局所相関演算に基づくオプティカルフローを用いた身振り動作の認識手法

西川 敦[†] 大西映生[†,☆] 西村正典[†]
平野敦士[†,☆☆] 小荒健吾[†] 宮崎文夫[†]

我々は，片手の身振り動作を撮影した連続動画像の局所相関演算に基づくオプティカルフローから算出される手の移動方向の変化率（本稿ではこれを“曲率”と呼ぶ）に基づいて，身振り動作をあらかじめ用意されたいくつかの「基本動作」の連続として分割・記号化する方法を提案する．また，オプティカルフロー検出時のいくつかのパラメータを“身振りを行う手とカメラ間の距離”に基づいて適応的に調整する方法もあわせて提案する．これらの手法を用いれば，広い距離範囲にわたって，不特定の動作者に対して高い認識率を達成可能な身振り認識システムを構築できる．身振り動作者とカメラ間の距離の大きな変動に対する認識システムの頑健性は，動作者をとらえるカメラを自律移動ロボットに搭載することを想定した場合，きわめて重要なポイントとなるにもかかわらず，従来はほとんど議論されなかった点である．手を折り返す，曲げる，回転させる，弧を描く，といった人間にとって直観的に理解しやすい複数の基本動作の組合せからなる4種類の身振り動作を対象とした認識実験を行った結果，提案手法が，手–カメラ間距離1～8mの広い範囲にわたって，6人の被験者平均で85%以上の高い身振り認識率を保持できることを確認した．

Recognition of Human Gestures from Optical Flow Based on a Correlation Method Between Local Image Regions

ATSUSHI NISHIKAWA,[†] AKIO OHNISHI,[†,☆] MASANORI NISHIMURA,[†]
ATSUSHI HIRANO,[†,☆☆] KENGO KOARA[†] and FUMIO MIYAZAKI[†]

In this paper, a new technique for description and recognition of human gestures is presented. A method is first proposed to transform the input gesture pattern into an ordered sequence of simple *basic motions* which are easy for human to understand. It is based on the rate of change of gesture motion direction (referred to as “curvature”) estimated from optical flow based on a correlation method between local image regions. Another method is then proposed for adaptive selection of flow detection parameters based on the distance between the gesturer's hand and camera. Proceedings from these two methods, a real-time gesture recognition system is shown that can achieve high recognition rates(overall 85% or more) for unspecific gesturers over a wide range of the gesture distance (1～8m).

1. はじめに

人間の身振り動作を非接触かつ無標点で自動認識する技術は人にやさしいヒューマンインタフェースシステムを構築するうえできわめて重要である．近年，人間にとってより自然で分かりやすい機械（コンピュータ，ロボット）との対話を目的として，動作者の自然な身振りをカメラで撮影し，その結果得られる動画像系列を解析することにより身振り認識を実現しようとする研究がさかんに行われている[1)]．

動画像を用いた最も基本的な身振り認識法は，動画像を構成する1枚1枚の画像から特徴量をボトムアップ的に抽出することにより得られる身振りの入力パターン（特徴量の時系列）と特定の動作者の例示によりあらかじめ作成しておいた標準パターン（モデル）とのマッチングに基づく方法である．代表的な手法として，Dynamic Time Warpingを用いた身振り認識法[2)]，連続DPを用いたスポッティング認識法[3)]，確率的な状態遷移モデルの一種である隠れマルコフモデル（HMM）を用いた身振り認識法[4)~6)]などがある．一方，人間の動作を単なる特徴量の時系列パターンとしてとらえるのではなく，複数の代表的な状態（基本

† 大阪大学大学院基礎工学研究科
Graduate School of Engineering Science, Osaka University
☆ 現在，松下電工株式会社
Presently with Matsushita Electric Works, Co. Ltd.
☆☆ 現在，アンダーセンコンサルティング株式会社
Presently with Andersen Consulting, Co. Ltd.

的なポーズ）で構成される一連の動きとしてとらえることによりロバストな身振り認識を実現しようとする研究がある．牛田ら[7]やBobickら[8]は，ファジーメンバシップ関数を利用して特徴量の時系列データを代表的な状態の遷移パターンに変換した．長屋ら[9]は，身振り画像の時間変化がパターン空間に描く身振りに固有の軌跡を「動作軌跡」と呼び，この動作軌跡の曲率の極大・極小値に基づいて身振り動作を代表的な構成要素に分割する方法を示した．渡辺ら[10]は，身振り画像系列をKL展開して低次元のジェスチャ空間を構成し，その空間上で表現されたジェスチャ曲線から特徴的な点をルールベース的手法により抽出し，身振りをこれら特徴点の系列として記号化した．

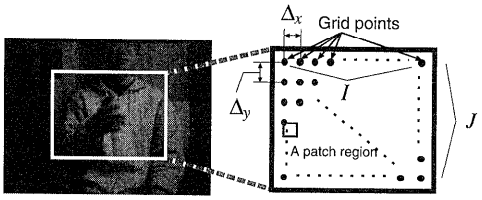
ここまでに紹介したタイプの手法[2)~10)]はいずれも1枚1枚の画像から独立に得られる“静的”な特徴量をベースにして身振りパターンを生成している．そのため，モデル生成時と背景や照明条件が大きく異なる状況下では利用できないことがある．この問題を克服するべく，最近，フレーム間差分やオプティカルフローなどから得られる“動き情報”をベースにした身振り認識法がいくつか提案されている．西村ら[11]は，連続するフレーム間での時間差分の2値化処理により動き領域を求め，さらにこれらを低解像度化することにより環境の変化にロバストな身振り認識を達成した．Davisら[12]は，画素が時間の関数となる2次元画像モデルMHI（Motion-History Image）を提唱し，人間の動きの認識に利用した．このモデルは，フレーム間差分の2値化により得られる動領域を時間履歴を考慮して重ねあわせることにより生成される．Kuritaら[13]の身振り認識手法では，まず時間平均画像を作成し，各フレームごとにこの画像との差分をとった後，線形自己回帰モデルを適用して時間方向に情報圧縮した背景に依存しない画像（PARCOR画像）を作成，利用している．一方，オプティカルフローを用いた身振り認識法としては最近次の2つの手法が提案されている．畠ら[14]の方法では，オプティカルフローにより得られる画像中の各点の動きベクトル群を身振りの特徴量とし，これらをKL展開により次元圧縮した後にHMMで解析することによって楽器を演奏する身振りの認識を行っている．Cutlerら[15]は，フレーム間差分とオプティカルフローを併用した．まずフレーム間差分値の大きな領域内のオプティカルフローのみを利用し，各動きベクトルの方向に基づいてそれらをたかだか2個のブロブにクラスタリングする；ブロブの大きさや方向情報を用いたルールベース的な手法により音楽のテンポを調節・指揮するためのいくつかの身振りの実時間認識に成功している．

我々は，片手の身振り動作を撮影した連続動画像のオプティカルフローから算出される手の移動方向の変化率（本稿ではこれを“曲率”と呼ぶ）に基づいて，身振り動作をあらかじめ用意されたいくつかの「基本動作」の連続として分割・記号化することにより，不特定の動作者に対しても高い認識率を達成できる新しいタイプの身振り認識手法を提案する．我々の手法は，身振りを複数の「基本動作」の列としてとらえるという点では文献7)~10)のタイプの方法に，一方，オプティカルフローを利用するという点で文献14)，15)の手法と深い関連がある．我々の手法とオプティカルフローに基づく従来の身振り認識法との最大の相違点は，従来法が動きベクトルの“大きさおよび方向”の時系列を直接利用して認識処理を進めているのに対して，本手法では，より身振りの個人差や癖に影響を受けないと思われる動きベクトルの“方向の変化率（角度変化）”の時系列をベースに身振りの記述を試みている点である．これは，近似的に手の動作軌跡の“曲率”を求めていることに相当するが，我々の方法では，動作軌跡を復元するプロセスを介さず，手の動きベクトルからダイレクトに“曲率”を求めており，身振りの動作軌跡の解析がベースとなる文献7)~10)のアプローチとも大きく異なる．

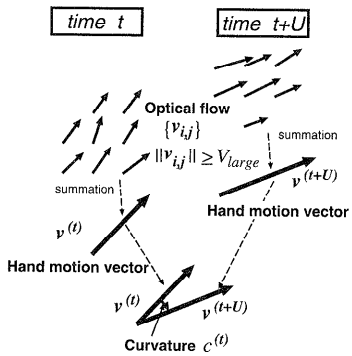
また，本稿のもう1つの特徴的な点として，人とカメラの相対距離が大きく変化しても身振りの認識率をつねに高いレベルに保持することを念頭において，アルゴリズムを構成していることがあげられる．具体的には，オプティカルフロー検出時のいくつかのパラメータを“身振りを行う手とカメラ間の距離”に応じて適応的に選択する手法の提案とその有効性の検討を行っている．これは，身振り動作者をとらえるカメラを自律移動ロボットに搭載することを想定した場合，きわめて重要なポイントとなるにもかかわらず，従来はほどんど議論されなかった点である．

本稿では，身振りを構成する「基本動作」として，手を折り返す，曲げる，回転させる，弧を描く，といった人間にとって直観的に理解しやすい8つのパターンを用意した．認識対象となる身振りのモデルは，単にこれら基本動作の組合せ（記号列）として与えられる．このアプローチによれば，従来の多くの手法のように身振りの標準パターンを作成する際に特定（あるいは複数）の動作者の身振りを入力・解析する処理がまったく必要なくなり，その結果，モデルの作成や追加・更新がきわめて簡単になるという特長もある．

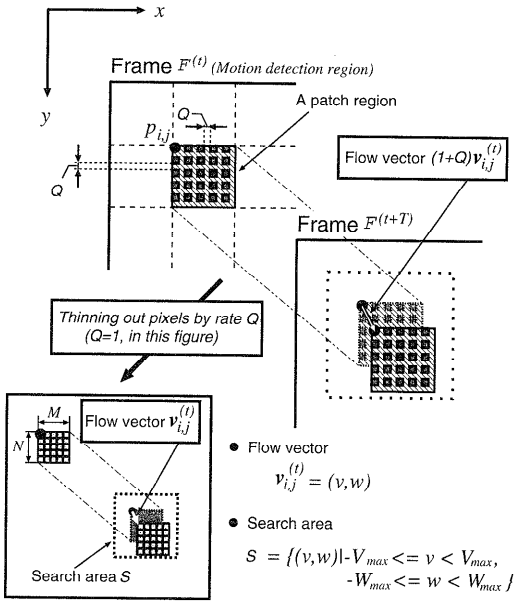
以下では，本研究における“曲率”の定義とそのオ



(a) Motion detection region and grid points



(c) Curvature computation



(b) Correlation method between local image regions

図 1 オプティカルフローと曲率
Fig. 1 Optical flow and curvature.

