2004－CVIM－145　(23)
2004／9／11

PCへのマルチモーダルな入力手段としてのジェスチャ認識

佐藤 洋平　　杉原 厚吉
東京大学大学院情報理工学系研究科数理情報学専攻

Human-computer interaction に関連した研究でジェスチャの認識があるが、本稿ではその中でも PC へのマウスやキーボード以外の入力手段としてのジェスチャ認識について研究したものである。ジェスチャの認識の手続きには大きく画像処理、特徴抽出、認識の３つのステップに分けられるが、今回は特に特徴抽出と認識において先行研究の改良を行った。特に特徴抽出では判別分析等の統計手法を用い、結果として先行研究の手法より高い認識率を得ることのできるジェスチャのセットを構成することができた。

Hand gesture recognition as multi-modal interface for PC

Yohei Sato　　Kokichi Sugihara
Mathematical Informatics，Graduate School of Information Science and Technology，University of Tokyo

Hand gesture recognition is one of the most important researches for Human-computer interaction, and this paper presents a gesture recognition approach for an interface between human and a PC, which will replace current interface tools such as a mouse and a keyboard. A typical process for hand gesture recognition consists of three steps: image processing, feature extraction and recognition. In this paper, we improve the previous work especially on the steps of feature extraction and recognition. In particular, in the feature extraction step, we exploit statistical methods including discriminant analysis. As a result, we could construct gesture set which can be recognized in a higher recognition rate than the previous approach.

1　はじめに

ジェスチャの認識は human-computer interaction (HCI) への利用の他，ビデオの圧縮，動画像検索，さらには顔の表現の認識，唇の動きの認識，人の動作の認識のような分野とも技術的関連性があるため研究が活発になっている．しかし，ジェスチャの認識は少なくとも次の 3 つの困難がある:

- 形や動作が時間とともに変化する.
- 手の形は複雑で剛体ではない物体である.
- 人によってわずかに動作に違いがある.

ジェスチャの認識の手法には大きく分けて 2 種類ある．3 次元モデリングを基に認識する手法と，手の動きによる画像上の変化をパラメータ化する手法である．前者は包括的にジェスチャの動きを扱えるという利点はあるが，3 次元モデリングは計算コストが高い上に，モデリングするのに多くの近似を用いているので，モデルの復元のパラメータが不安定であることから，本稿では後者の手法を用いている．ジェスチャの認識の過程を図 1 に示す.

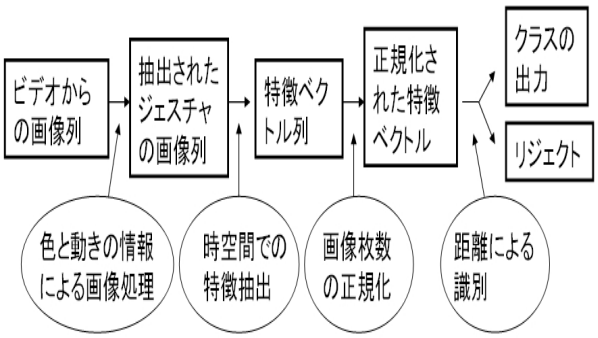

図 1: ハンドジェスチャが認識されるまでの手順

手の姿勢認識や HCI に関するサーベイは [1], [2] にある．静的な手の姿勢のみの認識を対象としたものと

して，例えば [3] では静的な形のモデル 5 種類の手の姿勢が認識され，実際マウスの代用としてシミュレーションが行われている．そして [4] ではゲームの操作へ応用している．ビデオストリームからのジェスチャの認識に焦点を当てたものに [5] があり，3 次元での手の速度や形の特徴量をとり，隠れマルコフモデルを用いて 5 種類のジェスチャを認識するシステムを構築している．さらに他の分野への貢献として手話の認識がある．これらは，3 次元での動き解析や隠れマルコフモデル等を用いて 40 種類以上のアメリカサインランゲージに対して 90 %以上の認識率を達成している [6]．本稿で参考にしているジェスチャ認識のシステムは [7] であり，ここでは 12 種類のジェスチャに対して最高 90.83 %の認識率を達成し，実際に PC への入力手段としてリアルタイムシステムが構築されている．

