

研究テーマ 顔および頭部の検出に関する研究

研究者 石井洋平 財団法人ソフトピアジャパン
金川 誠 三洋電機株式会社

雇用研究員
共同研究員

フェーズ I

1 研究の概要

顔画像処理技術を用いたマンマシンインタフェースや監視システムにおいて、先ずカメラに撮影された人物の顔領域の検出が行われる。顔検出の実環境における応用を考えた場合、長時間にわたって安定に動作する手法が求められる。そのため、照明や背景などの周囲の変動から受ける影響が小さい手法が必要となる。

顔検出のための探索領域の設定に肌色情報を利用することが考えられる。肌色情報は、人物検出を行う際に有用であるが、カメラや照明条件によって検出に用いるパラメータを適切に設定する必要がある。そのため、実環境で利用するためには、動的に肌色情報を獲得し、検出パラメータの調整を行わなければならない。本研究では、肌色情報と動き情報を併用して設定した探索領域から顔画像パターンを検出し、検出結果を利用して肌色領域の検出パラメータを動的に調整する手法に関して検討する。本手法では、顔が検出された領域の画素値を用いて肌色検出パラメータの調整を行うため初期設定が不要であり、カメラや照明条件が変化した場合にも、人手によるパラメータ調整が不要となる利点がある。

2 研究の目標

肌色を用いた顔検出手法は、照明条件に応じて検出に用いるパラメータを適切に設定する必要がある。一方、顔画像パターンを用いた検出手法は、入力画像全体を探索すると計算コストが大きくなる。本研究では、両手法を組み合わせ、肌色情報により探索領域を設定し、顔画像パターンを用いて、探索を行う手法を提案する。提案手法では、照明変動のある環境下で、動的に肌色パラメータを取得し、調整を行うことによって顔探索領域を設定する。提案手法の有効性を、固定的に肌色パラメータを設定する手法と比較によって検討する。

3 実施内容

肌の色には人種や個人による差があるため、色空間中で肌色を固定的に定義することはできない。また、実環境下では様々な照明条件が存在し、撮像するカメラの特性にも影響されるため、その見え方も多様である。そこで、顔画像パターンや動き情報を利用して動画像中から検出された顔領域内の色を肌色として獲得し、動作環境において肌色を検出するパラメータを動的に調整することによって環境変化や個人差に対応する。以下では、本手法での画像処理概要、設定領域内での顔検出、肌色検出パラメータの設定に関して述べる。