プティカルフローに基づく検出法，ならびに“曲率”に基づく身振り動作の分割・記号化アルゴリズムについて順次説明し，つづいて，オプティカルフロー検出用パラメータの決定方法について述べる．最後に 4 種類の身振り動作を対象とした認識実験の結果を示す．

2. オプティカルフローと曲率

本研究では，環境および認識対象に対して次の仮定をおく．

仮定 1 認識対象は片手全体で行う動きの大きな身振り動作とし，手指の形状の変化などの細かな動きは認識対象としない．

仮定 2 一連の身振り動作はカメラ視野内に設定されたあるウインドウ領域（以下，“動作検出領域”と呼ぶ．**図 1** (a) 参照）の中で行われる．

仮定 3 動作検出領域内には他の動物体は存在しない．

上記の仮定に基づき，本稿では，濃淡画像のオプティカルフローを用いて身振り動作の“曲率”を計算する．オプティカルフローを求めるための手法は，これまで多数提案されている[16]が，ここでは，①比較的ノイズやフレーム間の照明の変動等に頑健で大きな動きの検出も可能[17]，②専用のハードウェアによる実時間処理が可能[18]なことなどから，連続するフレーム間での局所相関演算に基づく方法を採用した．以下に，ここでの“曲率”の定義とその計算手順をまとめる．

(1) 図 1 (a) に示すように，動作検出領域内に $I \times J$ 個のフローベクトル検出用格子点 $p_{i,j}(x_{i,j}, y_{i,j})$ を等間隔に設定する．このときの水平および垂直方向の格子点間隔をそれぞれ Δ_x, Δ_y とする．また，局所相関演算ブロックのサイズを $M \times N$ とし，フローベクトル (v, w) の探索範囲を $\mathcal{S} = \{(v,w) | -V_{max} \leq v < V_{max}, -W_{max} \leq w < W_{max}\}$ とする．さらに，大きな身振り動作のみを検出するための閾値 V_{large} を定める．なお，画素の間引き率を Q（後述のようにこれは非負の整数である）とし，画像サンプリング周期を T で表すものとする．

(2) 各格子点 $p_{i,j}$ を基準とした連続するフレーム間での局所相関演算[19]を行うことにより，$I \times J$ 個のフローベクトルを求める．具体的には，まず，すべての $(v, w) \in \mathcal{S}$（ただし v, w は整数）に対してそれぞれ次のディストーション値 $\mathcal{D}_{v,w}$ を計算する．

$$\mathcal{D}_{v,w} = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \left| \mathcal{G}_{m,n}^{(t)} - \mathcal{G}_{v+m,w+n}^{(t+T)} \right| \quad (1)$$

ここに，

$$\mathcal{G}_{m,n}^{(t)} = \mathcal{F}^{(t)}(x_{i,j} + \delta_m,\ y_{i,j} + \delta_n) \quad (2)$$

$$\delta_l = (1 + Q) \cdot l \quad (3)$$

ただし，$\mathcal{F}^{(t)}(x, y)$ は時刻 t において撮影されたフレームの画素 (x, y) における輝度値を示す．式 (3)

は，相関演算には Q 画素おきに抜き出した画素のみを使用することを示している．

この $\mathcal{D}_{v,w}$ が最小値をとる (v,w) を時刻 t における $p_{i,j}$ のフローベクトル $\boldsymbol{v}_{i,j}^{(t)}$ と定める．この処理により得られるオプティカルフローは，実際に画像中で観測されるフローの $1/(1+Q)$ 倍の大きさになることに留意されたい（図1(b)参照）．

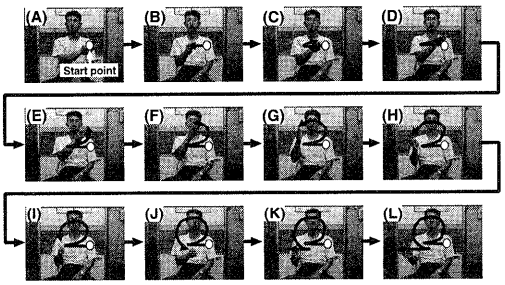
(3) こうして得られたフローベクトル群 $\{\boldsymbol{v}_{i,j}^{(t)}\}$ の中で，$||\boldsymbol{v}_{i,j}^{(t)}|| \geq V_{large}$ を満たすもの，つまり，動作者が意図的に行ったと考えられる大きな動作のみを選択し，それらのベクトルの和を時刻 t における手の動きベクトル $\boldsymbol{v}^{(t)}$ と定める（図1(c)参照）．

(4) 本研究では，手の動きベクトル $\boldsymbol{v}^{(t)}$ から測った $\boldsymbol{v}^{(t+U)}$ のなす角を反時計回りを正として $c^{(t)}$ で表し，これを時刻 t における“曲率”と定義する[20]．ただし，$-180° \leq c^{(t)} < +180°$ となるように角度を測る方向を選ぶものとする（図1(c)参照．この例では $c^{(t)} < 0$ である）．また，U は“曲率”のサンプリング周期を表す．これは，オプティカルフローの計算に要する時間，すなわち，格子点数 I, J や相関演算ブロックサイズ M, N および探索範囲 $\mathcal{S}$ の大きさに依存して決まることに留意されたい．

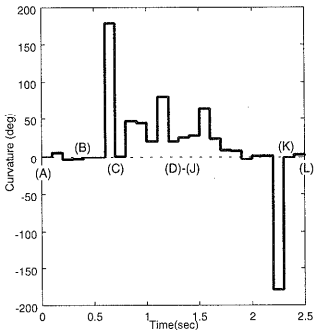
(1)で述べた11個のパラメータ（以後，これらを総称して“フロー検出パラメータ”と呼ぶ）の適切な設定のもとで上記手続きを行った一例として，**図2**(a)のような身振り動作（以下「オーム型」の身振り動作と呼ぶ）を行ったときの“曲率”の時系列 $C = \{c^{(t)}\}$ のグラフ（横軸：時刻 t，縦軸：曲率 $c^{(t)}$）を同図(b)に示す．なお，本手法の性能を左右する“フロー検出パラメータ”の決定方法については4章で詳しく述べる．

3. 身振り動作の曲率に基づく分割と記述

次に，対象となる身振り動作を“曲率”に基づいて分割し，後述する8パターンの基本動作の連続として記号化する方法について説明する．提案手法は，次の3つのプロセスから構成される：(i) **分割プロセス**（身振り動作の全区間を曲率の大きな区間と小さな区間に分割する），(ii) **併合プロセス**（いったん分割したものの，互いに隣接する区間の間での“曲率特徴”の類似性からそれらを1つの区間に併合し直す），(iii) **分類プロセス**（各分割区間をその“曲率特徴”に基づいて8つの基本動作のいずれかに分類する）．ここに，分割区間の“曲率特徴”とは，当該区間の曲率の大小や符号，区間内での曲率の平均値・和などを意味する．一例として，図2(a)のような身振り動作を行った場合に，



(a) An example of gesture motion(image sequence)



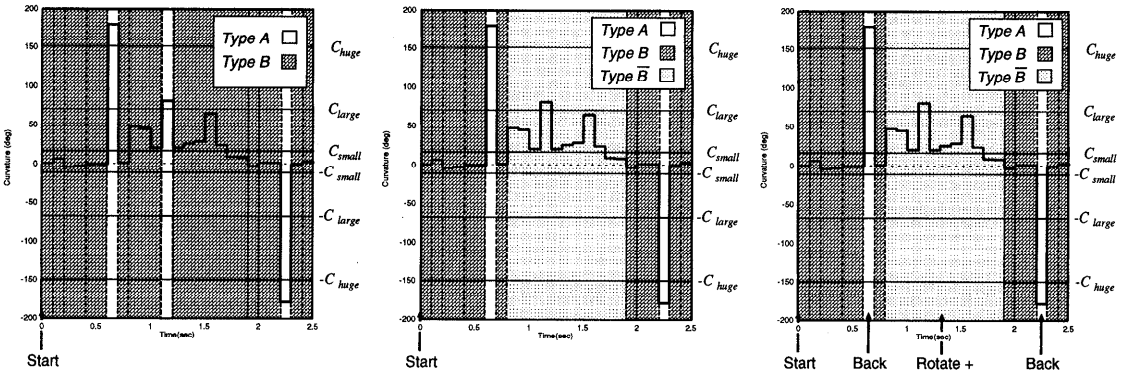
(b) Transition of "curvature" of the above gesture

図2 身振り動作とその“曲率”の時系列の一例
Fig. 2 An example of gesture motion and its curvature transition.

本アルゴリズムによりその曲率の時系列データ（同図(b)）が分割，記号化されていく様子を**図3**(a)～(c)に示した．以下で，身振り動作の分割要素となる基本動作および提案アルゴリズムの詳細について順番に説明する．

3.1 8パターンの基本動作

身振り動作の分割要素となる基本的な動作パターンは**図4**に示すように次の8通りとする：①Start（動作の立ち上げ），②Back（折り返し動作），③Turn +（正の方向に曲げる動作），④Turn −（負の方向に曲げる動作），⑤Rotate +（正の方向に回転させる動作），⑥Rotate −（負の方向に回転させる動作），⑦Arch +（正の方向に弧を描く動作），⑧Arch −（負の方向に弧を描く動作）．ただし，動作者から見て時計回りの方向を正の方向とした．また，身振り動作の開始点を表すStartは，動作開始時の身振りの方向（手を最初に上下左右のいずれの方向に動かしたか）に関する情報（属性）を持っているものとする．具体的には，①Left（最初に手を動作者から見て左に動かした場合），②Right（右に動かした場合），③Up（上に動かした場合），④Down（下に動かした場合）の4つのいずれかがStartの属性となる．



(a) Segmentation process (b) Integration process (c) Classification process

図3 身振り動作の“曲率”に基づく分割・記述プロセス

Fig. 3 “Curvature”-based segmentation and description of gesture motion.

