

ジェスチャ認識に基づく「棒」入力システムの研究

—人物追跡を中心に—

代表研究者 江島 俊朗 九州工業大学情報工学部教授

あらまし 人が歩きながら計算機に入力できる一手法としてジェスチャ認識に基づく「棒」入力システムは魅力溢れるものである。本研究では、そのような入力システムの実現に必要な基本技術である人物の動き追跡と手の動き追跡を同時に効率的に行う方式の研究を、「棒」入力システム実現の第一歩として行う。本論文では MITのMedia Lab で開発された Pfinderを基に、画像を用いて人物の追跡や動作追跡を行なう。Ptrackerを提案する。Ptracker の特徴は人の「動きの階層」に着目し、それに合わせて追跡処理の階層化を図った点にある。すなわち、Ptracker では、歩行している人物の胴体部分に焦点を当てた追跡処理が可能であるとともに、ジェスチャをしているときの手領域の追跡も可能である。「動きの階層」に合わせた処理を実現するために、人物が移動しているときは大まかな粗い部分の追跡を行い、停止してジェスチャをしているときには手の細かい動きを追跡するような機構をPtracker に組み込んだ。また それぞれの処理層は、追跡処理を行うフェイズと 初期化を行うフェイズで構成し、追跡処理中に blobモデルの整合性が失われたときには直ちに初期化を行うことで、素早い回復処理を実現した。

キーワード 動画処理、人物追跡、ジェスチャ認識、画像分割、動きの階層性

1 はじめに

計算機の発展により、画像を用いて人物の動きの追跡や認識・理解を実時間で行なうことができるようになってきた[4, 5, 6, 7, 8, 9]。MIT Media Lab.で開発されたPfinder[cite[1]]は実時間で動画を処理し、人物像の抽出及び追跡ができる手法として広く知られていて、評価も高い。Pfinderの特徴は、認識対象の各部分、および、背景を画像平面上の「位置の近接性(proximity)」と色空間で計られる「見かけの近接性」の基づいて簡潔に記述した「blob(球状の塊)モデル」を採用した点にある。認識を行うべき部分について、それぞれblobを作成し、入力画像の各画素は各blobとの尤度計算(blobとの近さを求める)を行ない、最も尤度が高い部分に割り当てが行われる。Pfinderは、このように人物像の抽出と部分の形状や位置の決定を一挙に行う手法であり、その簡潔さは魅力的である。

しかし、Pfinderの人物追跡では、注目すべき部分の大きさが互いに大きく異なる場合や部分の動きの特性が互いに大きく異なるような場合では、部分を近似するblobモデルの粒度に違いが生じ、尤度値の比較が意味をなさなくなる。その結果、画素の精度良い分類が難しくなり、画像からの部分の抽出がうまくいかなくなり、追跡に失敗することとなる。

この問題を解決するために、混合モデルの提案[3]や尤度計算に工夫を施す[2]等の改良案が提案されている。いずれも画像の確率モデルの改良により問題解決を図ったものである。これに対し、本論文では、対象の動き特性に沿うように処理システムの改良を図る。すなわち、人の「動きの階層」に着目し、「動きの階層」に沿うように追跡処理の階層化を図ることにより、上記の問題を解決しようとするものである。階層化により、歩行している人物の胴体部分に焦点を当てた追跡処理が可能であるとともに、ジェスチャをしているときには「動きの階層」の変化を察知し、動的に処理階層を変えることにより、手領域の追跡も可能にしている。「動きの階層」に合わせた処理を実現するために、人物が移動しているときは大まかな粗い部分による追跡を行い、停止してジェスチャをしているときには人物の肌色に注目して手の細かい動きを追跡するような機構を組み込んだ。また それぞれの処理は、追跡処理を行うフェイズと 初期化を行うフェイズで構成され、追跡処理中に blobモデルの整合性が失われたときには直ちに初期化を行うことで、整合性の素早い回復を可能にしている。本論文ではこの人物追跡システムをPtrackerと呼んでいる。