3.1 処理概要

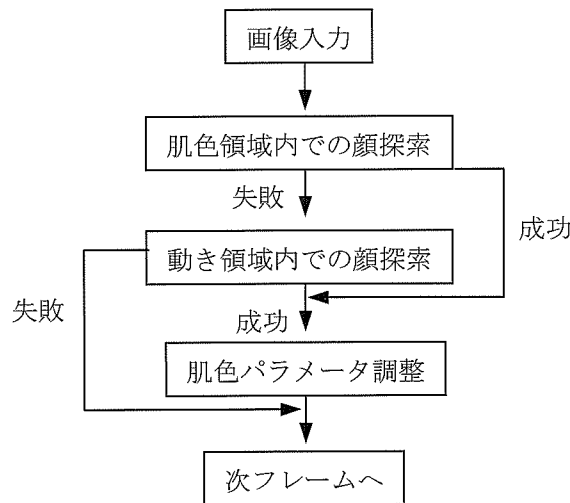


図 1 処理概要

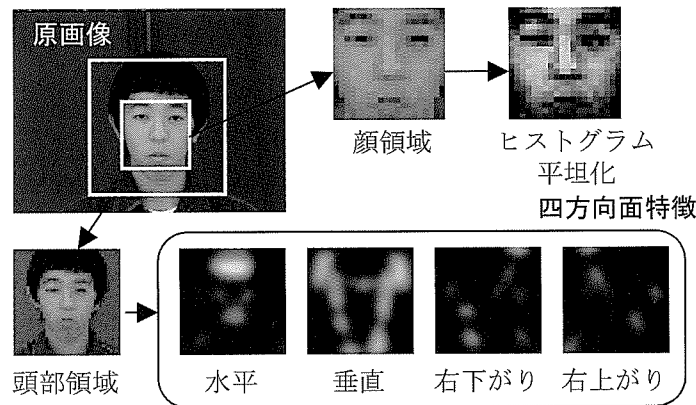


図 2 顔領域画像と頭部領域の四方向面特徴

本手法のフローチャートを図 1 に示す。まず、入力画像から肌色情報を用いて探索領域を設定し、設定した探索領域内から顔画像パターンを検出する。探索領域から顔画像の検出に成功した場合には、肌色の検出パラメータの調整を行う。肌色領域からの顔検出に失敗した場合には、現在の肌色検出パラメータが不適切であり、顔が存在するにも関わらず、探索範囲に含まれなかったことが考えられる。このような可能性を考慮し、肌色領域内での顔検出に失敗した場合には、人物の動き情報から顔探索領域を設定し、顔画像パターンの検出を行う。その結果、動き領域から顔を検出した場合には、以降のフレームで適切に肌領域が検出できるように肌色検出パラメータを調整する。

3.2 探索領域の設定

探索領域の設定は、顔検出を効率的に行うために重要となる。肌色情報を用いた探索領域の設定では、それまでのフレームの検出結果を利用して設定されたしきい値によって検出された肌色領域から探索領域を設定する。肌色検出に用いるしきい値の設定については後述する。また、動き情報を用いた探索領域の設定では、フレーム間差分によって検出された動き領域から探索領域を設定する。何れの場合においても、肌色領域や動き領域が検出されるのは顔や人物の一部であることが多いと考えられるため、検出領域の近傍を探索領域に設定した。

3.3 顔検出

設定領域内からの顔検出手法として、本稿では SVM (Support Vector Machines : サポートベクタマシン) を利用した手法を用いた。基本的には線形の識別手法である SVM は、カーネル法の導入によって高い汎化能力を持った非線形な識別手法へと改良された。近年では、高い識別率を達成する 2 クラス識別手法として注目を集めており、顔画像と顔以外の画像を学習する

ことによって、顔検出にも応用されている。

今回、検出対象とする顔画像はほぼ正面から撮影した画像とした。固定カメラによる人物検出という応用を考えた場合には、人物とカメラの距離が離れ、顔画像領域の解像度は低い場合も多いと考えられる。そのため、検出する顔画像は 20×20 (pixels) の領域内に、両目と口が含まれる大きさとした。肌色領域内で顔検出に用いる特徴量は、候補領域の輝度値をヒストグラム平坦化した 400 次元のベクトルを用いた。動き領域内で顔検出を行う際には顔領域の輝度値に加え、頭部領域の 40×40 (pixels) から抽出した四方向面特徴を併用することによって過検出の低減を図った。四方向面特徴は、画像パターンのエッジに着目した特徴量であるため顔の周辺を含む頭部領域のエッジ特徴を利用できると思われる。検出に利用する際には、四方向面特徴を 8×8 に低解像度化し、256 次元のベクトルとした。図 2 に顔領域の画像と四方向面特徴の例を示す。

学習に用いた顔画像のデータは、HOIP データベースから 900 枚、WWW から取得した画像から 1144 枚、新聞や雑誌などからスキャナを用いて取得した画像より 1484 枚の合計 2644 枚である。顔以外の画像は学習の初期に 600 枚収集した。その後、WWW などから得た顔が撮影されていない画像を用いて顔検出を行い、誤検出した領域を再度学習する bootstrap 法を用いて学習を行った。

3.4 肌色検出パラメータの調整

人物の肌領域を検出するための色空間として、正規化 r - g 空間を用いた。正規化 r - g 空間では、肌色モデルを正規分布として表現することができるという研究結果が得られている。また、計算コストが小さいという利点もある。

肌色検出パラメータの調整は次の手順で行う。まず、 r および g に対して、検出された顔領域の画素から現在の肌色分布のヒストグラム H_c を作成する。次に、このヒストグラムをこれまでに得られているヒストグラム H_t に以下の式で加え、ヒストグラム H_{t+1} を得る。

$$H_{t+1} = (1-a)H_t + aH_c \quad (1)$$

但し、式(1)中で $0 < a < 1$ である。このようにして得られた r および g に対するヒストグラムから各色の平均値 m_r 、 m_g と分散値 σ_r 、 σ_g を算出し、これらの値を用いて、以下の条件を満たす画素を次のフレームで肌色画素とする。

$$\begin{cases} m_r - \sigma_r < r < m_r + \sigma_r \\ m_g - \sigma_g < g < m_g + \sigma_g \end{cases} \quad (2)$$