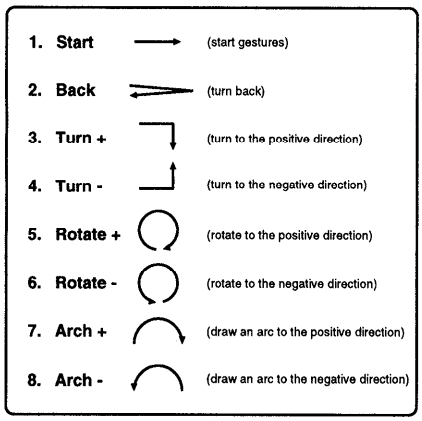


図4 8パターンの基本動作

Fig. 4 Basic motions (eight patterns).

3.2 身振り動作の記号化アルゴリズム

本稿で提案する身振り動作の“曲率”に基づく分割・記号化アルゴリズムを次に示す．

(1) **初期設定**

曲率に関する閾値 C_{small}, C_{large}, C_{huge} $(0 < C_{small} < C_{large} < C_{huge})$ ならびに曲率の和に関する閾値 T_{sum} を設定する．動作の開始点を Start と記述し，さらに身振り動作開始時の動作方向 $\boldsymbol{v}^{(0)}$ により，その属性（{Left/Right/Up/Down} のいずれか）を定める☆（**図5**参照）．

(2) **分割プロセス**（図3(a)）

曲率の時系列 $\mathcal{C} = \{c^{(t)}\}$ を「曲率の符号が同一で曲率の絶対値が C_{large} 以上の区間（以下，**Type A** の区間と呼ぶ）」と「曲率の符号が同一でその絶対値が C_{large} 未満の区間（以下，**Type B** の区間と呼ぶ）」に分割する．

(3) **併合プロセス**（図3(b)）

Type A の区間の両側に **Type B** の区間が隣接している場合，

(a) これら3つの区間の曲率の符号が同一で，

(b) 隣接する2つの **Type B** の区間内の曲率の平均値の絶対値がともに C_{small} 以上で，かつ，

(c) この両区間に挟まれた **Type A** の区間の曲率の絶対値が C_{huge} 未満であれば，

これらの3つの区間を1つの区間に併合する．こうして得られた区間を新たに **Type $\overline{\mathbf{B}}$** の区間とする．

(4) **分類プロセス**（図3(c)）

各区間を8種類の基本動作に分類する．

(a) **Type A** を次の3つに分類する．

その区間内の曲率の極値を C_m としたとき，

- $|C_m| \geq C_{huge}$ ならば，Back と定める．
- $|C_m| < C_{huge}$ ならば，

  $C_m > 0$ のとき Turn +

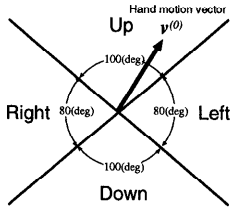


図5 “Start”の属性（動作開始時の身振りの方向）

Fig. 5 Attribute of “Start”(the direction of gesture motion at an initial time).

☆ 動作者を基準に身振りの方向を決めているため，画像上での手の動きベクトルの方向を示している図5では，Left が右側に，Right が左側になることに留意されたい．たとえば，図2(a)の (A) → (B) の場合，動作者は右手を自分から見て“左から右へ”動かしているが，画像上，つまりカメラから見た場合には，手は“右から左へ”移動する．なお，図5では，上下方向の検出領域が左右方向に比べてやや広めに設定されているが，これは，一般に，手を振る動作は，上下方向の方が左右に比べて動かしにくくぶれやすい[21]ことを考慮したためである．

表1 身振り動作分割のための閾値［単位：deg］
Table 1 Thresholds for segmentation of gesture motion.

	value	what to discriminate
C_{small}	15°	to be symbolized or not
C_{large}	70°	{Back, Turn±} vs {Rotate±, Arch±}
C_{huge}	150°	Back vs Turn ±
T_{sum}	270°	Rotate ± vs Arch ±

[NB] C_{small}, C_{large}, and C_{huge} are thresholds of curvature, while T_{sum} is a threshold of the summation of curvature in the interval.

$C_m < 0$ のとき Turn − と定める．

(b) **Type B, Type $\overline{\text{B}}$**を次の5つに分類する．その区間内での曲率の平均値の絶対値が C_{small} 未満ならば，変化のない直線的な動作と見なし，8パターンの動作にあてはめない．それ以外の場合，その区間での曲率の合計を C_{sum} としたとき，

- $|C_{sum}| \geq T_{sum}$ ならば，
  $C_{sum} > 0$ のとき Rotate +
  $C_{sum} < 0$ のとき Rotate − と定める．
- $|C_{sum}| < T_{sum}$ ならば，
  $C_{sum} > 0$ のとき Arch +
  $C_{sum} < 0$ のとき Arch − と定める．

このアルゴリズムにより，図2の身振り動作は，基本動作の系列：{Start(Right), Back, Rotate +, Back}として記述できる（図3(c)参照）．なお，上記(1)で導入した4つの閾値は，各閾値の主な役割とそれぞれの基本動作が行われる場合の典型的な曲率値を考慮し，ここでは経験的に**表1**のように設定した．

4. フロー検出パラメータの決定

ここで，2章で導入した11個の“フロー検出パラメータ”：$\{I, J, \Delta_x, \Delta_y, V_{max}, W_{max}, Q, T, V_{large}, M, N\}$ の決定手順を順番に説明する．

4.1 格子点の数と格子点間隔

はじめに，格子点の数（水平方向 I，垂直方向 J）と格子点間隔（水平方向 Δ_x，垂直方向 Δ_y）を設定する．これらのパラメータが満たすべき条件は動作検出領域の大きさおよび手の画像上での大きさに関する拘束条件から導出される．

4.1.1 動作検出領域の大きさから導かれる条件

いま身振りを行う手とカメラ間の距離を D(mm)とおく．身振り動作者は，距離 D の位置にある画像平面と平行な横 X(mm)，縦 Y(mm) の矩形領域の中で身振りを行うと仮定する．ここでは，この矩形領域の投影が動作検出領域にほぼ一致するように格子点の数と格子点間隔を調整することを考える☆．動作検出領域は，その定義より，横 $(I \cdot \Delta_x)$ (pixel)，縦 $(J \cdot \Delta_y)$ (pixel) の大きさの矩形となるので，ただちに，(I, J) と (Δ_x, Δ_y) が満たすべき次の関係を得る：

$$I \cdot \Delta_x \approx F_x \frac{X}{D}, \quad J \cdot \Delta_y \approx F_y \frac{Y}{D} \tag{4}$$

ここに，F_x, F_y はそれぞれ水平方向，垂直方向の等価焦点距離（単位：pixel）である．我々の用いたシステムでは $F_x = 912$, $F_y = 864$ である．

4.1.2 手の画像上での大きさから導かれる条件

次に手の画像上での大きさについて考察するために，その基準の姿勢として，“親指を上にして手の全指先をカメラの正面に向ける”場合を考える．このときの手の幅および高さをそれぞれ H_w(mm)，H_h(mm) とおく．いま，この手を取り囲む矩形領域 $(H_w \times H_h)$(mm) の画像上への投影がある程度の数（$K \times L$ 個）以上の格子点を占めるように格子点間隔を制御することを考える．手とカメラ間の距離を D(mm) とするとき，ただちに，(Δ_x, Δ_y) が満たすべき次の関係☆☆を得る．

$$K \cdot \Delta_x \leq F_x \frac{H_w}{D}, \quad L \cdot \Delta_y \leq F_y \frac{H_h}{D} \tag{5}$$

ただし，K, L はある正定数とする．

4.1.3 格子点数と格子点間隔の決定

次に，式(4)と式(5)を同時に満たす格子点数 (I, J) と格子点間隔 (Δ_x, Δ_y) の組を選択することを考える．数学的には，この組合せは無数に存在するので，ここでは次の方策をとる：まず，式(4)を満たす任意の (Δ_x, Δ_y) について式(5)がつねに満たされることを保証する (I, J) の条件を求め，その中で I, J とも最小となるものを選択する．こうして定まった格子点数 (I, J) と式(4)に基づいて，格子点間隔 (Δ_x, Δ_y) を手とカメラ間の距離 D の関数として定める．最小の I, J を選ぶ理由は，オプティカルフローの計算コストをできる限り小さくするためである．また (Δ_x, Δ_y)

☆ この矩形領域の投影に比べて十分大きな動作検出領域を設定する方法も考えられるが，この場合，① 背景のテクスチャに起因するノイズの影響を受ける可能性がある，② 背景に他の動物体が含まれている場合には，動作検出領域を大きくとりすぎるとアルゴリズム上問題が生じる（2章の仮定3を満足しなくなるため），③ 多くの格子点が必要となりオプティカルフローの計算コストが増大する，などの問題が生じるため，ここでは本文中に示した方針をとった．

☆☆ 実際には，カメラに対する手の向きは身振りに応じて様々に変化するが，多くの場合，手のひらあるいは甲の一部分が見える/肘の一部のフローも同時に検出されるなどの理由により，実質的には，幅・高さともここで設定した H_w, H_h より大きくなる．このような場合，式(5)（格子点間隔の設定に関してはより厳しい条件）が満たされるならば，とくに問題は生じないことに留意されたい．