2 Pfinderについて

2.1 概要

MIT Media Lab.にて開発されたPfinderは、実時間で動画を処理する手法である。簡単な処理のわりに人体部位を安定して抽出できることから、広く知られておりその評価も高い。

Pfinderでは、頭、胴、足などの人体の各部分と背景を2次元画像上における「位置の近接性」と、色空間における「見かけの近接性」から作成されるblobモデルとして記述している。カメラからの入力画像1フレーム毎に、全ての画素に対して、各blobモデル(頭、胴、足、背景など)との尤度計算を行い、最も尤度の高い部分のラベルを割り振っていく。同じラベルが付けられた画素をまとめて、ひとつの部分の抽出と追跡を行なうとともに、blobモデルの更新を行ない、次の時刻の入力画像のラベリング処理の準備をする。

2.2 blobモデル

第 k 番目($k=1,2,\dots,N$)のblobを下記のように定義する。

$$\text{Blob}(k) = \{(\bar{x}_k, \Sigma_k^x), (\bar{y}_k, \Sigma_k^y)\}$$

$\text{Blob}(k)$ は、画像平面上のある点 $\bar{x}_k$ (2次元ベクトル)に重心を置く楕円領域(広がり共分散行列 Σ_k^x で規定される)を有するとともに、色空間上のある点 $\bar{y}_k$ (3次元ベクトル)に重心を置く、楕円体領域(その広がり共分散行列 Σ_k^y)をも有するものとする。

2.3 尤度の計算

画像平面上の位置 x 、画素値 y で表される画素が $\text{Blob}(k)$ に属する尤度の計算は以下の式で与えられる。

$$\begin{aligned} d_k = & -(\mathbf{x} - \bar{\mathbf{x}}_k)^T (\Sigma_k^x)^{-1} (\mathbf{x} - \bar{\mathbf{x}}_k) \\ & - (\mathbf{y} - \bar{\mathbf{y}}_k)^T (\Sigma_k^y)^{-1} (\mathbf{y} - \bar{\mathbf{y}}_k) \\ & - \log |\Sigma_k^x| - \log |\Sigma_k^y| \end{aligned}$$

各 $\text{Blob}(k)$ ($k=1, 2, \dots, N$)に対して尤度を計算し、 d_k を最大にするblobのラベル k をその画素に割り当てる。

2.4 初期化と更新

1. blobの初期化処理では、各画素毎にあらかじめ用意しておいた背景blobモデルとの尤度に基づき、人物領域と背景領域を分離する。
人物領域を雑音除去、クラスタリング処理等により初期blobを求める。
2. 逐次入力してくる画像の分割を各blobに対する尤度(類似度)に基づき行なう。すなわち、人物領域として切り出されたすべての画素について、blobとの尤度(類似度)を計算し、最も尤度(類似度)の高かったblobのラベルを画素に割り当てる。
3. 同じラベルを持った画素を集め、平均と共分散を計算し、blobの更新を行なう。

入力画像に対し、2と3の処理を繰り返し行っていくことで、動きのある対象物を追跡するための新しいblobが常に用意できる。更に、出力として、各blobを楕円で近似した画像を簡単に作れるので、対象物体を抽象化した映像表現が利用でき点もblob表現の利点である。また、各blobの動きを解析することにより、対象物の姿勢や動作の認識も可能である。

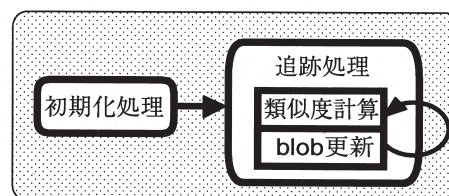


図1 blobの初期化と更新

2.5 Pfinderの問題点

本節では、Pfinderの問題点として以下の三つの点を取り上げる。次章で提案する人物追跡システムPtrackerは、以下の問題点の解決を試みたシステムである。

1. 人がドアを開けて部屋に入り、椅子に座って手話で話かけるという動作を考えた場合、歩いているのか止まっているのか

かを調査するために人物像を追跡する場合と、手話の自動認識のための手の追跡を行なう場合では、注目すべき対象の大きさや動きの特性(速度や加速度)が大きく違う。全身部分、あるいは手の部分のみの追跡はそれぞれ可能ではあるが、両者が混在した動きを追跡する場合は工夫が必要となる。

