスト画像 A' に一番近いのは学習画像 A であり、全探索を行えば学習画像 A が正しく認識される。しかし本手法では、この 2 点の間に境界があるため学習画像 A ではなく、ビンの中で最も近い学習画像 B が選択され、認識率低下の原因となる。一方、ノイズを含むテスト画像の点 B' の本来（ノイズが含まれなければ）対応する学習画像が、同じビンの点 B であるとき、 B' に対して距離最小の点が隣接するビン内の A であっても、 A が選択されず B が選択されることになり、これは認識率上昇に貢献する。

しかし一般には、最短距離にある学習画像が正解である確率が高いので、前者が発生する可能性、すなわち境界問題が悪影響を及ぼす可能性のほうが大きいと思われる。

この問題は、境界付近にテスト画像が存在する場合に発生するため、このような状況をできるだけ避けるためには固有空間を細かく分割することを避け、セクションの境界を少なくすればよい。こうすることにより境界線付近に存在するデータ数を減少させることができる。しかし、固有空間を粗く分割することは各ビンに含まれる画像数の増大を招くため、高速検索を阻害する。このことはセクションの幅 L に依存するため、最適な L の設定が重要となる。

境界問題に対する他の対応策として、同一クラスに含まれる類似のデータを多数分布させるという方法が考えられる。なお、テストデータの投影点が境界に近い場合はその境界に接するビンも探索するという方法が考えられるが⁸⁾、隣接するビンの個数は次元の増大に伴って指数的に増大するため、慎重に決定する必要がある。本稿では最適な L を定めることによって境界問題に対応する。

4.3 最適セクション幅の決定

セクションの幅 L の決定方法について述べる。 L の値をパラメータとして変化させたとき、それに伴い認識率と、ビンの中に存在する候補画像の数に変化する。認識率 $r(L)$ は次式で定義される。

$$r(L) = \frac{n_s(L)}{n_t} \tag{8}$$

ここで、 n_t はテスト画像を用いてデータベース検索を行った回数、 $n_s(L)$ はその中で認識に成功した回数である。従って、 $r(L)$ は $0 \leq r(L) \leq 1$ となる。また、候補画像の数を評価するために、候補率 $c(L)$ を次式のように定義する。

$$c(L) = \frac{n_R(L)}{n} \tag{9}$$

ここで $n_R(L)$ は 1 枚のテスト画像に対してデータベース検索を行った場合に得られる候補画像の数であり、 n はデータベースに登録されている全画像数である。 $n_R(L)$ は $0 \leq n_R(L) \leq n$ の関係を常に満たすため、 $c(L)$ は正規化され $0 \leq c(L) \leq 1$ となる。 $c(L)$ の値は検索速度の指標であり、 $c(L) = 1$ のとき、検索速度は全探索の場合と等しく、値が小さくなるほど検索速度は向上する。

L が小さければピンは小さくなり、その中に含まれる候補画像は少なくなる。そのため検索速度は向上するが、境界問題が発生しやすくなるため認識率は下がると考えられる。逆に L が大きければ、境界問題は発生しにくいので認識率は高くなるが、検索速度は全探索の場合に近づく。従って $r(L)$ と $c(L)$ はトレードオフの関係にあり、何らかの評価値が必要となる。そこで、本研究では以下の評価値 D_{rc} を用いる。

$$D_{rc} = r(L) - c(L) \tag{10}$$

評価値 D_{rc} が最大となるときの L を採用すれば、認識率が高く、かつ候補率の低い L を決定することができる。また、 $r(L)$ と $c(L)$ はどちらか一方を任意に指定することもできる。認識率を犠牲にしても検索をリアルタイムで行いたいなら、 $c(L)$ がこの要求を満たすように L を定めることができる。

5. 実験 1 (姿勢認識実験)

提案法の有効性を検証するために、大規模データベースを用いた人の姿勢認識実験を行った。被験者の前方に約 30~40 度の間隔で 4 台のカメラを配置し、被験者 11 名に各々 6 種類の動作を行ってもらった。動作 1~6 はそれぞれ 1: 重いものを持ち上げる、2: ものを拾い上げる、3: おなかを抱え込む、4: 頭を抱え込む、5: 足踏み、6: 転倒の動作である。なお本研究は、人の生活支援を行うヘルパーロボットの開発を将来の目的とした研究の一環であるため、動作 1, 3, 4, 6 は、ロボットによる支援が期待される動作として採用している。また動作 2 はロボットが注視する必要のある動作として、動作 5 は歩行者認識のために採用している。図 5 に各動作を示す。