他の多くの研究者が述べているように，これらのジェスチャ認識システムで使われてる手法の有効性を直接比較するのは困難である．何故なら，応用先がそれぞれ異なるため，ジェスチャのセットの複雑さ，背景の仮定やシステムの設定などが違うためである．よって他のシステムとの比較は直接出来ないが，[7] の特徴を挙げると次のようになる．

- 手話認識を除いた他のシステムと比べて多くのコマンドセットを持っている．
- 高い認識率をもち，かつ実時間で制御できる．
- ジェスチャ認識はカメラの映る範囲ならカメラに対する位置に依存しない．
- ジェスチャは手に何らかのマークをつけることや特定の背景や照明の状態を仮定していない．
- PC とビデオカメラのみが必要となる装置である．

[7] で提案された手法は大きく分けて，画像処理，特徴抽出，認識の 3 ステップから成る．この先行研究の特に特徴抽出と認識部を改良することにより，本稿では 14 種類のジェスチャを，計算コストをあまり変えずに，より高い認識率で認識することが出来た．

2　手の領域の抽出

まず手の領域を抽出することを考える．手の抽出で最もよく使われるのが，色の情報と動きの情報である．色による識別ではヒストグラムマッチングか又は単純に look-up table が用いられる．これらは一般に肌の色のばらつきや照明の状態に左右されやすいという欠点があるが，HSI 表色系や L*a*b*表色系に変換することによってある程度肌の色のばらつきや照明の状態に対してロバストになる事が知られている．本実験では先行研究同様 HSI 表色系を使用する．HSI 表色系とは色を色度明度彩度の尺度によってあらわす表色系である．また最近では手の色を RCE (Restricted Coulomb Energy) ニューラルネットワークで学習して成果をあげた例も報告されている [8]．今回は HSI 表色系による色度彩度 look-up table を学習によって作成し，手の領域の識別に用いた．

動きの情報を用いるために，具体的には，連続した画像同士での明度の差分をとり，閾値処理することによって識別する．今回はその色による識別と動きによる識別の 2 つを満たした画素を手の領域とする．また穴や雑音を除去するため，モルフォロジー演算を適用する．詳しくは [7] を参照していただきたい．

3　時空間での手の特徴抽出

ここでは，前章によりあるジェスチャがなされた時のその一連の画像列を取得し，かつ前章で述べた画像処理手法により，手の領域が抽出されたとして話を進める．この場合，なされたジェスチャの画像枚数には幅があるが, その対処法は後で述べる画像枚数の正規化のところで述べる．さしあたりその画像列の１つ１つの画像間での特徴抽出を行うことが本章の目的である．なお，[7] では，以下に述べるオプティカルフローによる特徴抽出の他に，空間での特徴抽出として，各画像の手の領域を 2 次元の楕円とみなして，主軸の傾きや主軸長と副軸長の比を特徴量としているが，ここでは割愛する．

3.1　オプティカルフローを用いた動きの特徴抽出

ここでは特徴抽出のためのオプティカルフローを用いた推定を行い，それによって得られたパラメータを画像間の特徴量とする．オプティカルフローを用いた推定法は数多く存在するが，ここでは簡潔で計算コストが低く安定性に比較的優れている方法を用いる [9]．オプティカルフローとは画像間で生じる画像上での濃度の速度分布のことである．オプティカルフローでは，微小時間での明るさは一定という仮定が用いられ，それは以下の式で表せる．

$$I(\boldsymbol{x}, t) = I(\boldsymbol{x} - \boldsymbol{u}\Delta t, t + \Delta t), \quad \forall \boldsymbol{x} \in \mathrm{R} \qquad (1)$$

ここで $I(\boldsymbol{x}$,t) は時間 t，位置 $\boldsymbol{x}$ での明るさを表し，R は対象とする物体の画素集合，$\boldsymbol{u} = [u, v]^T$ の各要素はそれぞれ $\boldsymbol{X} = [x, y]^T$ の画素位置における水平，垂直方向の動きの速度である．さらにここでは u，v を planar model で以下のようにモデル化する．

$$\begin{aligned}\boldsymbol{u}(x,y) &= \begin{bmatrix} u(x,y) \\ v(x,y) \end{bmatrix} \\ &= \begin{bmatrix} a_0 + a_1 x + a_2 y + a_6 x^2 + a_7 xy \\ a_3 + a_4 x + a_5 y + a_6 xy + a_7 y^2 \end{bmatrix}\end{aligned}$$