2. 胴体や脚の領域に対応するblobと手の領域に対応するblobでは領域の大きさ Σ_k^x 、 Σ_k^y が大きく異なる。尤度計算から、大きなblobに対する尤度の方が小さなblobに対する尤度よりも高い値になる確率が高くなる。結果として、小さなblobに対応する領域の抽出が難しくなり、追跡に失敗することが多く生じる。粒度の大きく異なるblobに関する尤度の比較には工夫が必要である。
3. 初期化処理と追跡処理の役割分担や負荷のバランスの問題は重要である。追跡に失敗すると、初期化処理に戻り態勢を整える必要がある。初期化処理と追跡処理の処理内容と遷移条件を検討する必要がある。

3 Ptrackerについて

前節(25節)で述べたPfinderの問題点の(1)と(2)を解決するため、注目する部分の大きさと動きの速度変化に相関関係があることを考慮に入れ、動作の階層性を導入する。すなわち、同じ層の動作は注目する部分の大きさは大きくは変わらず、従って動きの特性(速度や速度変化等)も余り変わらない。各層の動きの追跡では同じ粒度のblobが使えるので安定した部分の抽出が期待できる。層の遷移を適切に推定できれば階層遷移のある動作に対しても安定した追跡が可能になる。また、25節で述べた問題点の(3)に関しては、それぞれの層の追跡処理で、blobの初期化フェイズと追跡フェイズの2つの相を用意し、状況に応じて相を移ることにより、追跡処理が失敗したときに柔軟な対応をとれるようにする。相遷移の詳細は3.2節で述べる。

3.1 動きの階層に基づく処理の階層化

人体を対象として動画像を処理する場合、歩行等の人体全体の動き、身振り認識のための手の動きや人物特定のための顔の表情、サインランゲージ認識のための指先の動きなど、種々の動きが処理対象になる。本研究では人体の動きは、人物全身の動き、胴や足、腕の動き、手の動き、指や口の動き、といった階層を持つものとして考える。階層ごとに注目すべき部分が異なり、その大きさや動きの特性(速度や加速度)も異なる。注目する部分の大きさが大きい程下位の階層であるとする、図2のような階層を考えることができる。このような階層を考えると、注目すべき動きの階層の下位の階層の動きは静止状態にあることが多く、画像処理の点で下位の層からの干渉は少ないことに気が付く。このことから、下位層から動きを捕らえ、下位の層の動きの状態に基づいて上位の層への動きの遷移が起こっているかどうかを推定し、上位層の動きを捕らえる座標系を設定することができる。