のみを距離 D の関数とする理由は，距離 D によらずフローの計算量を一定にするためである．

この方針に従えば，まず，式(4)，(5)より，ただちに，格子点数 (I,J) が満たすべき次の条件を得る：

$$I > C \cdot K \cdot \frac{X}{H_w} \quad \text{かつ} \quad J > C \cdot L \cdot \frac{Y}{H_h} \qquad (6)$$

ただし，C は1より大きな正数であり，ここでは $C = 1.1$ とした．いま，実測に基づき，身振りを行う矩形領域ならびに手の幅および高さをそれぞれ $X = 350\,(\mathrm{mm})$，$Y = 300\,(\mathrm{mm})$，$H_w = 50\,(\mathrm{mm})$，$H_h = 100\,(\mathrm{mm})$ と設定する．定数 K, L については，手の幅と高さの比率も考慮し，ここでは，$K = 2.5$，$L = 5$ とする．このとき，式(6)より，$I > 19.3$，$J > 16.5$ となるので，ここでは，

$$I = 20, \quad J = 17 \qquad (7)$$

と定める．次に，式(4)に実際のパラメータおよび式(7)を代入して (Δ_x, Δ_y) について整理すると，$\Delta_x \approx 16000/D$ および $\Delta_y \approx 15200/D$ となるので，ここでは，簡単のため，両者の平均をとって，

$$\Delta_x = \Delta_y = \left[\frac{15600}{D} + 0.5\right] \qquad (8)$$

と定める．ここに $[\,]$ はガウスの記号である．

4.2 フロー探索範囲，画素の間引き率，画像サンプリング周期，大きな身振り検出用閾値

まず，これらのパラメータの中で，使用する専用ハードウェア[18]の制約からフロー探索範囲の半径（水平方向 V_{max}，垂直方向 W_{max}）(pixel) を身振りを行う手とカメラ間の距離によらない固定値として，

$$V_{max} = W_{max} = 8 \qquad (9)$$

に設定した．以下では，単位サンプリングあたりの身振り動作者の手の見掛け上の移動量がこの探索エリア内につねに収まるように画素の間引き率と画像サンプリング周期を制御することを考える．

いま一連の身振り動作が平均速度 $H_v\,(\mathrm{mm/frame})$ で行われたと仮定する．ただし，$1\,(\mathrm{frame}) = 1/30\,(\mathrm{sec})$ である．サンプリング周期 $T(\mathrm{frame})$ で身振り動画像を取り込むものとし，さらに，サンプリングされた各画像において，各格子点を基準に間引き率 Q で画素の間引き処理が行われるものとする．身振りを行う手とカメラ間の距離が $D\,(\mathrm{mm})$ のとき，単位サンプリングあたりの間引き処理画像上での手の“平均的な”移動量 $v\,(\mathrm{pixel})$ は，3つのパラメータ D，Q，T の関数として次式により近似的に与えられる．

$$v \approx \frac{T}{1+Q}\left(F' \cdot \frac{H_v}{D}\right)$$

ただし，$F' = (F_x + F_y)/2$ である．ここで，ある正定数を“基準移動量” $V_{std}\,(\mathrm{pixel})$ として設定し，次の問題を考える．

問題1 与えられた D に対して次式が成立するように非負の整数 Q および 正の整数 T を制御する．

$$\frac{T}{1+Q}\left(F' \cdot \frac{H_v}{D}\right) \approx V_{std}$$

この問題は，“基準距離” $D_{std} = F' \cdot H_v / V_{std}$ を導入することにより，次のように書き換えることができる．

問題2 与えられた D に対して次式が成立するように非負の整数 Q および 正の整数 T を制御する．

$$\frac{T}{1+Q} \cdot \frac{D_{std}}{D} \approx 1$$

ここに，D_{std} は，$Q = 0, T = 1$ のときに $v = V_{std}$ となる距離に相当する．基本的には，問題2を解けばよいが，数学的には同一の解を与える (Q,T) の組合せが無数に存在するので，ここでは次のような方策をとった．

(1) $D \leq D_{std}$ ならば，画像サンプリング周期 $T = 1$ と固定し，画素の間引き率 Q を次式で与える．

$$Q = \left[\frac{D_{std}}{D} - 0.5\right] \qquad (10)$$

(2) $D > D_{std}$ ならば，画素の間引き率 $Q = 0$ と固定し，画像サンプリング周期 T を次式で与える．

$$T = \left[\frac{D}{D_{std}} + 0.5\right] \qquad (11)$$

ただし，$[\,]$ はガウスの記号である．

図6は，上記手法に基づいてパラメータ Q と T を制御することにより得られる“手のフローの平均的な大きさ” v をある定義域 $\mathcal{D} = \{D | 600 \leq D \leq 16000\}$ の範囲内で図示したものである．ただし便宜上，横軸 D は2を底とする対数目盛で表示されている．図中の太実線が本手法により調節されるフローの平均的な大きさの推移を示している．間引き率 Q とサンプリング周期 T が距離 D に応じて適切に切替えられ，フローの大きさ v がつねに V_{std} 近傍に保持されている様子が分かる．なお，この v を求めるための手の平均速度 H_v については，複数の被験者に実際にいくつかの身振りを行ってもらった結果の平均的な値を採用し，ここでは，$H_v = 15\,(\mathrm{mm/frame})$ とした．また，基準移動量 V_{std} は，フロー探索範囲の定義（2章参照）と式(9)の結果より，許容できる最大の移動量の半分程度の値が適当と考え，ここでは，$V_{std} = 3.5\,(\mathrm{pixel})$ とした．これらより，ここでの基準距離は，$D_{std} = 3800\,(\mathrm{mm})$ となる．また，同図にも示されているように，動作者の無意識的な小さな動き

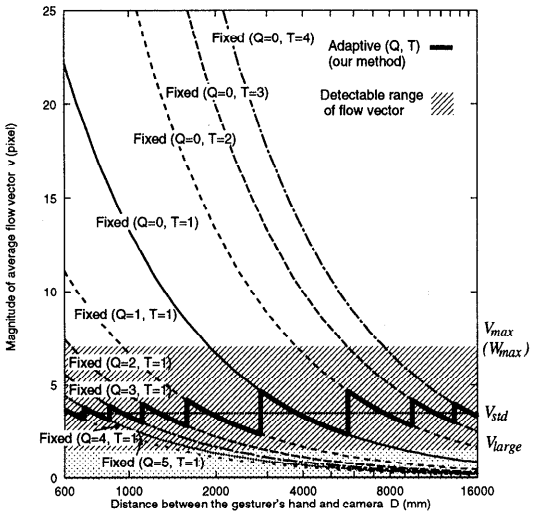


図 6 画素の間引き率と画像サンプリング周期の制御によるフローの平均的な大きさの推移

Fig. 6 Transition of the magnitude of average flow vector v against the distance D by controlling the rates Q and T.

に起因するフローを除去するための閾値 V_{large} (pixel) は，手の動きによるフローを $V_{std} = 3.5$ (pixel) 近傍におさえていることも考慮し，

$$V_{large} = 2 \tag{12}$$

と設定した．

4.3 相関演算ブロックサイズ

最後に，相関演算ブロックサイズ（幅 M，高さ N）（単位：pixel）を決定する．基本的には，手の投影幅あるいは高さの標準的な値に基づいて適当に選択すればよい．ただし，ここでは，使用するハードウェアの制約[18] から M, N とも 8 の倍数とする．

いま，基準距離 $D_{std} = 3800$ (mm) における "基準姿勢"（4.1.2 項参照）での手の投影幅を求めると，$F_x \cdot H_w / D_{std} \approx 12$ (pixel) となる．一般には，この大きさはカメラまでの距離の逆数 $(1/D)$ に比例するが，すでに 4.2 節で示したように，手–カメラ間距離が基準距離 D_{std} 以下の場合については，ほぼ同じ比率で画素の間引き処理も行われるため，結果として，相関演算ブロックに占めるであろう手領域の面積はほとんど変わらない．そこで，簡単のため，M, N は D によらず一定とし，上記の値をもとに，ここでは，

$$M = N = 16 \tag{13}$$

と設定する．

4.4 ま と め

以上，本章で述べた手順をまとめたものを図 **7** に示す．本手法の最大の特徴は，オプティカルフローを検出するための計 11 個のパラメータのうち，格子点間隔

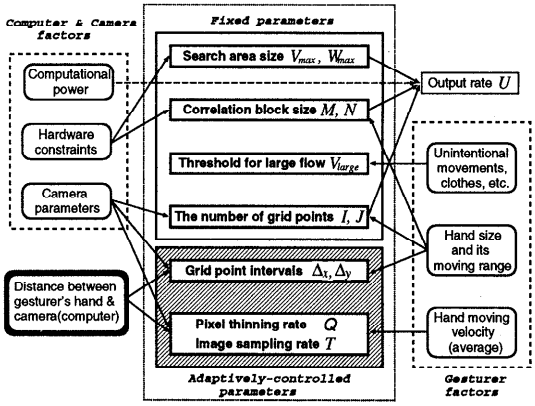


図 7 フロー検出パラメータとその決定要因

Fig. 7 Flow detection parameters and their decision factors.