ここで $a_i (i = 0, ..., 7)$ は定数である．(1) の右辺をテイラー展開し 2 次以上の項を取り除くと

$$\nabla \boldsymbol{I}^T \boldsymbol{u} + I_t = 0, \quad \forall \boldsymbol{x} \in \mathrm{R}$$

ここで $\nabla \boldsymbol{I} = [I_x, I_y]^T$ は勾配ベクトル，I_x, I_y, I_y は位置座標と時間による偏微分を表す．その時パラメータ集合 $\boldsymbol{a} = \{a_0, a_1, ..., a_7\}$ は次の目的関数を最小化することによって推定される:

$$E(\boldsymbol{a}) = \sum_{\boldsymbol{x} \in \Re} \rho(\nabla \boldsymbol{I}^T \boldsymbol{u}(\boldsymbol{a}) + I_t, \sigma).$$

ここで ρ はあるロバストなエラーノルムであり，σ はスケールパラメータである．コンピュータビジョンで数多くの ρ が使われているが，ここでは German-McClure 関数

$$\rho(r, \sigma) = \frac{r^2}{\sigma^2 + r^2}$$

を用いる．(1) の最適化手法も数多くあるがここでは continuation method を用いた SOR (simultaneous over-relaxation) 法を使用する．特に $E(\boldsymbol{a})$ を最小にする $n+1$ 回反復目の更新の式は次のようになる:

$$a_i^{n+1} = a_i^n - \omega \frac{1}{T(a_i)} \frac{\partial E(\boldsymbol{a})}{\partial a_i}, \quad i = 0, 1, ..., 7.$$

ここで ω は緩和パラメータであり $0 < \omega < 2$ の時収束する．$T(a_i)$ は $E(\boldsymbol{a})$ の 2 階偏微分の上限である．

$$T(a_i) \geq \frac{\partial^2 E(\boldsymbol{a})}{\partial a_i^2}, \quad i = 0, ..., 7.$$

σ は各反復で一定比率で小さくする．このことによって最初は全ての領域が解に関わっているのがだんだんと雑音の影響が減っていく．また，大きい動きに対応するためにガウスピラミッドを用いた coarse-to-fine strategy を行う必要がある．

ここで求めたパラメータはそれぞれ独立に幾何的な解釈を持った要素に分解できることが知られており次のようになる：

- 水平移動 $(m_1) : m_1 = a_0$.
- 垂直移動 $(m_2) : m_2 = a_3$.
- isotropic expansion $(m_3) : m_3 = a_1 + a_5$.
- 目的画像の歪み (m_4):

$$m_4 = \sqrt{(a_1 - a_5)^2 + (a_2 + a_4)^2}.$$

- 2-D rigid rotation $(m_5) : m_5 = -a_2 + a_4$.
- yaw $(m_6) : m_6 = a_6$.
- pich $(m_7) : m_7 = a_7$.

3.2 更なる時空間での特徴抽出

オプティカルフローによる特徴抽出は比較的よい認識率をもたらすが，これにさらによい特徴量を加えることによってより高い認識率を導き出すことを目指す．適した特徴量であるかどうかの観点において大切な事のひとつに不変性がある．特にここで大切となる不変性を並べると

- ジェスチャの動きの速さによる不変性
- 照明等による色の不変性
- カメラに対する手の位置の不変性
- カメラの遠近差等による手の大きさの不変性

である．さらに，識別するのに適した特徴量とは，クラスごとの分布がなるべく交じわらずかつクラスごとにまとまった特徴量が，識別するのに適した特徴量と言える．オプティカルフローによる特徴量は上に挙げた不変性を全て備えている．しかし，[7] では識別するのに適した特徴量抽出を考慮してはいない．ここで，上に挙げた不変性をなるべく持ちつつ，識別するのに適した特徴量を目指した特徴抽出の手法を述べる．