3.2 処理階層における層遷移の条件

動作の階層に基づいて行った処理の階層化について、層間の遷移、及び層内の初期化相・追跡相の相遷移を図3に示す。

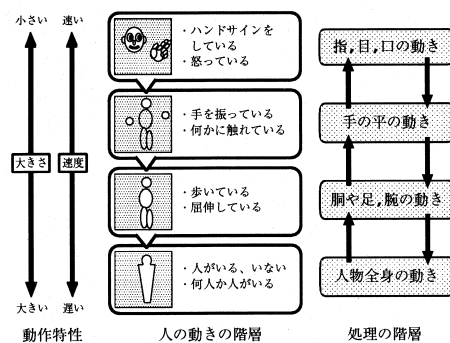


図2 階層性

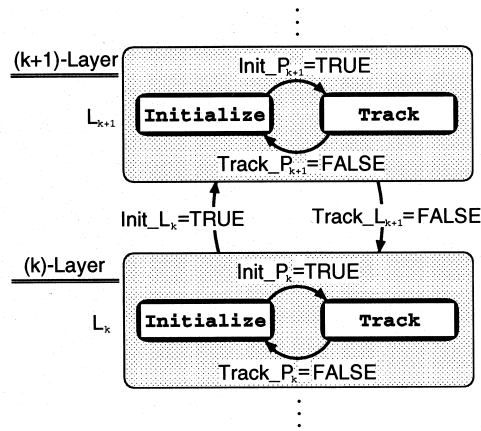


図3 状態遷移図

- Init_P\\
初期化フェイズの状態変数。初期blobモデル作りが成功したときTRUE、失敗したときFALSEとなる。
- Track_P\\
追跡フェイズの状態変数。順調に追跡が行えているときTRUE、追跡に失敗したときFALSEとなる。
- Init_L\\
上位層への遷移を決定する変数。注目している層の Init_PがTRUEで、かつ遷移条件が満たされている場合にTRUE、そうでない場合FALSEとなる。
- Track_L\\
階層内の処理の継続を表す変数。下位層からの遷移条件が満たされている場合TRUE、そうでない場合FALSEとなる。

上位層への遷移は、下位層での追跡処理が正常に行われているときにのみ可能となるため、階層内で初期化・追跡処理があるのと同様に、階層間でも下位の層が初期化処理、上位の層が追跡処理を行うと考えることができる。

本システムは現在、

- 人領域抽出・追跡部分
- 手領域抽出・追跡部分

で構成され、それぞれ人体のおおまかな動作を得るための[頭][胴][足]を追跡する下位の処理層と、ジェスチャ認識や仮想空間での動作を認識するための[顔][手]を追跡する上位の処理層から構成されている。この2つの処理層の遷移のための判断は、下位の処理層での[頭]および[顔]の動きの大きさをを用いている。

4 下位層の動き追跡処理

下位層では、人物がフレームインし移動している間、人物の[頭][胴][足]部分の追跡を行う。

4.1 人物領域抽出手法

4.1.1 処理の流れ

人物領域における初期化フェイズでは、以下に述べるような順序で処理を行う。

1. あらかじめ保存しておいた背景画像と、入力画像との差分をとり、人物領域を特定する。
2. 人物領域を人物抽出フィルターにより[頭][胴][足]と思われる部分を抽出する。
3. 抽出した[頭][胴][足]の領域を基にblobモデルを作成し、追跡フェイズに遷移する。

4.1.2 人物領域の特定

色情報をもとに、入力画像と背景画像との差分を求め、人物領域を特定する。人物領域を表す画素の数が、ある閾値を越えたら人物が画像内にフレームインしたものとみなす。ここでは、背景のみの画像内に人物が外から入って来る状況を想定しているので、画素数が閾値を越えた時点で、領域を特定してしまうと、図4のように人物が完全にフレームインしていない状態でblobを作成してしまう恐れがある。

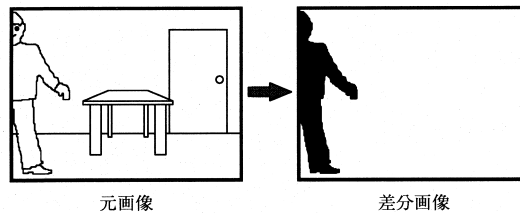


図4 不完全な人物領域

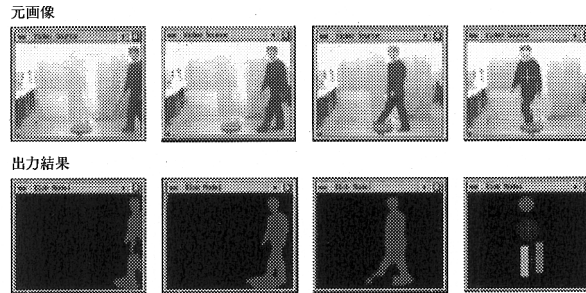


図5 人物領域特定処理の遅延

4.2 人物領域追跡手法

4.2.1 処理の流れ

追跡フェイズでは、人物領域に対してblobモデルとの尤度(類似度)に基づく分類を行なう。分類した領域に基づきblobモデルを更新する。この処理を繰り返すことで、人体の各部分の動きをblobの動きとして捕らえ、追跡する。処理の順序を以下に示す。