4 結果

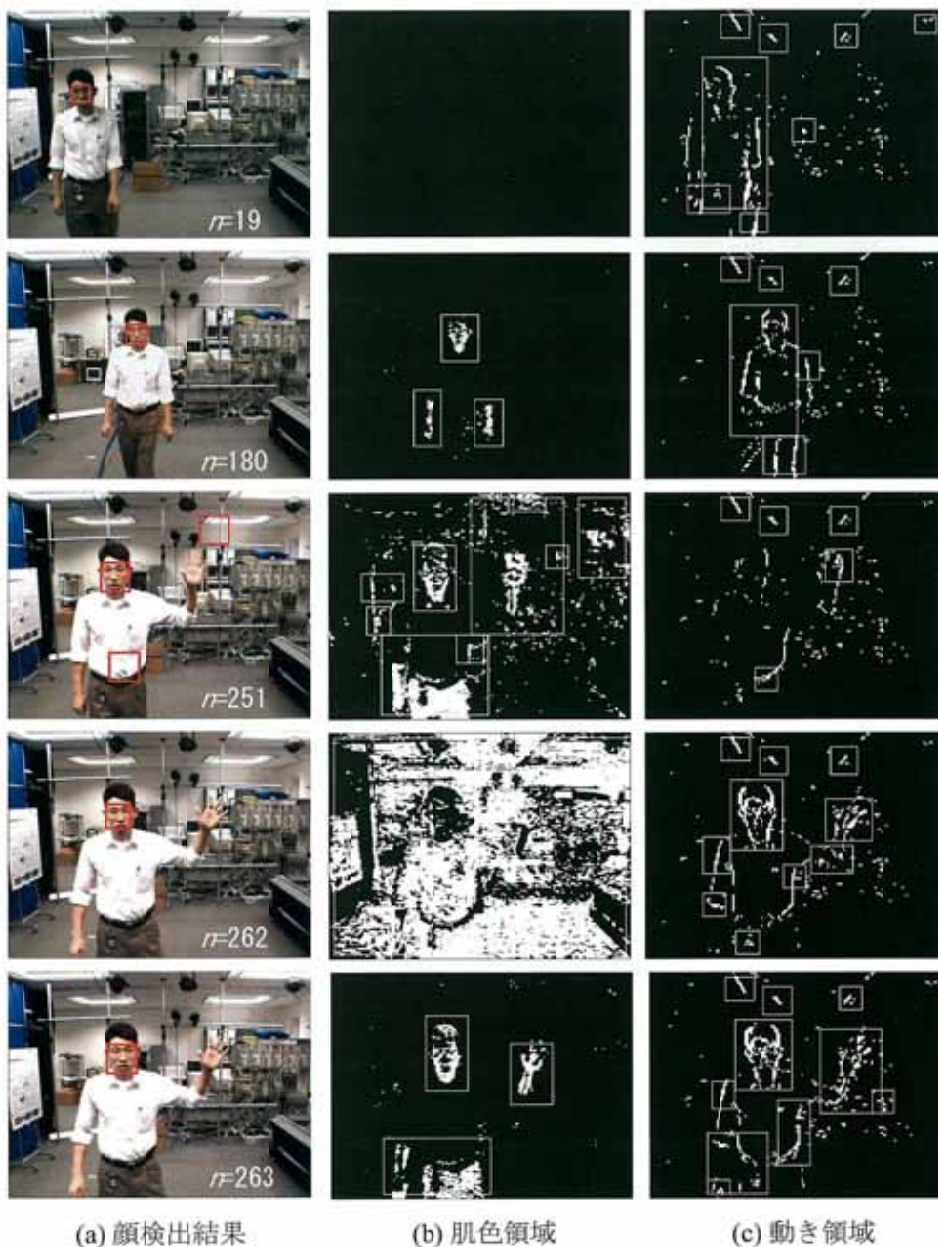
室内環境下で、照明条件を動的に変動させ、顔検出実験を行った。天井から吊り下げられた照明器を天井に向けて点灯させ、反射光を追加照明として利用することによって被験者に対する照明条件を変動させた。肌色検出パラメータは平均値、分散値の初期値を共に 0 とし、顔検出結果から獲得することとした。

図 3 に動画像より得られた顔検出結果、検出された肌色領域および動き領域を示す。図 3 において、顔検出結果中の矩形は検出された顔領域を示し、肌色および動き検出結果における矩形は、設定された探索領域をそれぞれ示す。また、図 4 に r と g について、顔検出結果を用いて調整された肌色平均値の推移を曲線で示す。図 4 中で手動切出しと表示された曲線は、手動で切出された顔領域の r と g の平均値を示す。照明はフレーム 115 からフレーム 272 の間追加された。