与えられた画像列の各画像を画素分だけの次元を持つ特徴ベクトルとする．本実験では 160 × 120 の画像を縮小させた 20 × 15 の画像を実際には使うので，特徴空間は 300 次元である．しかし，これをこのまま特徴量とするには次元が多きすぎるので何らかの基準に従って次元を削減する．ここでは KL 展開や判別分析といった統計的手法を用いることにする．KL 展開は与えられたデータを用いて情報をなるべく失わずに次元を圧縮する手法であり，クラスという事を特に意識しない教師なし学習の一つである．また判別分析は与えられたクラスラベル付きのデータをある基準にした

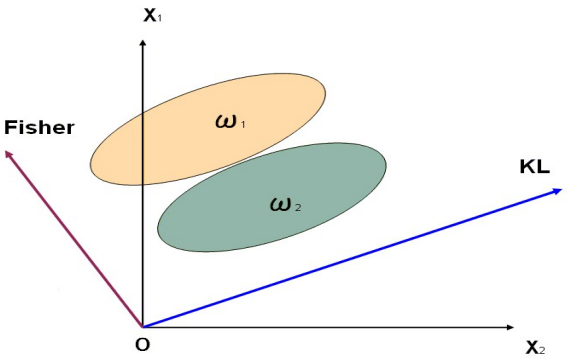

図 2: 二次元空間にクラス ω_1, ω_2 が分布している場合の次元削減．KL，Fisher はそれぞれ KL 展開と判別分析による新しい軸．

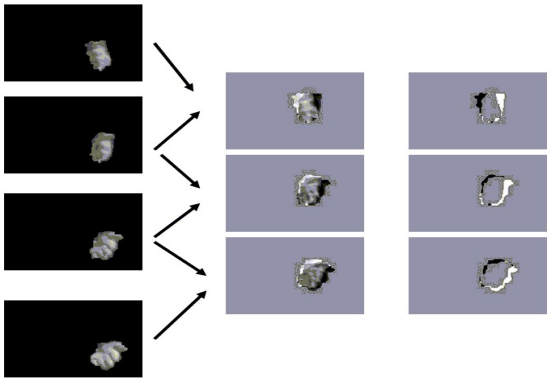

図 3: 左の画像列があるジェスチャの画像処理後の画像列の一部．真ん中の画像列は，位置の正規化後，その差分をとった値をもつ画像列．右は明るさに対して正規化した画像列．

がって最も分離するように次元削減を行う教師あり学習法である (図 2)．

そこであらかじめ集めたジェスチャのサンプル画像列を用いて，これらの次元削減を行う写像を求める．KL 展開では，全てのジェスチャ画像列の各画像を一元的に学習することによって，次元削減のための線形写像を求める．線形判別法ならばクラスごとのジェスチャ画像列の各画像を一元的に学習することによって次元削減のための線形写像を求める．そして識別したいジェスチャの画像列が入力された時には，事前に学習によって得られた写像に代入することによって，次元が削減された特徴ベクトルが得られる．本稿では 300 次元から 4 次元に射影させ，それを特徴量とした．しかし，このままでは不変性を保っていない．そのために不変性をもつように次のように工夫をする．

まず動きの速さについての不変性を保つために，対象の画像と前または後の画像との差分をとった画像を学習や写像を適用する対象の画像とする．これによって次式が成立する．

$$F(\boldsymbol{x}_i - \boldsymbol{x}_j) + F(\boldsymbol{x}_j - \boldsymbol{x}_k) = F(\boldsymbol{x}_i - \boldsymbol{x}_k), \quad i \leq j \leq k \tag{2}$$

ただし，F は n 次元空間から m 次元空間への線形写像 $(m \leq n)$，$\boldsymbol{x}_i$ は時刻 i での画像によって得られた特徴ベクトルである．この性質と特徴ベクトル列の長さの正規化の手法を組み合わせることによって，動きの速さに対する不変性が得られる [3]．

次に，明るさや色に対する正規化は，画像列をあらかじめ手の領域か否かの 2 値画像にしてから，各々差分をとることによって達成される．この時，差分画像の特徴ベクトルは結果的に 3 値化されている．これによって (2) の式を成立させつつ明るさの正規化も出来る．