1. 過去のblob中心位置の時系列から、現在のblob位置を推定する(Kalman Filter を使用)。
2. 抽出した人物領域の画素に対して各blobモデルとの類似度を計算し、最も高い尤度(類似度)のblobのラベルを割り付ける。
3. ラベリング結果を基にblobの更新する。
4. 更新した blob間の整合性を調べ、問題が無ければ処理を繰り返す。整合性が保たれていなければ初期化フェイズに戻りblobを作り直す。

4.2.2 blobモデルの整合性

blobモデルの更新が終わると、更新したblobモデルについて、その整合性について検証する。以下に示す項目について調査する。

●blob間の相対位置関係

胴blobの中心位置を基準として頭blob、及び足blobへの相対ベクトルを求め、その値が初期blob作成時に得られるものに比べて、違いが大きい場合、整合性が失われたものと判断する(図6(左))。

●各blobの面積の大きさ

各blobを構成する画素数がある一定の基準よりも小さくなった場合、正確な抽出に失敗したものと判断する (図6(中央))。また、blobの面積に極端なアンバランスが生じた場合は整合性が失われたと判断する(図6(右))。

上記のチェックで整合性が失われたと判断された状況でそのまま処理を続けると、次フレーム以降の認識結果が正しく行われなくなるので、初期化フェイズへ移行し再度モデルを作成し直す。

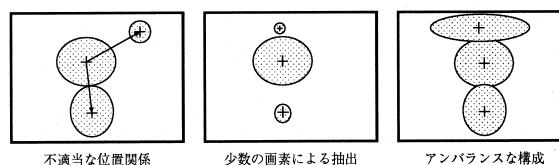


図6 整合性を失った状態

5 上位層の動き追跡処理

人物追跡の下位層の処理中に、上位層への遷移の条件が満たされたら(例えば、blobが静止状態とき)、上位層の動き追跡処理に遷移する。

5.1 手領域抽出法

5.1.1 処理の流れ

手領域における 初期化フェイズでは、以下に述べるような順序で処理を行う。

1. あらかじめ保存しておいた背景画像と、入力画像との差分をとり、人物領域を特定する。
2. 閾値処理により人物領域から抽出された肌の画素をラベリングし、[顔][左手][右手]部位の特定を行う(初期化フェイズ)。
3. [顔]と[両手]が認識されたらblobモデルを作成し、追跡フェイズに移行する。
4. 3つの部位が確実に抽出できなかった場合、一定の回数 抽出処理を繰り返す。抽出が失敗したとき、人物領域初期化フェイズに移る。

5.2 手領域抽出

5.2.1 肌色情報取得

本システムでは、手が肌色であるという色情報をもとにして、手領域の特定を行っている。YUV表色系を用い、予め肌色を表す平均的な色情報を内部に保存しておく。カメラの状態や撮影場所の全体的な明るさなどにより小さくない変動を吸収するため、肌色の平均値と分散をもとに肌色領域の分類を行なう。

5.2.2 ラベリング

肌色画素で構成されるblobモデル(顔、左手、右手)作成のために、肌色画素の集合にラベル付けを行う。伸縮膨張処理を施した後、肌色領域が結合したいいくつかの塊のうち面積が大きい3領域に[顔][左手][右手]のラベルを割り付ける。

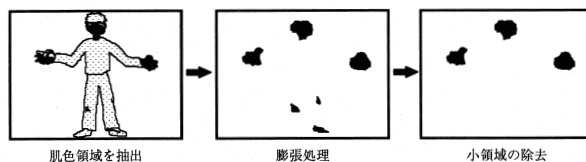


図7 肌色領域抽出

5.3 手領域追跡手法

5.3.1 処理の流れ

追跡フェイズでは、人物領域のうち肌色と認識された画素について、blobモデルとの尤度(類似度)を計算し新たなラベリングを施し、blobモデルを更新する。人物領域については、[顔][両手][背景]blobのみのマッチングを行い、[頭][胴][足]blobは背景と見なす。