各動作を構成するフレーム数はおよそ 121, 58, 81, 83, 39, 96 である。人の動作は連続した姿勢の連なりと

見なすことができるため、動作中の1フレームを1姿勢と見なし、各フレームが与える姿勢の認識を行う。なお、画像の前処理として、まず図5に示される各動作の濃淡画像を微分し、あらかじめ撮影しておいた背景画像の微分画像との差を取った差分画像を生成する。次に、その差分画像から人物を含む最小の長方形領域を切り出し、そ



図5 各動作の表現

Fig. 5 Presentation of a motion

れを 32×32 画素のサイズに正規化した画像を認識対象画像とする。微分画像を用いるのは、衣服問題（固有空間法で人物を扱う場合、原濃淡画像をそのまま用いれば、同じ人物・同じ動作でも衣服によって異なるデータとなるという問題）をできるだけ抑えるためである。

Leave-one-out 法に基づき、11名の被験者から1名を任意に選んでその4方向6動作(1,912枚)をテストデータとし、残りの10名の被験者の4方向6動作(18,868枚)を学習データとする。学習データを用いて構造化固有空間を作成し、それを用いてテストデータの認識を行い、認識率を求める。初めの1名の選び方は11通りあるから、同様の認識実験を11回繰り返して平均認識率を求める。本データベースは、18,868点の画像データを含むことになる。

テスト画像の認識は式(2)を用いて行われるが、境界問題のため式(2)の与える最近傍画像が正しい結果でないことが考えられる。そこで、テスト画像 I_t と認識した結果画像 I_p が同一姿勢かどうかの判別に、次式で定義される相違度 D を用いる。

$$D = \frac{\sum_i^M \sum_j^N |I_p(i,j) - I_t(i,j)|}{MN \times A} \quad (11)$$

ただし、 M 、 N はそれぞれ画像の幅と高さ、 A は画素値の値域の最大値である。相違度 D がある正の値を持つしきい値に対して不等式 $D < D_t$ を満足すれば、 I_t と I_p は同じ姿勢を持つと判断する。実験結果を図6に示す。同図は横軸が L 、縦軸が認識率および候補率である。Line.1は式(8)で与えられる認識率、Line.2は式(9)で与えられる候補率、またLine.3は式(10)で与えられる評価値を表す。Line.0は全探索の場合の認識率である。図6より、評価値を最大にするセクションの幅は $L = 0.4$ であった。

$L = 0.4$ のときの各パラメータを、探索時間と共に表1

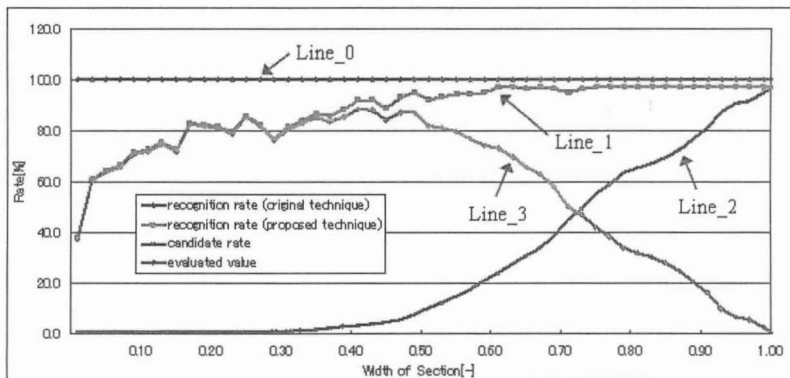


図6 姿勢認識実験の結果

Fig. 6 Experimental result for posture recognition

表1 $L = 0.40$ の場合と全探索の場合の探索時間の比較
 Table 1 Experimental result at $L = 0.40$ compared with blind search

	Original Technique	Proposed Technique
Recognition rate	73%	61%
Candidate rate	100% (18868 frames)	4% (755 frames)
Query time	2705 msec	111 msec

に示す。比較のため、表1には全探索の場合の数値も挙げている。提案法の候補率は4%であり、全探索の場合に比べ検索は24倍速く、1画像の平均検索時間は111msecである。一方、認識率は全探索の場合より12%ほど低下しているが、これは境界問題に起因すると考えられる。

6. 実験2 (動作認識実験)