(Δ_x, Δ_y)，画素の間引き率 Q および画像サンプリング周期 T の 4 つを単なる固定値ではなく "身振りを行う手とカメラ間の距離" D の関数として定めている点にある．オプティカルフローを利用する従来法[14),15)] では，ある一定の "身振り動作者–カメラ間距離" を想定し，すべてのフロー検出パラメータは，その距離関係においてうまく身振り動作が検出される値に固定されていた．本手法の導入によって，身振り動作者とカメラ間の距離が大きく変わっても柔軟に対応可能な身振り認識システムが構築できるものと考えられる．

5. 身振り認識実験

3 章で示した身振り動作の記号化アルゴリズムと 4 章で示したフロー検出パラメータの決定方法の有効性を検証するために，身振り認識実験を行った．以下で，実験システム，認識対象，実験方法について順番に説明し，最後に認識実験の結果を述べる．

5.1 実験システム

本実験では，1 台の CCD カメラを用いて身振り動画像（濃淡画像列）を撮影する．画像のサイズは 1 枚あたり 640×480 (pixel)，各点 8 ビットである．レンズの焦点距離は 8 mm（水平方向等価焦点距離 912 画素，垂直方向 864 画素）のものを用いた．これらの画像入力およびその局所相関演算に基づくオプティカルフローを検出するための専用の画像処理装置として，富士通社製のカラートラッキングビジョン PCI 版（TRV-CPW5）[18] を利用した．これは，局所相関を行うための演算ブロックのサイズや格子点の数およびその間隔，画素の間引き，画像の時間的なサンプリング周期などのパラメータ変更が容易に制御できる PCI バス対応の画像処理ボードであり，提案手法の実装に適している．

文献18)にもあるように，$M \times N = 16 \times 16$ (pixel) のモノクロテンプレートの場合，水平垂直 $-8 \sim +7$ 画素を探索範囲（すなわち $V_{max} = W_{max} = 8$）とする局所相関演算を1フレームあたり約125回実行できる．一方，式(7)に示したように，ここでは，オプティカルフローの計算に $I \times J = 340$ 回の相関演算を必要とする．その結果，この演算に曲率を求めるための処理を加えた本実験における曲率のサンプリング周期は $U = 3$（frame）となった（図7参照）．

前章までに述べたように，曲率の計算プロセス（2章）は身振り動作開始と同時にスタートし，一連の動作終了後に身振りの記号化プロセス（3章）が実行されなければならない．ここでは，これらのプロセスの実行とトラッキングビジョンの制御を行うホストの計算機に1台のPC/AT互換機（Intel MMX Pentium, 166 MHz）を用い，Linuxをその OS とする PC ベースのシンプルな構成[22]により，身振り動画像の入力から記号列の出力までの一連の処理を自動で行う“実験システム”を実現した．いまのところ，本システムでは，$V_{large} = 2$ (pixel) 以上の大きさを持つフローベクトルが，ある一定数（ここでは10個）以上検出されたとき身振り動作が開始されたと見なし，それらがある一定時間（ここでは10フレーム）以上連続して検出されなくなったときその身振りが終了したと判定している．その後に起動される本システムによる身振り動作の記号化アルゴリズムの実行時間は約0.1 (sec) である．

5.2 認識対象

本稿では，認識する身振りとして以下にあげる4つ（**#1**〜**#4**）を選択した（**図8**参照）．

#1： 手を左右に2回振る動作（図8(a)）
　モデル：{Start(Left), Back, Back, Back}

#2： 「オーム型」の身振り動作（図8(b)）
　モデル：{Start(Right), Back, Rotate +, Back}

#3： ワイパーのように手を振る動作（図8(c)）
　モデル：{Start(Up), Arch +, Back, Arch −}

#4： 四角形を描く動作（図8(d)）
　モデル：{Start(Up), Turn +, Turn +, Turn +}

これらのモデルはいずれもあらかじめ用意した8パターンの基本動作の組合せとして簡単に与えられ，複数の動作者による認識システムへの身振りの例示プロセス（モデルの学習プロセス）を経ないことに留意されたい．たとえば，**#2**の身振り動作であれば，「まず手を動作者から見て右方向に動かし（Start(Right)），いったん反対に折り返したのち（Back），動作者から見て時計回りに回転させて（Rotate +），再度折り返す（Back)」動作として，記号列：{Start(Right), Back, Rotate +, Back} のみが計算機にたくわえられる．従来の多くの身振り認識システムでは，身振りの標準パターン（モデル）を作成するのに手間と時間がかかりすぎるという欠点があったが，本システムでは，このように，身振りのモデルを直観的かつ簡単に，短時間で大量に与えることができるという特長がある．参考までに，**#1**，**#3**，**#4**の身振り動作を行った一例（ある動作者による身振り動画像系列の抜粋）を**図9**に示す．なお，**#2**の身振りについてはすでに示した図2(a)を参照されたい．

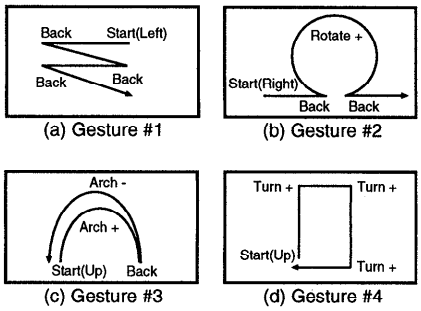


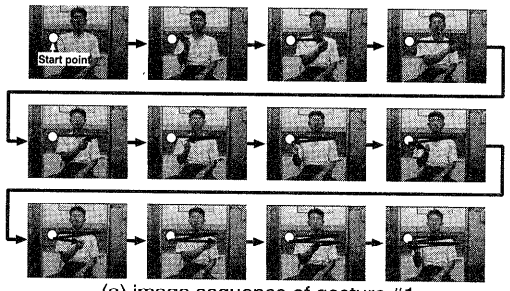
図8 認識実験に用いる身振り
Fig. 8 Target gestures.

5.3 実験方法

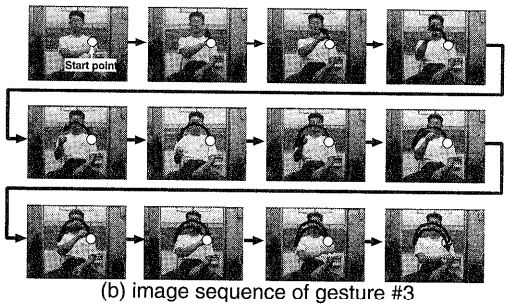
認識実験の基本手順は次のとおりである：いくつかの異なる距離において撮影された複数の被験者による身振り動画像系列を入力として，「4章の方法によるパラメータ設定のもとで2章の手続きを通して得られる身振りの“曲率”の時系列データを3章で述べたアルゴリズムに基づき基本動作の系列に変換する」．得られる出力結果（記号列）が対応する身振りのモデルと“完全に一致”した場合のみ“正しく認識できた”と見なして，対象となる身振りの認識率（＝正しく認識できた数/試行回数）を被験者ごとおよび手–カメラ間距離ごとに算出する．

上記手順を実行するため，ここでは次のような実験条件を設定した．

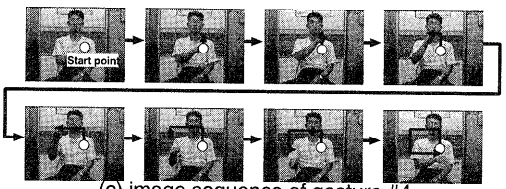
- 身振りを行う手とカメラ間の距離 D は，$D =$ 1000, 2000, 4000, 8000 (mm) の4種類とする．式(7)〜(13)より，それぞれの距離 D におけるフロー検出パラメータを**表2**のように設定する．
- 身振り動作者（被験者）として，本システムに比較的精通している3人（**Subject A, B, C**）とシステムに関する知識を持たない3人（**Subject D, E, F**）の計6人を選ぶ．**D**〜**F**の3人に対しては，実験の直前に，認識対象となる身振り動作の実演と口頭による説明をそれぞれ1度だけ行う



(a) image sequence of gesture #1



(b) image sequence of gesture #3



(c) image sequence of gesture #4

図 9 認識対象となる身振り動作を行った例（身振り#2については図 2 (a) を参照）

Fig. 9 Examples of target gesture motion (For the gesture #2, see Fig. 2 (a)).

ものとする．

- 同一の条件で 1 つの身振りを 30 回行ってもらう．すなわち，本実験の入力データの総数は 2880 個（距離 4 種類 × 被験者 6 人 × 30 回 × 身振り 4 種類）である．

なお，本実験を円滑に進めるにあたり，便宜上，次の 3 つの制約を設けた．① 画像平面と床面が垂直になるようにカメラを設置する，② 被験者は椅子に座って身振り動作を行う，③ 動作検出領域内に被験者の頭部を除く上半身がほぼ収まるように，被験者の座る椅子の位置および高さ（あるいはカメラの設置高さ）をあらかじめ調整しておく．これらの制約は，実際の応用上は大きな問題となるが，これについては 6 章で議論する．

図 10 は，手–カメラ間距離 $D = 1000\,(\mathrm{mm})$ における 6 人の身振り動作者のスナップショットである．図に示すように，本実験では身振りを行う際の服装についてはとくに指示していない．また，被験者 **A** の距離 $D = 1000, 2000, 4000, 8000\,(\mathrm{mm})$ における身振りのスナップショットを**図 11** に示す．図 10，図 11 はすべて実際の CCD カメラからの画像であり，図中に白実線で描かれている矩形領域がそれぞれの動作検出領域を表している．特に図 11 において，動作検出領域の大きさ（$I \cdot \Delta_x \times J \cdot \Delta_y$）が距離 D により大きく異なっていることに留意されたい（I, J, Δ_x, Δ_y の実際の値については表 2 参照）．

表 2 実験に用いたフロー検出パラメータ

Table 2 Flow detection parameters for experiments.

name	D				*cf.*
	1000	2000	4000	8000	
I	20				Eq. (7)
J	17				
Δ_x	16	8	4	2	Eq. (8)
Δ_y					
V_{max}	8				Eq. (9)
W_{max}					
Q	3	1	0		Eq. (10)
T	1		1	2	Eq. (11)
V_{large}	2				Eq. (12)
M	16				Eq. (13)
N					

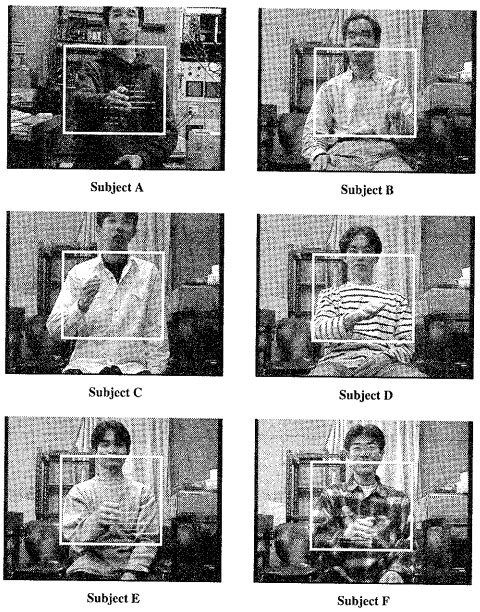


図 10 手–カメラ間距離 $D = 1000\,(\mathrm{mm})$ における 6 人の身振り動作者のスナップショット

Fig. 10 Snapshots of six gesturers at $D = 1000\,(\mathrm{mm})$.