図 3 および図 4 より、初期値を全て 0 とした肌色検出パラメータが、動き領域からの顔検出結果によって肌色画素を獲得し、設定できたことが分かる ($n=19$)。照明を追加している間は、天井からの反射光によって顔領域が明るくなったため、 r および g の値が変化したが ($n=170$)、肌色パラメータの調整を行うことができた。また、顔検出時に生じた過検出によって肌色パラメータが誤った値に設定されることもあったが ($n=251$)、後のフレームで動き領域から正しく

顔を検出することにより($n=262$)、適切に調整することができた ($n=263$)。

実験では、動き領域を用いた顔探索領域より顔検出を行うことによって肌色画素を動的に獲得し、肌色パラメータの検出を行うことができた。また、過検出によって生じる誤った肌色パラメータの設定値に対しても、動き領域からの顔検出によって適切な値に再調整できることを確認した。しかし、本手法では、過検出の影響を強く受けるため、顔検出の高精度化は今後の課題である。



(a) 顔検出結果 (b) 肌色領域 (c) 動き領域

図3 設定された探索領域と顔検出結果

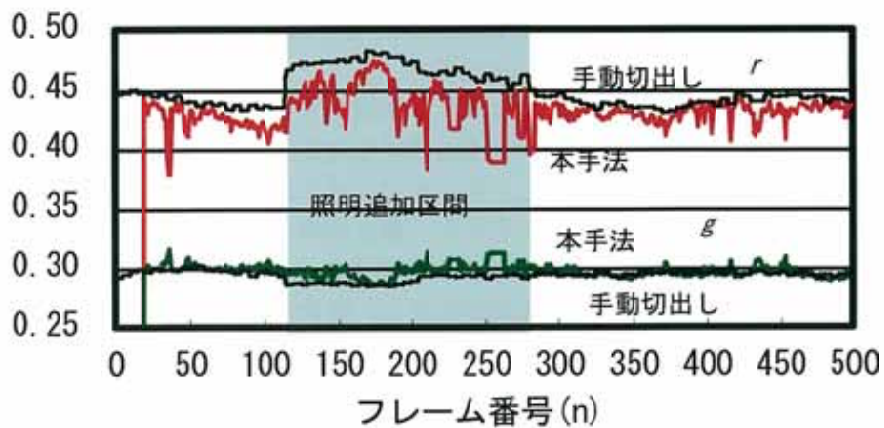


図4 肌色平均値の推移

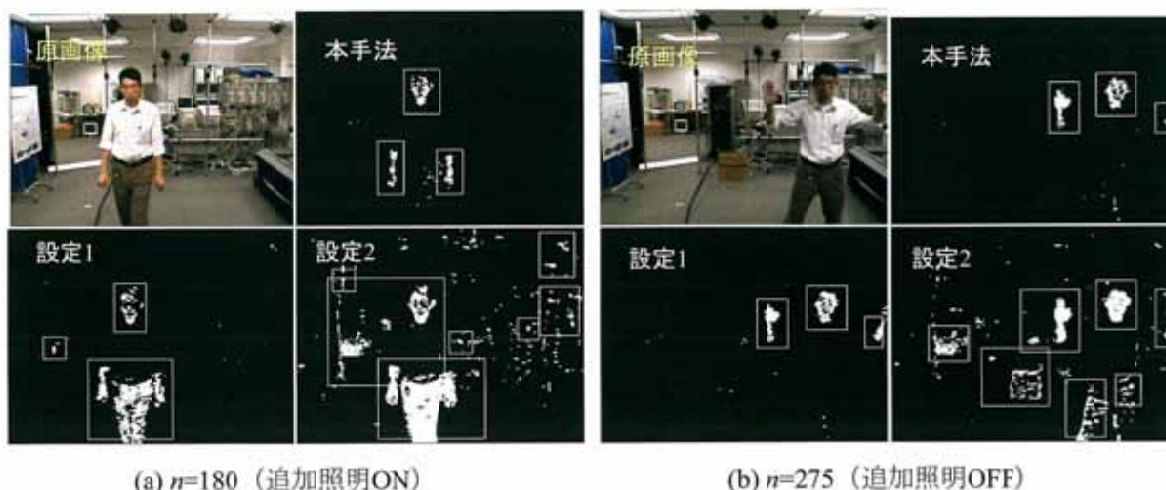


図 5 従来手法との比較

図 5 に従来手法である固定の肌色パラメータを用いた場合の肌色領域検出結果と本手法での肌色領域検出結果を示す。図中の矩形は、検出された肌色領域から設定される探索領域を示す。検出パラメータは、図 4 に示した実験結果を用いて以下のように設定した。