提案法を用いて動作認識を行う。固有空間法を用いた動作認識の手法として文献3)~6)が提案されている。文献3), 4)では次の性質に着目し動作認識を行う。固有空間内において、連続的に変化する時系列画像は緩やかな曲線を描く。固有空間上では相関が高い画像どうしは近い位置に投影されるため、類似する動作を表す時系列画像も固有空間上で近接した曲線を描く。従って、未知の動作が固有空間上で描く曲線と、学習された曲線を比較し、最も近接した曲線を持つ動作を検索することにより未知動作を特定する。しかし、この手法を構造化固有空間に適用すると、動作は固有空間上で曲線を描くため、セクションの境界をまたぐことになり認識法が煩雑になり、また認識時間も長くなる。更に、1動作が多数枚の連続フレームとして与えられるためデータベースの増大を招く。

そこで、本研究では文献6)に示されるJK法を導入する。JK法は動作を表す時系列画像を1枚の画像に圧縮し、固有空間上の1点として表現する手法である。同様な方法としてMHI (Motion History Image) を用いた手法^{9), 15)}があるが、文献6)では、JK法は動作認識に関してMHI法よりもよい結果を得ている。JK法は背景画像に配慮することなく人物を抽出できる。また微分画像を用いるため光の影響を受けにくく、人物の衣服問題に対しても有効であることが期待される。こうして、膨大な動作に対する簡潔かつ効率的なデータベースが構築可能となる。1動作が1枚の画像となるため、姿勢認識と同じ手法が適用できる。

JK法は以下の方法で画像の圧縮を行う。

- (i) h 番目の人物から得られる c 番目の観察方向の m

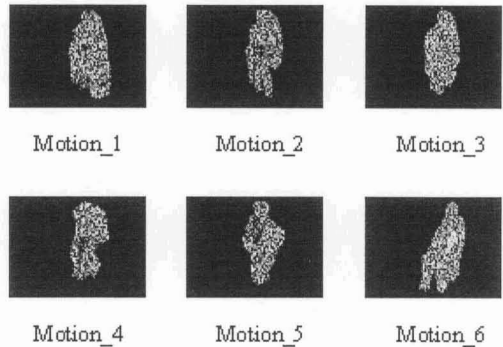


図7 圧縮された6動作

Fig. 7 Compressed images of the selected 6 motions

- 番目の動作を表す R 枚の画像に微分演算を施す。
- (ii) 画像すべてを2値化する。
- (iii) 全画像にXOR演算 (eXclusive OR演算) を施し、1枚の画像を生成する。

h ($h = 1, 2, \dots, H$) 番目の人物から得られる c ($c = 1, 2, \dots, C$) 番目の観察方向の m ($m = 1, 2, \dots, M$) 番目の動作を表す画像を $f_c^{m,h}$, その r ($r = 1, 2, \dots, R$) 番目のフレームを $f_c^{m,h}(r)$ と定義する。XOR処理は次式で与えられる。

$$U_c^{m,h}(2) = f_c^{m,h}(1) \text{ XOR } f_c^{m,h}(2) \quad (12)$$

$$U_c^{m,h}(r) = U_c^{m,h}(r-1) \text{ XOR } f_c^{m,h}(r) \quad (13)$$

$$U_c^{m,h} \equiv U_c^{m,h}(R) \quad (14)$$

ここで、 $r = 3, 4, \dots, R$ である。図7に圧縮された各動作を示す。

式(12)~(14)を用いて、動作を表す時系列画像を1枚の画像に圧縮する。これにより、20,780枚のフレームは264枚の動作画像に圧縮される。得られた圧縮画像にLeave-one-out法を適用し、認識実験を行う。すなわち、11名の被験者から1名を任意に選んでその4方向6動作(24枚)をテストデータとし、残りの10名の被験者の4方向6動作(240枚)を学習データとする。学習データを用いて構造化固有空間を作成し、それを用いてテストデータの認識を行い、認識率を求める。初めの1名の選び方は11通りあるから、同様の認識実験を11回繰り返して平均認識率を求める。

実験結果を表2に示す。比較のために全探索の場合の数値も示している。評価値を最大にするセクションの幅は $L = 0.23$ であった。また候補率は3.9%、探索時間は2.2msecであり、十分高速化されている。一方認識率を見ると、提案法は、全探索を行った場合よりわずかであるが高い。これは4.2で述べた境界問題が、たまたま認識率上昇の方向に作用したためと考えられる。実際、1