次に手の位置の正規化がある．これを誤ると特徴ベクトルの値全体がずれたものになってしまうので深刻である．ここでは画像列の最初の差分画像の手の画素の位置の重心を画像の中心にし，それ以後の画像もその分だけ位置を移動させるという方法をとった．なお毎回差分画像の手の領域を画像中心に正規化すると，動きの速さに対する不変性が成立しなくなるから，それはやってはいけない．

カメラからの距離による手の大きさに対する正規化の方法は明確なものはないが，画像を縮小することによってその影響をなるべく小さくする事が出来るだろう．今回は 160 × 120 の画像を 20 × 15 に縮小している．

これによって様々な不変性を考慮した新たな特徴抽出ができるが，欠点として手の位置の正規化がある．先に提案した正規化では画像列の最初の画像のみから計算された移動距離を残りの画像全てに適用するため，最初の画像の正規化が雑音等でうまくいかない場合，それ以降の全ての画像に影響を及ぼしてしまい，やや耐久性にかける．ここでもう一つの手法として，手の動きの速さは一定と見なして，各画像ごとに手の領域を画像中心に正規化する方法が考えられる (図 4)．これによって手の位置に関してより耐久性の高いものとなる．さらにこの場合は適用する統計手法は線形でなくてもよいので実験ではカーネル関数を用いた統計手法 [10]，[11] も適用した．

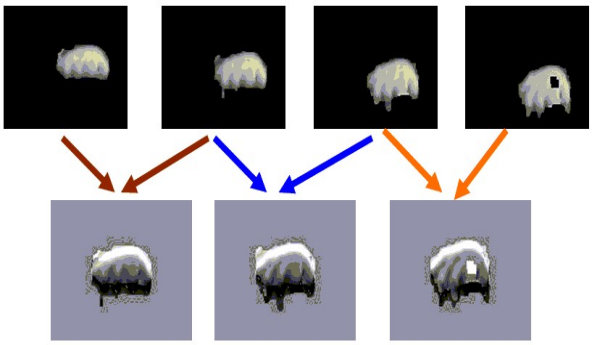

図 4: 動きを一定とみなして各ジェスチャの画像で位置を正規化した．上が画像処理後の手の画像列で，下がさらに差分し位置を正規化した後の画像列．

4　ジェスチャの認識

ここでは，画像処理，特徴抽出，認識の３ステップのうち最後の認識の部分について述べる．学習サンプルが十分に多い場合はニューラルネットワークや隠れマルコフモデルがよく利用されるが，それらは学習すべきパラメータが多く，学習に多大な時間がかかるという問題がある．よってここでは一度ジェスチャの特徴ベクトルの長さを正規化して，それからテンプレートマッチングするという手法をとる．またこの特徴ベクトルの長さを，正規化の仕方によって特徴量が動きの速さに対して線形補間の範囲内で不変にすることが出来る．

4.1　Time Normalization

ジェスチャの画像列の枚数は人によっても違い，またその時その時によっても異なる場合が多くあるので，それにも対処しなければならない．音声認識の分野では dynamic time warping (DTW) がよく使われている．しかし音声認識とは異なり，ジェスチャは約 10Hz という低い率でサンプリングを行うことと，音声認識と比べ長さが不安定であり，また DTW は不安定な程度が高いと十分に効果が出ないことと，計算コストも比較的高いことにより，[7] では代わりの手法を提案しており，本稿もそれに従う．概要を言うと，まず全ての画像列の枚数を正規化するために，正規化後の枚数を決め，その枚数にあわせて特徴ベクトルも線形補間する．以下にそのアルゴリズムを以下に示す．

Motion-Feature-Warping Algorithm

入力:$m_{i,t}, i = 1, ..., 7, t = 0, ..., L-1.$
出力:$\tilde{m}_{i,k}, i = 1, ..., 7, k = 0, ..., K-1.$

手続き

1. $A_{i,t} = \sum_{n=0}^{t} m_{i,n}, \quad t = 0, ..., L-1.$

2. $t_k = \frac{L(k+1)}{K} - 1, \quad k = 0, ..., K-1.$

3. $\tilde{A}_{i,t_k} = (t_k - \lfloor t_k \rfloor) A_{i,\lceil t_k \rceil} + (\lceil t_k \rceil - t_k) A_{i,\lfloor t_k \rfloor}.$