1. 過去のblob中心位置をもとに、blobごとにマッチングを行う範囲を定める。
2. 肌色領域を抽出し、blobモデルとの尤度(類似度)を計算し、最も高い尤度(類似度)のblobをラベリングする。
3. ラベリング結果から[手][顔]blobモデルの更新する。
4. blob間の整合性が保たれていれば処理を繰り返す。整合性が保たれていなければ手領域初期化フェイズ及び人物領域初期化フェイズでblobモデルを作り直す。

6 実験、結果

6.1 実装における仮定、および制限

Ptrackerで人物領域認識、手領域認識の実験を行なう際に、以下のような仮定や制限を設け、処理量の削減を図った。

- システム起動後数秒間は背景画像を取得するために必要な時間であるとし、人物はフレームインしてこないものとする。
- [顔]、[手]blob追跡のために利用する肌色情報は処理の前にあらかじめプログラム内部に読み込んでおくとする(前回の

情報があればそれをする事も可能)。

- 画面内に複数の人物が現われた際、人物がすれ違う状況での追跡は可能であるが、人物が重なった状態が続くと、いくつかのblobが一方の人物に持って行かれることがある。このためフレームインしてくる人物は高々1人であるとする。
- 対象物を人物のみとしているので、全ての人体部分はその位置関係のみが変化し、色や形状の情報は変化しないものとする。よって、属性を与えたblobの、色と位置に関する共分散パラメータは初期化处理によって決められた値に固定する。
- 人物が移動しながらジェスチャ等を行うことは無いと考え、手領域の追跡は人体位置が静止した状態でのみ行うものとした。
- 画面内に現われる肌色の領域は人物の顔と両掌に限定し、服の色は背景とは異なるものとする。

6.2 実験環境

人間の動作を行うのに十分に広い部屋で、以下に示す環境のもとに実験を行った。

表1 動作環境

計算機	SGI 社製 O2
CPU	R5000(180MHz)
カメラ	O2Cam
取り込み画像サイズ	160x120

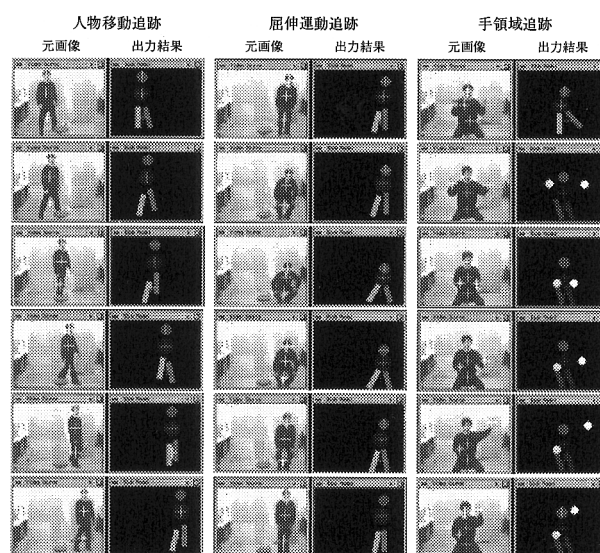


図8 実験結果

6.3 動作確認

部屋の中を自由に動き回る1人の人物の人体部分を、リアルタイムで追跡する実験を行なった。人物が移動している状態では、[頭][胴][足]の各部位を追跡し人体のおおまかな状態を知ることができ(図8(左))、屈伸運動のような画像内の人体の大きさが変化する動きも追跡できることを確認した(図8(中央))。また、人物が一定時間静止したとき、肌色情報に基づいて[手]の動きを追跡できる事を確認した(図8(右))。また、その状態から再び人物が移動を始めると人物抽出・追跡フェイズに切り替わることから、人領域処理を行うフェイズと手領域処理を行うフェイズをうまく遷移できていることも確認できた。さらに、それぞれのフェイズにおいてblobの整合性が失われると、初期モデルが作り直されることから、階層内で抽出処理と追跡処理の間を繰り返し遷移していることも分かった。秒間8～10フレームの処理を行うことができ、実時間での追跡ができた。