表2 $L = 0.23$ の場合と全探索の場合の探索時間の比較
Table 2 Experimental result at $L = 0.23$ compared with blind search

	JK technique + Original technique	JK technique + Proposed technique
Recognition rate	92.4%	92.8%
Candidate rate	100% (240 frames)	3.9% (9.4 frames)
Query time	14 msec	2.2 msec

データに対して、このことが生じた。反対に、境界問題がマイナスに作用しなかったのは、図7に示す圧縮画像で与えられる動作のクラス間距離が比較的大きく、境界をはさんで学習データとテストデータが接近するという状況があまり発生しなかったためと思われる。

7. 考 察

三次元空間における人の姿勢・動作認識では、二次元画像に基づく(見え方指向の)認識法は、三次元復元による認識法におけるアルゴリズムの煩雑さはないが、基本的にあらゆる見え方を記憶させるため、膨大なデータベース検索をする必要がある。この問題に対し我々は、B-treeによる構造化固有空間を用いたモーションデータベースの高速検索法を提案し、姿勢認識実験および動作認識実験で高速認識を確認した。このように、B-treeによるモーションデータベースの管理によって高速検索を実現した。

また、動作認識実験ではJK法による動作表現を行った。動作映像は複数のフレームからなるので、それらを1枚の映像に圧縮すれば当然認識は高速になる。よって、動作認識ではJK法も高速検索に貢献している。時間変化する動作を表現する圧縮法はモーションヒストリーメージ¹⁶⁾など、バリエーションが考えられるが、本研究で用いた排他的論理和に基づくJK法は、背景像の消去が容易であるという特徴がある。ただし時間方向に対称な動作、例えば「座って立ち上がる」と「立ち上がって座る」の分離が不得手であるため、更なる改良を検討している。

モーションデータベースを固有空間法に基づいて構成する理由は、固有空間法は画像を表現するのに、画像の前処理が少なくすむという点にある。パターン認識問題では、画像処理によって画像から特徴を抽出し、それから特徴空間にクラスタを作って未知特徴の認識を行うが、固有空間法では特徴抽出過程がなく、画像をほぼそのまま固有空間に記録し、未知画像も同様の処理に基づいて認識を行うため、認識処理が容易である。そのかわりできるだけ多くの見え方の画像をデータベースに記

録しそれらを検索する必要がある。従って、認識対象の増加とともにデータベースも膨大な容量となることが予想されるが、本法では、B-treeの導入によってデータベース検索を高速化したため、固有空間法による認識処理の容易さを用いることができた。

構造化固有空間の各軸の分割幅 L のサイズについては更なる検討が必要である。本稿では一定幅分割である程度高い認識率が得られた。しかし、軸ごとに分割幅を変えるなど、データのばらつきも考慮した分割法を検討する必要がある。

また本法のような空間の分割構造には境界問題が存在し、境界付近のデータは誤認識を発生させる恐れがある。この問題の解決も今後の課題である。更に動作認識では、ビデオ映像上の個々の動作のセグメントを手動で行った。この自動化も今後の検討課題とする。

8. 結 論

固有空間にB-treeを適用した構造化固有空間を用いてモーションデータベースを構築し、人の姿勢・動作認識を高速に行う手法を提案した。また、提案法の有効性を実験的に示した。動作の圧縮表現の改良、構造化固有空間の最適な分割幅の決定法、境界問題への対処、ビデオ映像からの動作のセグメンテーションの自動化等が今後の課題である。

実験を手伝って頂いた被験者諸氏に感謝申し上げる。本研究の一部は日本学術振興会科学研究費補助金を受けた。

参 考 文 献

- 1) R. Bayer and E. McCreight: "Organization and maintenance of large ordered indexes", *Acta Informatica* 1, fasc. 3, pp.173-189 (1972).
- 2) H. Murase and S.K. Nayar: "Recognition and spotting of 3D objects using parametric eigenspace representation", *Proc. of MIRU*, 2, pp.49-56 (1994).
- 3) H. Murase and R. Sakai: "Moving object recognition in eigenspace representation: gait and lip reading", *Pattern Recognition Letters*, 17, pp.155-162 (1996).
- 4) J.K. Tan, S. Houman, and S. Ishikawa: "Human motion representation using eigenspaces", *Proc. of TENCON*, CD-ROM (2005).
- 5) T. Ogata, J.K. Tan, and S. Ishikawa: "High-speed human motion recognition based on a motion history image and an eigenspace", *IEICE Transactions on Information and Systems*, Vol.E89-D, No.1, pp.281-289 (2006).
- 6) J.K. Tan, K. Kouno, and S. Ishikawa: "Human activity recognition independent of observation orientation", *Proc. of IWAIT*, pp.655-660 (2007).
- 7) J.Y. Chen, C.A. Bouman, and J.P. Allebach: "Fast image database search using tree-structured VQ", *Proc. of IEEE International Conference on Image Processing*, Vol.2, pp.827-830 (1997).
- 8) J. McNames: "A fast nearest-neighbor algorithm based on a principal axis search tree", *IEEE Transactions on*