4. $\tilde{m}_{i,k} = \tilde{A}_{i,t_k} - \tilde{A}_{i,t_{k-1}}$

ここで $\tilde{A}_{i,-1} = 0.$

これを行うことによって，線形補間の範囲内では，アルゴリズム上では動きの速さによらない特徴量を生成することが出来る．またこれは，前章で述べたように，統計的手法を用いた新しい特徴抽出の場合にも合理的に適用される．L は元のジェスチャの画像列の枚数，K は正規化する枚数，$\lfloor \cdot \rfloor, \lceil \cdot \rceil$ はそれぞれ小数点以下切捨て，切り上げである．また，動きの速さを考慮しない場合の正規化は動きの総量を考慮せず単に線形補間したものである．

ここで正規化する長さが問題になるが，[7] で最も結果が良かった４フレームに今回の実験では合わせることにする．

4.2　距離の測定

前節の特徴ベクトルの長さの正規化を行った後，どの動作に分類するかを決めるために各クラスのテンプレートと特徴ベクトル同士の距離を測る必要がある．代表的なものにユークリッド距離やマハラノビス距離やベクトル角による距離がある．実験ではベクトル角による距離とユークリッド距離の二つを使用した．[7] ではベクトル角による距離を使用している．ベクトル角による距離とは，$\boldsymbol{A} = (a_{ij})_{10\times K}$,$\boldsymbol{B} = (b_{ij})_{10\times K}$, K は正規化された長さとすると，$\boldsymbol{A}$ と $\boldsymbol{B}$ の距離は

$$D(\boldsymbol{A}, \boldsymbol{B}) = 1 - \frac{\sum_{j=0}^{K-1}\sum_{i=0}^{9}(a_{ij})(b_{ij})}{\sqrt{\sum_{j=0}^{K-1}\sum_{i=0}^{9}(a_{ij})^2}\sqrt{\sum_{j=0}^{K-1}\sum_{i=0}^{9}(b_{ij})^2}}$$

と定義される．また，実験では特徴ベクトルの各要素はあらかじめ得られた学習サンプルによる標本標準偏

差で正規化しておく．これは直感的には特徴空間での特徴点間の距離が最小になるようにしていると解釈できる [12].

4.3 テンプレートによる認識

ここではテンプレートマッチングの仕方について具体的に議論する．多くの学習データが与えられた時，個々のバリエーションが大きい時には，単に学習データの特徴ベクトルの各要素をそれぞれ平均するというのはあまり好ましくない．よって [7] では音声認識などでも使われる minimax selection 技法を用いている．しかし，そもそも代表ベクトルによるテンプレートマッチングは，ベイズの識別で各クラスの事前分布と共分散行列が等しく，かつ，共分散行列が等方的である場合に相当するので，各クラスで特徴ベクトルにばらつきがある場合はふさわしいとはいえない．よって各クラスで特徴ベクトルにばらつきがある場合にも比較的ロバストなｋ-nearest neighbor 法 (k-NN 法) と minimax selection 技法を実験では比較した．ｋ-NN 法とは学習データを全て記憶しておき，新しい入力を識別する時には，記憶されてる学習データの中から入力に近い順にｋ個をとり，多数決をとるという方法である．特に今回はｋのうち一定以上の占有率を持つクラスがあればそのクラスを採用し，それ以外はリジェクトするという方法をとった．

k-NN 法の欠点として，学習サンプルを全数記憶するため，学習サンプルが増えるにつれて記憶容量が大きくなるということと，新たな入力サンプルとの距離を測り，最小の距離を持つ学習サンプルを決定する際の計算コストも特徴次元や特徴サンプルが増えると共に増大するということがあげられる．しかし，収集した学習サンプルから識別に不必要なサンプルを削減する編集アルゴリズムや，学習サンプルに，特殊なデータ構造を導入することによって高速に最近傍点を検索するというアルゴリズムが多数提案されており，さらに近年の計算機の性能の飛躍的向上により，実用性の面でも現実的な手法になってきている．

4.4 ジェスチャかどうかの判定

実際にオンラインの時に得られた画像列がジェスチャであると判定するためには，以下のような基準を用いる [7].