5.4 結果と議論

それぞれの身振り（**#1**～**#4**）の認識率を被験者ごとおよび手–カメラ間距離ごとにまとめたものを**表 3**,

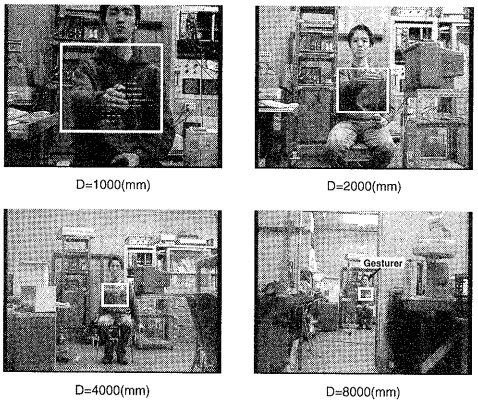


図11 手–カメラ間距離 D が 1000, 2000, 4000, 8000 (mm) の場合のある身振り動作者のスナップショット
Fig. 11 Snapshots of a gesturer at D =1000, 2000, 4000, and 8000 (mm).

表4, **表5**, および**表6**に示す. これらの表より, 手–カメラ間距離による差異, 身振りの種類による差異, および被験者ごとの差異に関してそれぞれ以下のことが分かる.

結果1 手–カメラ間距離 $D = 1000$ (mm) の場合には, いずれの身振りについても被験者平均で90%以上の高い認識率が得られている. 距離 D が大きくなるに従い, 各身振りとも多少の認識率の低下 ($D = 1000$ のときに比べ最大で5%程度減) が見られるが, どの距離においても**4種類すべての身振りの認識率が被験者平均で85%以上に保持されている.**

結果2 すべての被験者, すべての距離において, 身振り "**#1** および **#2**" の方が "**#3** および **#4**" に比べて認識率が高くなる傾向にある. 全体の平均値を見ても, 前者が 95~96%, 後者が 87~88%の認識率となっており, 8%程度の明らかな差異がある. 一方, **#1** と **#2** の間の認識率の差および **#3** と **#4** の間の認識率の差はどちらも比較的小さい (平均で1%程度).

結果3 被験者ごとに認識率 (4つの距離における値の平均) を見た場合, 身振り **#1** において認識率の高い人と低い人の差が最も小さくなっている. 続いて, 身振り **#2**, **#4** の順にこの差が大きくなり, 身振り **#3** の認識率の開きが最も大きい. 一方, システムに精通している者 (**A**~**C**) の認識率とそうでない者 (**D**~**F**) の認識率を比べると, 表3~表6より, 前者が平均で 90.5%, 後者が平均で 93.4%と算出され, むしろシステムに関する知識のない者の方が全般的に認識率が高くなった.

以下にそれぞれの結果に対する考察を行う.

議論1 結果1より, 4章で示したフロー検出パラメータの設定手法が有効に作用していることが分かる. 距離の増大にともない認識率がやや低下する主要因としては,「身振り動作者はカメラの中心に向かって身振りを行おうとするが, 実際には, カメラまでの距離が遠くなるに従って, カメラ中心に対する身振り動作 (手) の位置および方向のずれがどうしても大きくなってしまい, その結果, 手が動作検出領域の外にはみ出してしまう可能性が大きくなる」ことがあげられる. この問題については, 現在は固定としている動作検出

表3 身振り#1の認識率
Table 3 Recognition rates of gesture #1.

subject	distance D				total
	1000	2000	4000	8000	(%)
A	29/30	29/30	27/30	26/30	92.5
B	30/30	30/30	27/30	29/30	96.7
C	30/30	29/30	28/30	29/30	96.7
D	30/30	30/30	30/30	27/30	97.5
E	30/30	30/30	30/30	28/30	98.3
F	30/30	29/30	28/30	30/30	97.5
total (%)	99.4	98.3	94.4	93.9	96.5

表4 身振り#2の認識率
Table 4 Recognition rates of gesture #2.

subject	distance D				total
	1000	2000	4000	8000	(%)
A	29/30	27/30	26/30	28/30	91.7
B	29/30	27/30	30/30	28/30	95.0
C	28/30	30/30	29/30	28/30	95.8
D	29/30	30/30	25/30	26/30	91.7
E	30/30	30/30	30/30	30/30	100
F	30/30	29/30	28/30	29/30	96.7
total (%)	97.2	96.1	93.3	93.9	95.1

表5 身振り#3の認識率
Table 5 Recognition rates of gesture #3.

subject	distance D				total
	1000	2000	4000	8000	(%)
A	27/30	28/30	24/30	26/30	87.5
B	29/30	29/30	26/30	25/30	90.8
C	24/30	25/30	24/30	25/30	81.7
D	23/30	25/30	26/30	24/30	81.7
E	29/30	29/30	28/30	26/30	93.3
F	30/30	27/30	29/30	28/30	95.0
total (%)	90.0	90.6	87.2	85.6	88.3

表6 身振り#4の認識率
Table 6 Recognition rates of gesture #4.

subject	distance D				total
	1000	2000	4000	8000	(%)
A	27/30	24/30	24/30	25/30	83.3
B	30/30	27/30	25/30	29/30	92.5
C	25/30	25/30	25/30	23/30	81.7
D	26/30	24/30	26/30	25/30	84.2
E	29/30	28/30	29/30	26/30	93.3
F	30/30	27/30	24/30	29/30	91.7
total (%)	92.8	86.1	85.0	87.2	87.8

領域の画像上での位置を手の動きにあわせて可変にする（検出されるフローベクトル群の重心に一致するように，動作検出領域の重心位置を制御する）ことで十分に解決可能と思われる．

議論2 次に結果2（つまり3章で示した手法の性能）について考察するために，それぞれの身振りの認識率とその時系列データをもう少し詳しく分析する．

身振り**#1**は，「直線的な動作とその折り返し動作」（つまり“曲率”がほとんど変化しない動きとそれが急激に変化する動きの2種類）からなるきわめてシンプルな動作であり，提案手法により高い認識率が達成可能なことは容易に想像できる．実際，表3より，すべての被験者，すべての距離において，92.5%以上の安定した高い認識率が得られており，このようなシンプルな身振り動作に対する提案手法の有効性が明らかになった．

身振り**#2**は，これにさらに「円を描く（手を滑らかに曲げる）動作」が加わるが，その認識率は**#1**のそれとほとんど変わらず高い（表4参照）．一般に，人は手を滑らかに動かす動作を行っているつもりでも無意識のうちに大きな角度で手を曲げてしまうことがある，あるいは，手を早く動かしてしまった場合に曲率のサンプリング周期との兼ね合いで区分的に大きな曲率を生じてしまうことがあるなどの理由により，曲率値の局所的な判定では手を滑らかに曲げたのか急に曲げた動作なのかを区別することが難しい．それにもかかわらず，このような高い認識率を保持しているのは，提案手法における「併合プロセス」（曲率の変化を大局的にチェックするプロセス）に頼る点が大きいと思われる．このことを確認するために，実際の曲率の時系列データを解析したところ，“正しく認識された”身振り**#2**のデータの30.1%が併合プロセスを経なければ正しく記号化されないことが明らかになった．

身振り**#3**は，基本的には，身振り**#2**に含まれているのと同様の「手を滑らかに曲げる（ここでは弧を描く）動作」および**#1**，**#2**の両方にも含まれている「折り返し動作」の組合せからなる．しかしながら，その認識率は，**#1**や**#2**に比べて平均で8%程度低下している（表5参照）．この原因を探るため，実際の出力結果を分析したところ，正しく認識されなかった身振りの28.6%にあたる24個のデータが，本来は，{Start(Up), Arch +, Back, Arch −}と記号化されるべきところを{Start(Up), Arch +, Turn +, Arch −}または{Start(Up), Arch +, Turn −, Arch −}と記号化されていることが明らかになった．これは，**#3**に含まれる折り返し動作が“曲線から曲線への”折り返しであるために生じる；動作者は直線から直線に折り返すのと違って，曲線から曲線に折り返すときは元の方向にまっすぐ折り返す（基本動作ではBack）ことが難しく，無意識のうちにTurnと判定される範囲の曲率を持った動きをすることがあると考えられる．この問題を解決するためには，たとえばBackとTurnを判別するためのもう1つ別の閾値C'_{huge}（ただし$C_{large} < C'_{huge} < C_{huge}$．表1参照）を設け，曲率値が$C'_{huge}$以上$C_{huge}$未満の場合には，Back，Turnの両方の可能性を残す方法などが考えられる[21]．

身振り**#4**は，ここまでの身振りと異なり，単純な「直線から直線への曲げ動作」の繰返しのみからなる．それにもかかわらず，その認識率は，**#3**と同様，身振り**#1**，**#2**に比べて平均で約8%低い（表6参照）．この原因を探るため，実際の身振りの出力結果を解析したところ，本来は，{Start(Up), Turn +, Turn +, Turn +}と記号化されるべきところが，{Start(Up), Back, Turn −, Turn +, Turn +}と記号化されることが多いことが明らかになった．一般に，動作者は直線から直線へ曲げるところ（基本動作ではTurn）で，手を急に静止させようとするために，その反動で手が少しもとの方向に戻る（これは折り返し動作Backに相当する）傾向にあると考えられる．この部分の動作は一瞬で行われるために被験者自身も気づかないようである．具体的には，この例では，被験者は1つ目のTurn +の動作をしようとしていったん手を静止させるが，そのとき微妙に少し手が下がってから横に移動した（この一連の動きは曲率の1サンプリング周期内で行われた）ために，このような出力結果となったものと考えられる．2つ目以降のTurn +の動作についても同様の現象が生じることが考えられ，実際，正しく認識されなかったデータの20.5%（18個）がこの理由によるものであった．この問題の解決法としては，単純に{Back, Turn +}は{Turn −}に変換し，{Back, Turn −}は{Turn +}に変換するという一種の補正プロセスを導入する（ただしBackからTurnへの動作の変化が曲率のサンプリング周期内で行われた場合に限る）やり方が考えられる[21]．