Pattern Analysis and Machine Intelligence, Vol.PAMI-23, No.9, pp.964-976 (2001).

- 9) T. Shibata, T. Kato, and T. Wada: "K-D decision tree: An accelerated and memory efficient nearest neighbor classifier", Proc. of 3rd IEEE International Conference on Data Mining, pp.641-644 (2003).
- 10) D.Y. Cheng, A. Gersho, B. Ramamurthi, and Y. Shoham: "Fast search algorithms for vector quantization and pattern matching", Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing, Vol.1, pp.9.11.1-9.11.4 (1984).
- 11) S.A. Nene and S.K. Nayar: "A simple algorithm for nearest neighbor search in high dimensions", IEEE Transactions on Pattern Analysis Machine Intelligence, Vol.19, No.9, pp.989-1003 (1997).
- 12) J.L. Bentley: "Multidimensional binary search trees used for associative searching", Communications of the ACM 18, pp.509-517 (1975).
- 13) E.I. Barnea and H.F. Silverman: "A class of algorithms for fast digital image registration", IEEE Transactions on Computers, Vol.C-21, pp.179-186 (1972).
- 14) 島田伸敬, 有田大作, 玉木 徹: "関節物体のモデルフィッティング", 情報処理学会研究報告コンピュータビジョンとイメージメディア, 2006-CVIM-154, No.51, pp.375-392 (2006).
- 15) 申 熙卓, 渡辺俊典, 菅原 研: "テンポラルテンプレートを用いた動画解析手法", 信学技報, PRMU2002-111, pp.53-58 (2002).
- 16) A.F. Bobick and J.W. Davis: "The recognition of human movement using temporal templates", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.PAMI-23, No.3, pp.257-267 (2001).

(2007年4月11日受付)

(2007年8月6日再受付)



タン ジュークイ

九州工業大学大学院設計生産工学専攻博士後期課程修了。博士工学。同大SVBLポストドク研究員を経て、現在、同大工学部機械知能工学科助教。主に画像計測・解析、動作認識、コンピュータビジョンなどに関する研究に従事。1999年、計測自動制御学会九州支部第13回奨励賞、2005年、知能生命・ロボット国際シンポジウム若手奨励賞受賞。IEEE, 計測自動制御学会, 情報処理学会等会員。



河野 憲 亮

2005年、群馬大学工学部機械システム工学科卒。2007年、九州工業大学工学研究科機械知能工学専攻博士前期課程修了。同年、計測自動制御学会九州支部奨励賞受賞。在学中、主に画像処理に関する研究に従事。



石川 聖 二 (正会員)

東京大学工学部計数工学科卒。東京大学大学院工学系研究科博士課程了。工学博士。九州工業大学情報工学科講師を経て、現在、九州工業大学工学部教授(機械知能工学科)。この間、連合王国シェフィールド大学客員研究員。オランダ王国ユトレヒト大学客員教授。三次元画像計測、医学画像処理の研究に従事。計測自動制御学会, 電子情報通信学会, IEEE 各会員。



金 亨 斐

1994年、九州工業大学工学部卒。1996年、同大学大学院工学研究科博士前期課程修了。1997年より九州工業大学工学部助手(機械知能工学科)。2003年、同大学工学部講師。2005年より同大学工学部准教授。医用画像処理、パターン認識に関する研究に従事。博士(工学)。電子情報通信学会, 日本医用画像工学会, IEEE 各会員。



四宮 孝 史

1972年、日本大学生産工学部管理工学科卒業。1996年、九州芸術工科大学(現、九州大学)大学院博士後期課程(生活環境専攻)修了。博士(工学)。1972年、日本光学工業(株)(現(株)ニコン)入社、現在コアテクノロジーセンター企画部ゼネラルマネジャー。人間工学, 画像処理の研究開発に従事。