- 画像処理の結果，手と判断された領域が画像列に現れる．
- 動いてる物体の中で，手と判断された領域は画面中で最も大きな領域を占める．
- 手は一度止まってから短時間スムーズに動き，そして最後にスローダウンするという 3 ステップを踏む.
- 動いているステージは L_1 フレーム以上 L_2 フレーム以下である．

さらにジェスチャの種類を特定するときに，(1) 距離による閾値処理，(2) k-nearest neighbor の識別の場合はｋのうち過半数を占める，という 2 つの条件を満たすクラスが存在したときのみそれを出力し，それ以外はリジェクトする．特に今回は k-nearest neighbor の識別を用いることによって新たにリジェクト領域を増やしている．

5 実験結果

実験は WindowsXP のもとで，Pentium4 の 2.4GHz の PC で行った．この PC には USB カメラが取り付けられている．実験で用いたジェスチャの種類は表 5 のように 14 種類である．各ジェスチャの画像列は 10HZ でカメラから撮られたものであり，各画像は 160 × 120 ピクセルで，1 ピクセルは 24 ビットである．実験サンプルは 3 人の人のジェスチャを各 3 セットの計 9 サンプルでおこなった．学習方法としてはジャックナイフ法を用いた．また今回の実験結果は定量的な評価をするため全てオフライン行ったものである．ベクトル角の距離を用いた実験結果を表 1，ユークリッド距離を用いた実験結果を表 2 に示す．また動きの速さを一定と仮定した特徴抽出の場合の実験結果を表 3，4 に示す．またカーネル関数にはガウシアンカーネルと多項式カーネルを用いた．これらのカーネル関数はパラメータの設定が必要だが，いくつかのパラメータで実験して，その中で一番結果が良かったカーネル関数の一番よいパラメータでの実験結果を表に載せた．minmax を用いた識別と k-NN を用いた識別では，k-NN を用いた識別の方が k-NN で多数決をとってる分リジェクト領域があり，誤識別率が下がるので，比較しやすいようにこれらの表では minmax を用いた識別が k-NN を用いた識別と同じ誤識別率になる距離によるリジェクト領域を作り調節した．今回の実験ではｋを 5 に設定し，3 以上占めるクラスがなければリジェクトとした．

実験結果からわかるように，本稿で提案した特徴抽出とｋ-NN による識別にすることによって，先行研究の手法よりも認識率が向上している．minmax を用い

表 1: 距離としてベクトル角の余弦を用いた実験結果．+KL 展開，+判別法はそれぞれ先行研究に KL 展開又は線形判別法による特徴量を加えた実行結果.

	minmax を用いた識別		k-NN を用いた識別	
	認識率	誤識別率	認識率	誤識別率
先行研究手法	83.9%	8.0%	87.5%	8.0%
+KL 展開	86.6%	5.4%	91.1%	5.4%
+判別法	90.2%	5.4%	92.0%	5.4%

表 2: 距離としてユークリッド距離を用いた実験結果.

	minmax を用いた識別		k-NN を用いた識別	
	認識率	誤識別率	認識率	誤識別率
先行研究手法	83.9%	10.7%	86.4%	10.7%
+KL 展開	80.4%	6.3%	90.0%	6.3%
+判別法	83.9%	6.3%	91.2%	6.3%

表 3: 動きの速さを一定と仮定した場合で距離としてベクトル角の余弦を用いた実験結果．+KL 展開，+判別法, +KKL，+KF はそれぞれ先行研究に KL 展開，線形判別法，カーネル KL 展開，カーネル線形判別法による特徴量を加えた実行結果.

	minmax を用いた識別		k-NN を用いた識別	
	認識率	誤識別率	認識率	誤識別率
+KL 展開	87.5%	6.3%	91.1%	6.3%
+判別法	90.2%	5.4%	92.0%	5.4%
+KKL	86.6%	4.5%	92.9%	4.5%
+KF	88.4%	6.3%	91.1%	6.3%

表 4: 動きの速さを一定と仮定した場合でユークリッド距離を用いた実験結果.

	minmax を用いた識別		k-NN を用いた識別	
	認識率	誤識別率	認識率	誤識別率
+KL 展開	84.8%	4.5%	92.9%	4.5%
+判別法	84.0%	2.7%	95.5%	2.7%
+KKL	80.4%	3.6%	91.1%	3.6%
+KF	85.7%	7.1%	92.0%	7.1%