議論3 結果3は，システムに関する知識が必ずしも認識率の向上に寄与しない（システムに関する知識がなくても高い認識率が得られる）ことを意味している．換言すれば，この結果は，提案手法が不特定多数の動作者の身振り認識に有効であることを示している．なお，身振りの種類によって異なる被験者間での認識率の差に多少の違いが生じるのは，その身振りの“しやすさ”に起因する点が大きいと思われる．たとえば，

手を左右に振る動作（身振り**#1**）は，我々が日常においてもよく行う身振りであり，だれがやってもそれほど差異は生じない．一方，身振り**#3**や**#4**は，普段はあまり行うことのない不慣れな身振りであり，このことが異なる被験者間での認識率の差異を増大させているものと思われる．

なお，本実験では，入力データ総数2880個のうちその8.1%にあたる232個のデータが用意したどの身振りのモデルとも一致しなかった（他の身振り動作と"誤認"した数は「ゼロ」であることに留意されたい）が，この主な原因としては，議論2で述べた認識対象の特性に起因する理由以外に，次の3点が考えられる．

- 身振り動作があまりにも大きすぎて手が動作検出領域からはみ出してしまった．
- 手を速く動かしすぎて移動ベクトルが正しく検出されなかった．
- 被験者にとっては意識にない余分な動きや，肩などの手以外の身体の部分の動きを検出してしまった．

まとめ 以上，本実験で得られた結果は次のようにまとめられる．

(1) 提案手法を用いれば，「直線的な動作」や「直線から直線，直線から曲線および曲線から直線への折り返し動作」ならびに「手を滑らかに曲げる（円弧を描く）動作」とそれらの組合せからなる身振り動作については，不特定の人間に対して高い認識率（実験では90%以上）を達成可能な身振り認識システムを構築できる☆．

(2) 「曲線から曲線への折り返し動作」または「直線から直線への曲げ動作」を含む身振り動作については，現状では(1)の身振り動作に比べ多少認識率が低下するものの，不特定の動作者に対して比較的高い認識率（実験では80% 以上）を達成できる．

(3) 身振りを行う手とカメラ間の距離が大きく変化した場合でも（実験では1～8m），その距離値に基づきフロー検出パラメータを調整することで，身振り認識率を同程度の高いレベルに保持することが可能である．

5.5 従来法に対する提案手法の優位性について

以上の実験に加えて，とくに従来法に対する提案アプローチの優位性を明らかにするために，本研究の新規な部分である次の2点：(1)「オプティカルフローからの"曲率"を用いること」(2)「"曲率"に基づいて身振りを分割/記述（記号化）すること」の有効性を個々に検証する実験も行ったので，以下に簡単に示す．この実験においては，身振りを行う手とカメラ間の距離は $D = 1000\,(\mathrm{mm})$ の1種類とし，フロー検出パラメータは表2に示したものをそのまま適用した．具体的な実験内容およびその手順は次のとおりである：まず，4人の被験者（**Subject G, H, I, J**）に5.2節で述べた4種類の身振り動作（**#1**～**#4**）を片手で30回ずつ行ってもらい，計480個（被験者4人×30回×身振り4種類）の身振り動画像系列をあらかじめ撮影・取得しておく．これを本実験の入力データとして利用し，次の3つのケースそれぞれに対して，5.3節と同様の計算方法で認識率を測定し，それらの結果の比較検討を行う．

ケース1 オプティカルフローから直接的に計算される身振りの"曲率"の時系列データを基本動作の系列として記号化した場合

ケース2 身振りの動作軌跡をいったん復元しそれを解析することにより得られる軌跡の曲率の時系列データを基本動作の系列として記号化した場合

ケース3 オプティカルフローから直接的に得られる身振りの"方向"の時系列データを基本動作の系列として記号化した場合

ケース1が提案手法に相当する．ケース2は曲率の計算方法を従来型にしたものであり，ケース3はオプティカルフローの利用法を従来型にしたものである．なお，ケース2における身振りの動作軌跡は，ここでは，2章(1)～(3)の手続きを通して得られるフローベクトル群の重心座標の時系列として近似的に復元した．また，ケース3においては，記号化プロセスに用いるデータの種類がケース1，ケース2と異なるため，3章のアルゴリズムをそのまま適用することができず，単純な比較実験は困難であるが，ここでは，2章(1)～(3)の手続きにより得られる身振りの"方向"の時系列（手の動きベクトル $\boldsymbol{v}^{(t)}$）を絶対的な上下左右斜めの8方向にチェーン符号化（たとえば"011222"）し，さらに同一の符号が連続して出現する場合には時間方向に圧縮する（上記例の場合，"011222" → "012"）という簡便な記号化手法により代替した．この場合，入力データと照合する身振りのモデルも異なってくるが，ここでは，被験者**G**の身振り（**#1**～ **#4**）を一通り事前に1度だけ撮影し，それらを上記手法により記号化したものをモデルとして用いた．

☆ この所見に基づいて，実際にこれらの動作を組み合わせた7種類の身振りコマンド（①前進，②後退，③右折，④左折，⑤時計回りのその場回転，⑥反時計回りのその場回転，⑦停止という7つの命令に相当）により，小型の車輪型移動ロボット（ただしカメラは環境中に固定）を誘導することにも成功している．この詳細については，文献23)を参照されたい．

表7 ケース1, 2, 3における身振り認識率の比較
Table 7 Comparison between recognition rates of target gestures in case 1, 2, and 3.

	subject	CASE 1	CASE 2	CASE 3
#1	G	30/30	13/30	30/30
	H	30/30	17/30	13/30
	I	29/30	1/30	10/30
	J	30/30	17/30	30/30
#2	G	30/30	0/30	22/30
	H	30/30	12/30	4/30
	I	30/30	6/30	16/30
	J	29/30	12/30	2/30
#3	G	28/30	2/30	25/30
	H	27/30	6/30	13/30
	I	26/30	9/30	16/30
	J	23/30	2/30	7/30
#4	G	29/30	2/30	25/30
	H	23/30	3/30	16/30
	I	27/30	2/30	25/30
	J	26/30	1/30	17/30
total (%)		93.1	21.2	56.5

実験結果（上記した3つのケースにおける身振り認識率を身振りの種類ごとおよび被験者ごとにまとめたもの）を**表7**に示す．この表より以下のことが分かる．

ケース2の認識率は全般的に低く（全体で21.2%），ケース1のそれ（同93.1%）を大きく下回っている．一般に，動作軌跡の解析に基づく従来型の曲率の計算法では，軌跡の2次微分の計算をともなうため，軌跡の復元誤差やノイズの影響を受けやすい[9]ことが知られており，これが，ケース2の低認識率の主要因であると考えられる☆．一方，ケース1の場合，同表に示すように，すべての被験者において安定した高い認識率を得ることに成功している．以上の結果より，「オプティカルフローから直接的に曲率を計算する本方式は，軌跡の復元誤差やノイズの影響を本質的に受けないという意味できわめて有効なアプローチである」といえる．

次に，ケース3の場合を見ると，モデルの例示者である被験者**G**に関しては，すべての身振りにおいて，ケース1とほぼ同レベルの高い認識率を達成している．**G**以外でも，場合によっては80%以上の高い認識率を示すことがあるが，被験者により認識率に相当のばらつきがあり，全体でも56.5%とケース1に比べてかなり低くなっている．これは，手を振る絶対的な方向の個人差に起因するものと推察される．これに対して，ケース1では，すべての被験者において安定して高い認識率を保持することに成功している．以上の結果より，「"曲率"に基づいて身振りを分割/記述する本手法の方が"方向"ベースで記述するアプローチよりも本質的に個人差の影響を受けにくい」ことが分かる．なお，この個人差の影響は複数の動作者による身振りの例示と学習により解消することが可能である．たとえば，オプティカルフローの方向ベクトルを身振りの特徴量とする文献14)の方法では，モデルの学習者を1人から6人に増やすことにより，ある楽器を演奏する身振りの認識率を50.0%から93.8%に向上させており，ケース3の認識率も同様のアプローチによりケース1と同レベルの認識率に向上させることが十分可能と思われる．逆にいえば，これは，「"方向"ベースの従来法は，高認識率達成のためには十分なモデル学習プロセスが必要不可欠であるが，提案手法では，事前の学習を経ることなく従来法と同等の認識率を達成可能である」ことを示している．

以上，本稿で扱っている問題の枠組内ではあるが，提案手法の従来法に対する優位性が実験的に示された．

6. おわりに

本稿では，オプティカルフローから検出される手の軌跡の"曲率"に基づいて，一連の身振り動作を簡単な「基本動作」の列として分割・記号化する方法を提案した．いくつかの身振り認識実験を通して，この「基本動作」による記述が不特定の人間の動作に対してロバストに獲得できることを示した．また，オプティカルフロー検出時のいくつかのパラメータを手とカメラ間の距離に基づいて適応的に調整する方法もあわせて提案し，広い距離範囲にわたって，この記述が安定して獲得できることも確認した．

今回行った実験では，便宜上，身振り動作者とカメラの配置に関していくつかの制約を設定した（5.3節参照）．これらはいずれも，環境中に固定されたカメラを用いたために必要となった制約であり，基本的には，パン・ティルト角の回転制御が可能な能動カメラ系を導入することによりすべて取り除くことが可能である．現在，「このような能動カメラ系を搭載した1台の移動ロボットが自らの視覚を通して人間の身振りによる指示を理解し，その指示内容に応じて環境中を動き回る場合」を想定した"身振りによるロボット誘導システム"の構築を進めており，人間（指示者）があらかじめ定められた位置に立っている場合には，本手法を適用することによりロボットを身振りで自由に（ただ