- ・ 設定 1 : $0.40 < r < 0.48$, $0.28 < g < 0.32$
- ・ 設定 2 : $0.38 < r < 0.50$, $0.26 < g < 0.34$

照明が追加されている場合は、固定しきい値の場合には、被験者の衣服（茶色）が検出されている（図 5(a)）。照明が追加されていない場合は、設定 1 と本手法でほぼ同じ結果が得られたが、設定 2 では被験者の後方に置かれた段ボール箱が検出されるなど過検出が生じている（図 5(b)）。また、これらの固定パラメータの手法では、照明条件と人物の肌色から適切な値を設定しなければならず、適切な値の決定も困難な問題であるが、本手法ではそのような事前の設定作業は不要である。

フェーズ II

1 研究の概要

フェーズ I では、変動環境下における顔検出に関して、肌色情報と顔画像パターンを用いる手法を検討した。しかしながら、顔画像パターンの探索には、SVM による手法を用いたため、肌色類似色領域が大きい場合によっては計算コストが大きくなるという問題点も残っている。そこで、SVM よりも計算コストが小さい線形判別分析を用いて高速な検出法を検討する。

一方、マンマシンインタフェースやセキュリティシステムにおいて、必ずしもカメラに人物の顔が撮影されるとは限らない。複数のカメラを協調させる場合でも、顔が撮影できないカメラが存在することは充分考えられるため、検出精度を向上させるためにも顔だけでなく、頭部を検出することは有用と思われる。また、サベイランスシステムのような応用を考えた場合にも、カメラと人物の相対位置は多様であるため、顔と頭部の検出は有効と思われる。

本研究では、顔および頭部を検出する手法を提案する。提案手法では、四方方向特徴と線形判別分析を用い、実時間処理を重視した実装を行う。動画像を入力画像とし、動き領域の検出による顔探索領域の削減と検出位置の連続性を考慮した後処理によって処理の高速化と高精度化を図る。また、提案手法をより高速化するために LSI 化を目指したアルゴリズムの再検討を行い、FPGA に実装を試みる。

2 研究の目標

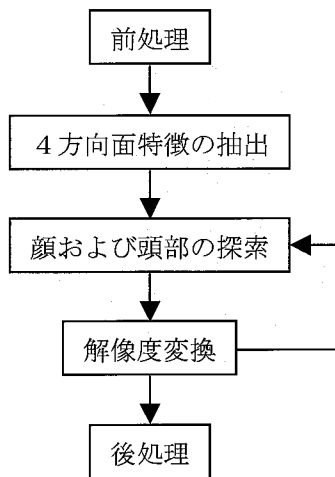


図6 検出処理の流れ

肌色情報と SVM を併用した顔検出手法では、検出された肌色領域が大きい場合に、計算コストが大きくなる問題があった。また、頭部の検出を行うためには、頭髪が影響することもあり、肌色はあまり有効ではない。そのため、肌色情報を用いることなく、画像パターンを用いて実時間処理が可能な顔および頭部の検出手法を検討する。提案手法では、動画像を対象とした実時間性を重視した手法を目指す。動画像を対象とするため、検出対象の動きや連続性を考慮することによって検出精度の向上を図る。提案手法の高速化においては、LSI 化のためにアルゴリズムを再検討し、FPGA に実装した上で評価を行う。

3 実施内容

3.1 顔および頭部の検出

本研究では、多様なカメラと人物の相対位置が考えられる状況での人物検出として、顔および頭部を検出する手法を提案する。そのため、パターン認識手法による顔および頭部の検出に加え、人物の動きや連続性に着目した前処理と後処理を行う。顔および頭部の検出では、四方向面特徴を特徴量とし、線形判別分析を用いて顔と頭部、非顔、非頭部を識別する。四方向面特徴と線形判別分析を用いた手法は、個人識別や顔向き推定において有効性を示している。