7 考察

7.1 動作の階層化

人体部分の動きの特性は、主に、質量に依存すると考えられる。たとえば全身の動きに注目した場合、ゆっくりとした移動速度でほぼ等速運動と見なすことができるが、手などの小さな部分においてはその移動速度は大きく変化する。Pfinderでは対象の動きの特性は考慮せず、全ての対象に同一の処理を施しblobを一挙に決定しようとしていた。一方、Ptrackerでは動作を階層化して考えることで、それぞれの階層に応じた処理を行い、動きの特性が異なる対象物に対しても柔軟で安定した追跡が行えるようになった。ある処理層でblobが整合性を失ってモデルを作り直すときも、既に求まっている下位層のデータは整合性を保ったままであり、再初期化を行うのは注目している処理層の対象だけで良い。これにより冗長な処理を行わずに済み、処理速度の向上を実現できる。

8 まとめ

人が歩きながら計算機に入力できる一手法としてジェスチャ認識に基づく「棒」入力システムは魅力溢れるものである。本研究では、そのような入力システムの実現に必要な基本技術である人物の動き追跡と手の動き追跡を同時に効率的に行う方式の研究を、「棒」入力システム実現の第一歩として行った。

本論文では MIT Media Lab.で開発された実時間動画画像処理システム Pfinderを改良し、人物追跡に特化した実時間人物追跡システム Ptrackerを提案し、計算機に実装し性能の確認を行なった。Ptracker の特徴を以下に述べる。

- 動き階層性を考慮し、処理の階層化を図った。
- 動きの状態遷移を推定することにより、処理層の自動切替えを実現した。
- 各処理層では初期化処理と追跡処理で構成し、追跡に失敗したときの回復処理の実現を図った。

今回提案した Ptracker を活用することにより、「棒」入力システム\cite{Stick}における「棒」の軌跡を安定して追跡できる見通しをつけることができた。

参考文献

- [1] C.R. Wren, A. Azarbayejani, T. Darrell, and A.P. Pentland, “ Pfinder: Real-Time Tracking of the Human Body ” *IEEE Trans. Pattern Analysis and Machine Intelligence* , Vol. 19, No. 7, pp. 780-785, July 1997.
- [2] 蓬原宏史、乃万司、大橋健、江島俊朗 “ 動画画像から人物像の抽出 ” 第三回映像メディア処理シンポジウム, pp17-18 1998
- [3] W.E.L. Grimson, C.Stauffer, R.Romano and L.Lee, “ Using adaptive tracking to classify and monitor activities in a site ” *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition* , pp22-29, 1998.
- [4] D.M.Gavrila D.M.Gavrila “ The Visual Analysis of Human Movement: A Survey ” *Computer Vision and Image Understanding* Vol.73, No.1, pp.82-98, 1999.
- [5] Tony Jebara and Alex Pentland “ Action Reaction Learning: Analysis and Synthesis of Human Behaviour ” MIT Media Lab. Perceptual Computing Technical Report \#452 1998.
- [6] Ismail Haritaoglu, David Harwood and Larry S. Davis “ W⁴S: A Real-Time System for Detecting and Tracking People in 2 $\frac{1}{2}$ D ” pp877-892 1998.
- [7] K.Toyama “ Prolegomena for Robust Face Tracking ” *Microsoft Research Technical Report MSR-TR-98-65*, 1998.
- [8] J.K.Aggarwal and Q.Cai “ Human Motion Analysis: A Review ” *Computer Vision and Image Understanding* Vol.73, No.3, March. pp.428-440, 1999.
- [9] Alex Pentland “ Looking at People: Sensing for Ubiquitous and Wearable Computing ” *IEEE Trans. on PAMI* Vol.22, No.1, January. pp.107-119, 1999.
- [10] 大橋 健 仲程 啓 吉田 隆一 江島 俊朗 “ 『棒』入力システムのためのジェスチャ認識の実現 ” *情報処理学会論文誌* Vol.40, No.2, January. pp.567-576, 1999.

＜発 表 資 料＞

題 名	掲載誌・学会名等	発表年月
「棒」入力システムのためのジェスチャ認識の実現	情報処理学会論文誌 Vol. 40, No2, pp. 567-576	1997年10月