☆ 軌跡の曲率を利用して身振り動作を記述する文献9), 10)の方法では，この問題を回避するために，今回のケース2のように一律に算出された曲率の時系列データを用いるのではなく，曲率の計算間隔を軌跡の長さに応じて適応的に調節する[9]，軌跡中の極値点の数や曲線の凹凸を綿密に調べてノイズに影響されずに安定に求まる点をルールベース的に選択する[10]などの様々な工夫が行われていることを付記しておく．

し半径4m以内）誘導できることを確認している[24]．この例では，ロボットは内界センサからの情報をもとに（指示者が身振りを行う位置は不変として），搭載カメラと指示者との距離および方向を時々刻々推定することにより，人とロボットの相対位置関係の変動にロバストな身振り認識を実現している．ただし，指示者自身も移動する場合には，指示者の発見（画像からの人物の切り出し）や指示者までの距離の推定などまだ解決すべき問題がいくつかあり，現在，順次研究開発を行っている．

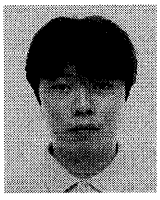
本手法は，複数の動作者による身振りの例示・解析プロセスを経ることなく，身振りのモデルを直観的かつ簡単に準備できるという利点があるが，その一方で，「基本動作」の列で表すことが困難な身振り動作を本質的に扱うことができないという問題がある．たとえば，現状では，カメラの光軸方向（奥行き方向）への変化をともなう身振りは正しく認識できない．また，片手の身振り動作を想定しているため，そのままでは両手で行う身振りにも対応できない．前者の問題に関しては，オプティカルフローの分布を考慮する，ステレオ視を導入することなどにより解決できる可能性があり，現在，検討を進めている．また，後者の問題については，画像中に両手検出用の複数の動作検出領域を混在させることにより対処する“拡張アルゴリズム”を構築中であるが，これらについては別の機会に報告したい．

参考文献

1) Pavlovic, V.I., Sharma, R. and Huang, T.S.: Visual Interpretation of Hand Gestures for Human-Computer Interaction: A Review, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.19, No.7, pp.677–695 (1997).
2) Darrell, T. and Pentland, A.: Space-Time Gestures, *Proc. 1993 IEEE Computer Soc. Conf. on Computer Vision and Pattern Recognition*, New York, USA, pp.335–340 (1993).
3) 高橋勝彦，関 進，小島 浩，岡 隆一：ジェスチャー動画像のスポッティング認識，電子情報通信学会論文誌，Vol.J-77-D-II, No.8, pp.1552–1561 (1994).
4) 大和淳司，倉掛正治，伴野 明，石井健一郎：カテゴリー別VQを用いたHMMによる動作認識法，電子情報通信学会論文誌，Vol.J-77-D-II, No.7, pp.1311–1318 (1994).
5) Starner, T.E. and Pentland, A.: Visual Recognition of American Sign Language Using Hidden Markov Model, *Proc. 1st IEEE Int. Workshop on Automatic Face and Gesture Recognition*, Zurich, Switzerland, pp.189–194 (1995).
6) Campbell, L.W., Becker, D.A., Azarbayejani, A., Bobick, A. F. and Pentland, A.: Invariant Features for 3-D Gesture Recognition, *Proc. 2nd IEEE Int. Conf. on Automatic Face and Gesture Recognition*, Killington, VT, USA, pp.157–162 (1996).
7) 牛田博英，山口 亨，高木友博：ファジー連想記憶システムを用いた動作認識，電子情報通信学会論文誌，Vol.J-77-D-II, No.8, pp.1571–1581 (1994).
8) Bobick, A.F. and Wilson, A.D.: A State-based Technique for the Summarization and Recognition of Gesture, *Proc. 5th IEEE Int. Conf. on Computer Vision*, Cambridge, MA, USA, pp.382–388 (1995).
9) 長屋茂喜，関 進，岡 隆一，向井理朗：A Proposal of Pattern Space Trajectory for Gesture Spotting Recognition, 画像の認識・理解シンポジウム（*MIRU'96*）論文集II，奈良，pp.157–162 (1996).
10) 渡辺孝弘，谷内田正彦：複数入力画像の固有空間法による実時間ジェスチャ認識，電子情報通信学会論文誌，Vol.J-81-D-II, No.5, pp.810–821 (1998).
11) 西村拓一，向井理朗，野崎俊輔，岡 隆一：低解像度特徴を用いた複数人物によるジェスチャの単一動画像からのスポッティング認識，電子情報通信学会論文誌，Vol.J-80-D-II, No.6, pp.1563–1570 (1997).
12) Davis, J.W. and Bobick, A.F.: The Representation and Recognition of Human Movement Using Temporal Templates, *Proc. 1997 IEEE Computer Soc. Conf. on Computer Vision and Pattern Recognition*, San Juan, Puerto Rico, pp.928–934 (1997).
13) Kurita, T. and Hayamizu, S.: Gesture Recognition using HLAC Features of PARCOR Images and HMM based Recognizer, *Proc. 3rd IEEE Int. Conf. on Automatic Face and Gesture Recognition*, Nara, Japan, pp.422–427 (1998).
14) 畠 直志，岩井儀雄，谷内田正彦：動き情報と情報圧縮を用いたロバストなジェスチャ認識手法，電子情報通信学会論文誌，Vol.J-81-D-II, No.9, pp.1983–1992 (1998).
15) Cutler, R. and Turk, M.: View-based Interpretation of Real-time Optical Flow for Gesture Recognition, *Proc. 3rd IEEE Int. Conf. on Automatic Face and Gesture Recognition*, Nara, Japan, pp.416–421 (1998).
16) Beauchemin, S.S. and Barron, J.L.: The Computation of Optical Flow, *ACM Computer Surveys*, Vol.27, No.3, pp.433–467 (1995).

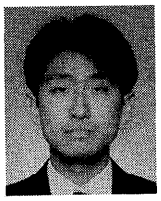
17) Liu, H., Hong, T. H., Herman, M. and Camus, T.: Accuracy vs Efficiency Trade-offs in Optical Flow Algorithms, *Computer Vision and Image Understanding*, Vol.72, No.3, pp.271–286 (1998).
18) 森田俊彦，沢崎直之，内山 隆，佐藤雅彦：カラートラッキングビジョン，第14回日本ロボット学会学術講演会予稿集，新潟，pp.279–280 (1996).
19) 井上博允，稲葉雅幸，森 武俊，立川哲也：局所相関演算に基づく実時間ビジョンシステムの開発，日本ロボット学会誌，Vol.13, No.1, pp.134–140 (1995).
20) 西川 敦，大西映生，宮崎文夫：連続動画像からのオプティカルフローを用いた身振り動作の曲率に基づく分割と認識，画像の認識・理解シンポジウム（*MIRU'98*）論文集 II，岐阜，pp.35–40 (1998).
21) 大西映生：オプティカルフローを用いた身振り認識—曲率に基づく分割と認識，修士論文，大阪大学大学院基礎工学研究科 (1998).
22) 松本吉央，坂井克弘，稲邑哲也，稲葉雅幸，井上博允：PCベースのハイパーマシン：知能ロボットの汎用カーネル，第15回日本ロボット学会学術講演会予稿集，東京，pp.979–980 (1997).
23) Nishikawa, A., Ohnishi, A. and Miyazaki, F.: Description and Recognition of Human Gestures Based on the Transition of Curvature from Motion Images, *Proc. 3rd IEEE Int. Conf. on Automatic Face and Gesture Recognition*, Nara, Japan, pp.552–557 (1998).
24) 西川 敦，西村正典，大西映生，宮崎文夫：オプティカルフローを用いた身振りインタフェース，平成10年電気関係学会関西支部連合大会論文集，大阪，p.G358 (1998).

(平成10年12月25日受付)
(平成11年 6 月 3 日採録)



西川 敦（正会員）

昭和42年生．平成7年大阪大学大学院基礎工学研究科物理系専攻博士課程修了．同年米国南カリフォルニア大学客員研究員．平成8年大阪大学基礎工学部助手．平成9年同大大学院基礎工学研究科助手となり現在に至る．この間平成6～8年日本学術振興会特別研究員．ロボットのステレオ視覚，人間とロボットのコミュニケーションに関する研究に従事．博士（工学）．日本ロボット学会，電子情報通信学会，IEEE各会員．

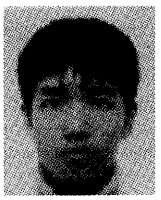


大西 映生

昭和47年生．平成8年大阪大学基礎工学部機械工学科卒業．平成10年同大大学院基礎工学研究科物理系専攻修士課程修了．現在，松下電工（株）に勤務．在学中は動画像に基づく身振り動作の認識手法の研究に従事．

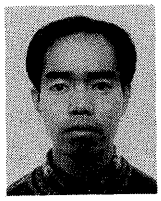
西村 正典

昭和50年生．平成10年大阪大学基礎工学部機械工学科卒業．現在，同大大学院基礎工学研究科システム人間系専攻修士課程に在学中．動画像を用いた身振りインタフェースの研究・開発に従事．



平野 敦士

平成11年大阪大学基礎工学部機械工学科卒業．現在，アンダーセンコンサルティング（株）に勤務．在学中は，身振りによるロボット誘導システムの研究・開発に従事．



小荒 健吾

昭和49年生．平成9年大阪大学基礎工学部機械工学科卒業．平成10年同大大学院基礎工学研究科システム人間系専攻修士課程修了．現在，同大学院基礎工学研究科システム人間系専攻博士課程に在学中．人間の身体運動の無標点画像計測に関する研究に従事．日本ロボット学会，日本機械学会，電子情報通信学会各会員．

宮崎 文夫

昭和27年生．昭和54年大阪大学大学院基礎工学研究科物理系専攻博士課程中途退学．同年大阪大学基礎工学部助手．昭和61年同助教授．平成3年同教授．平成9年同大大学院基礎工学研究科教授となり現在に至る．この間昭和62～63年米国カリフォルニア大学客員準教授．ロボットの知能化に関する研究に従事．工学博士．日本ロボット学会，システム制御情報学会，計測自動制御学会，IEEE各会員．