3.1.1 検出処理の流れ

顔および頭部の検出の流れを図6に示す。まず、前処理によって探索領域を設定する。次に、入力画像から四方向面特徴を抽出する。その後、抽出された四方向面特徴を線形判別分析で顔または頭部であるか否かを判定する走査探索によって人物検出を行う。この時、解像度を変換した四方向面特徴を用い、探索を繰り返すことによって、撮影される人物の画像解像度変化に対応する。最後に、後処理によって検出結果の評価と統合を行う。

3.1.2 前処理

前処理では、色情報や動き情報を用いることによって、顔や頭部が存在しないと思われる領域を探索領域から除くことによって計算量を削減できる。今回は、動画像を入力としているため、動き検出結果を用いて探索領域を設定した。

動き検出には3フレーム間差分を用いた。3フレーム間差分は、2値化した前後のフレームとの差分結果の論理積の算出によって、動き領域を精度良く検出できる利点がある。通常のフレーム間差分と比較すると計算量が増加する問題もあるが、入力画像を1/4に低解像度化した画像を用いることによって計算時間の低減を図る。顔および頭部の検出では、動き検出領域に膨張処理を施し、探索領域として設定する。

3.1.3 四方向面特徴

四方向面特徴は、ロバストな特徴量としてパターンマッチングに用いられている。四方向面特徴の抽出手順は以下の通りである。まず、入力画像に対し、Prewitt オペレータで求めた各画

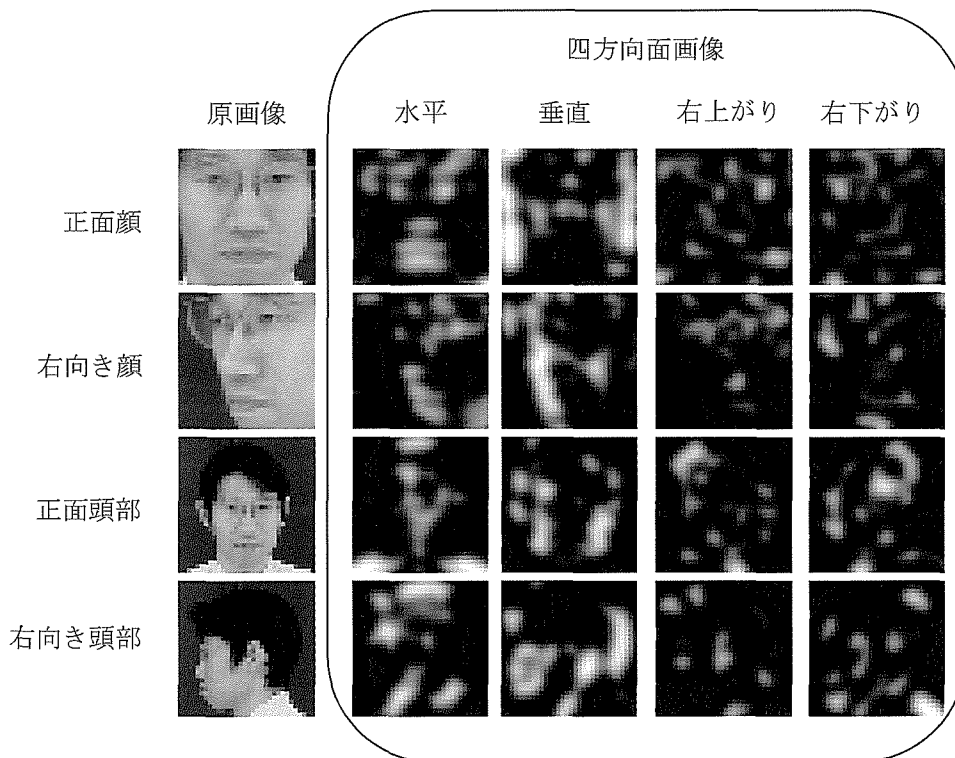


図7 四方向面画像の例

素のエッジ勾配をエッジ方向に基づいて、水平、垂直、右上がり、右下がりに分解し、四方向面画像を得る。得られた四方向面画像にガウシアンフィルタを適用し、ぼかし処理を行った後に低解像度化する。その後、各方向面の濃淡値を特徴量することにより四方向面特徴を得る。