たテンプレートマッチングと k-NN 識別では，k-NN 識別の方が距離によるリジェクト領域の他に，k の占有率もリジェクト領域の対象にしている分，誤識別率も考慮すると k-NN 識別の方がよい結果になった．本稿で提案した手法ごとに比べてみると，動きの速さ一定のモデルの方が動きの速さを考慮したモデルよりもいい結果となった．この程度の差なら誤差の範囲ではあるが，あえて理由を挙げるとすると，動きの速さ一定のモデルの方が各画像ごとに手の位置の正規化をおこなっている分，手の位置の正規化に関する頑強性が強くなった事が考えられる．ユークリッド距離とベクトル角の余弦距離では，k-NN 識別においてはユークリッド距離の方が好ましい結果が出ていたが，minmax を用いたテンプレートマッチングでは逆転している．これはユークリッド距離での距離の閾値処理があまりうまくいかなかったことが原因にあり，実際，あるジェスチャのクラスでは距離が概して低く出たり，その逆の場合になるクラスもあり，そのことによる影響は大きい．一方ベクトル角の余弦距離は値域が -1 から 1 であることから，比較的閾値を設定しやすく，かつユークリッド距離のようなクラスごとのばらつきはあまり見られない．よって，リジェクト領域を作るため距離による閾値処理をする場合には，ベクトル角の余弦距離の方が適していると言える．

6 まとめ

本稿では，先行研究でなされている PC へのコマンド入力手段としてのジェスチャの認識のアルゴリズムの改良を行った．特に特徴抽出部と認識部を改良することにより，実験では先行研究の手法を上回る結果を得ることが出来た．今後の課題の一つは，実用面を考えたときの新たなシステムテストの概念を導入することである．今のテストでは，ジェスチャがされたと仮定した時の定量的なテストにすぎない．特にオンラインでの定量的なテストの方法などを確立することによって，実用的な観点からみた新たな要求も生まれてくることになるだろう．

参考文献

[1] V. Pavlovic, R. Sharma, and T. S. Huang, Visual interpretation of hand gestures for human-computer interaction: A review, *IEEE Trans. Pattern Analysis and Machine Intelligence* 19, 1997, 677-695.

[2] A. Pentland, Looking at people: Sensing for ubiquitous and wearable computing, *IEEE Trans. Pattern*

図 5: 実験で使用する 14 種類のジェスチャ.

Analysis and Machine Intelligence 22, 2000, 107-119.

[3] T. Ahmad, C. J. Taylor, et al., Tracking and recognizing hand gestures, using statistical shape models, *Image Vision Comput.* 15, 1997, 345-352.

[4] W. T. Freeman, K. Tanaka, J. Ohta, et al., Computer vision for computer games, in *Proceedings of the Int'l Conf. Automatic Face and Gesture Recognition, Killington*, 1996, 100-105.

[5] D. A. Becker, Sensi: A Real-time recognition, feedback and training system for T'ai Chi gestures, Technical Report TR-426, MIT Media Lab, 1997.

[6] C. Vogler and D. Metaxas, ASL recognition based on acoupling between HMMs and 3D motion analysis,in *Proceedings of the Int'l Conf. Comput. Vision, Bombay*, 1998.

[7] Y. Zhu and G. Xu: A real-time approach to the spotting, representation, and recognition of hand gestures for human-computer interaction , *Comput. Vision Image Understand* 85, 2002, 189-208.

[8] X. Yin, D. Guo, M. Xie, Hand image segmentation using color and RCE neural network, *Robotics and Autonomous Systems* 34, 2001, 235-250.

[9] M. J. Black and P. Anandan: The robust estimation of multiple motions: Parametric and piecewise-smooth flow field. *Comput. Vision Image Understand* 63, 1996, 75-104.

[10] B. Schölkopf, A. Smola, and K.-R. Müller: Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation* 10, 1998, 1299–1319.

[11] K.-R. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Scholkopf: An introduction to kernel-based learning algorithms, *IEEE Trans. Neural Networks* 12(2), 2001, 181-201.

[12] 石井健一郎，上田修功，前田英作，村瀬洋: わかりやすいパターン認識．オーム社，1998.