四方向面特徴は四方向のエッジ画像を低解像度化することにより、各方向のエッジ情報を維持したまま頑健性が向上する特徴がある。また、低解像度化によって、マッチングや識別に要する計算コストを低減できる利点もある。本稿では、検出する矩形の入力画像内でのサイズを 32×32 とし、 8×8 に低解像度化した。そのため、顔と頭部の識別に用いる特徴量は 256 次元 ($8 \times 8 \times 4$) となる。図7に顔と頭部の4方向画像の例を示す。上2段は正面と右 45° を向いた場合の顔領域、下2段は正面と右 90° を向いた画像の頭部領域の原画像と4方向面画像である。この図より、顔は目や口等の顔器官の水平エッジに、頭部は楕円状の輪郭形状が各方向エッジに抽出されていることが分かる。四方向面特徴は各エッジ画像を低解像度化したものとなる。

3.1.4 線形判別分析

線形判別分析では、クラス内分散を最小化し、クラス間分散を最大化するという基準に従って、学習サンプルから最適な判別空間を与える係数行列 A を求める。入力ベクトル x は、判別に適したベクトル y へと $y = Ax$ によって線形写像される。判別では、写像変換された y と学習サンプルから得られた各学習クラスの写像ベクトルの平均値 \bar{y}_i を用いて、以下の式で得られる判別スコア D_i を算出する。

$$D_i = |y - \bar{y}_i|^2 \quad (3)$$

算出された D_i を最小とする i が判別結果として得られ、 i が顔クラスであれば、入力ベクトルは顔と識別する。また、その時の D_i の最小値を判別スコア D とする。

線形判別分析では、入力ベクトルの次元数を M 、判別クラスの数 K とすると、判別空間の次元 N は $\min(K-1, M)$ となり、係数行列 A は、 $M \times N$ となる。また、判別スコア D_i の算出も K 回行う。入力ベクトルの次元 M は四方向面特徴の特徴量次元であるため固定と考えると、学習クラス数 K によって計算量が決定する。そのため、実時間性を重視すると、少ない判別クラス数 K で識別器を構成することが計算量削減の面で重要となる。

検出スコアは、学習サンプルの平均ベクトルに対する判別空間内での距離値であるため、学習サンプルに対する類似度と解釈できる。検出スコアが小さい場合には、その検出領域は顔あるいは頭部である信頼度が高いと考えられる。

3.1.5 探索

探索では、四方向面特徴を特徴量とし、線形判別分析による顔と頭部、非顔、非頭部の判別を用いて走査探索を行う。

人物を正面付近から撮影できる場合、図 7 に示すように顔と頭部の画像は画像解像度の違いとなる。そのため、同一人物の顔画像に対して顔と頭部を同時に検出することがあり、冗長な処理といえる。しかし、重複した検出結果を用いた評価による検出の信頼性向上や解像度の異なる顔と頭部を同時に探索することによって探索における解像度変換を粗くできる等の利点もある。また、顔と頭部を分類する本手法は、顔の向きを推定することもでき、パンチルトズームカメラを用いた拡大顔画像の効率的な収集にも有用と考えられる。

3.1.6 識別器の構成と学習

多くの顔検出に関する研究では、顔検出のための識別問題を顔と非顔の 2 クラス問題として扱っていることが多い。また、顔 25 クラス、非顔 25 クラスという合計 50 クラスで構成した線形識別器を用いて、良い結果が得られることも示されている。本研究では、線形識別器を用い、顔検出よりも識別境界が複雑になると思われる顔および頭部の検出を目的としている。そのため、判別空間内での説明変数が多い多クラス識別器の利用は、高精度な検出を実現するために有効と考えられる。しかし、前述したように、検出に要する計算量を考えた場合、少数のクラスによる識別器の構成が望ましく、両面を考慮した構成が重要である。

サベイランスシステムでは、人物とカメラの相対位置が一定でないため、様々なカメラ方向から撮影される顔や頭部の「見え」の変化に対する識別器の頑健性が重要となる。そこで、学習サンプルとして、多様な方向から人物の顔および頭部を撮影した HOIP 顔画像データベースを用いた。HOIP 顔画像データベースに関しては、「顔画像データベースの構築」に詳細が記述されている。本稿で顔と頭部の学習サンプルとして使用したカメラに対する水平方向の顔向きを表 1 に示す。表では正面を 0° とし、カメラに対して右を向いている顔を $+$ としている。サベイランスカメラが天井に設置されることを考慮し、垂直方向として、 0° および上方 15° から撮影した画像を学習サンプルに加えた。

識別器を構成するためには、顔と頭部以外に非顔、非頭部の学習サンプルを収集し、学習を行う必要がある。しかし、顔や頭部の画像集合といった概念と比較して、それら以外の画像集合である非顔と非頭部という概念は非常に大きな集合を成す概念である。また、非顔画像と非

表 1 学習に用いた顔向き方向（水平）

顔	$0^\circ \pm 15^\circ \pm 30^\circ \pm 45^\circ$
頭部	$0^\circ \pm 15^\circ, \pm 30^\circ, \pm 45^\circ \pm 60^\circ \pm 90^\circ$

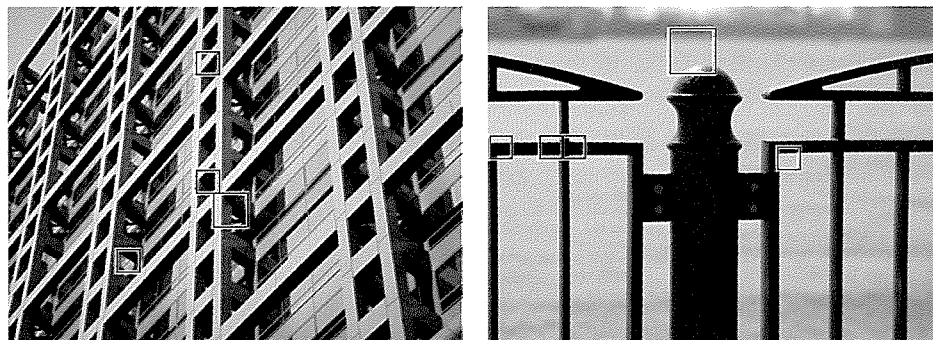


図 8 bootstrap 法に用いた画像の例

頭部画像のほとんどは、顔でも頭部でもない画像集合に含まれる。そのため、容易に分類することは出来ず、適切な学習サンプルの収集も困難である。そこで、bootstrap 法によって非顔画像および非頭部画像の学習サンプルの収集を行う。bootstrap 法は、初期学習後の識別器を用いて未知サンプルの識別を行い、誤識別を行ったサンプルを学習サンプルに追加し、再度学習を行う手法である。誤検出したサンプルを学習サンプルとして追加するため、識別境界付近の学習を効率的に行う手法の一つである学習サンプルの収集では、人物を含まない風景や物体の画像 2052 枚を用いた。bootstrap 法に用いた画像の例を図 8 に示す。図中の矩形は、収集された学習サンプルの領域である。

bootstrap 法による学習サンプルの収集を行うことにより、顔および頭部のクラス構成によって基本的な識別器のクラス構成が決定する。つまり、非顔・非頭部クラスは、検出対象とする顔・頭部クラスの対立クラスとして学習サンプルが収集され、同時にクラスの分類も行われる。顔および頭部のクラス分けとしては、対立クラスも各 1 クラスとして単純に 4 クラスとする構成や、顔および頭部を各々複数のクラスで構成し、各対立クラスの学習サンプルを収集する手法も考えられる。検出のための最適なクラス分け手法は今後の重要な課題としたい。

本研究では、計算量の抑制と顔画像の「見え」の違いに着目し、以下の手法で 6 クラスによる識別器の構成を試みた。まず顔クラスを右向き (+30°、+45°)、正面 (0°、±15°)、左向き (-30°、-45°) の 3 クラス、頭部クラスを 1 クラスとして検出対象とするクラスを 4 クラスで構成した。次に、各対立クラスを収集することによって合計 8 クラスの学習サンプルを収集した。最後に、最初に 3 クラスで構成した顔クラスを 1 クラスに統合し、識別器を 6 クラスに再構成した。この再構成では、形成した非顔クラスを維持したまま顔クラスの学習サンプルを統合した。判別空間内で分割されていた顔クラスが統合されることで、顔画像の変動を吸収する判別空間に広げることができるため、検出のロバスト性向上が期待できる。

3.1.7 後処理

後処理では、検出結果の評価を行い、最終的な判断を行う。また、同一人物に対して重なりを持つ場合には結果を統合する。本手法では、動画像を入力とし、顔と頭部を検出対象としているため、連続性や顔と頭部の重なり等を用いた評価が考えられるが、今回は連続性と検出スコアによる信頼度を用いた後処理を行った。

人物とカメラの距離が数 m 以上あり、10fps 程度のフレームレートであれば、画像内の人物位置は大きく変化しない。そのため、顔および頭部は前フレームの検出位置の近傍に検出される。そこで、連続性判定では、前フレームと現フレームで検出位置の距離値がしきい値よりも小さい場合に、顔および頭部が検出されたと判定する。また、前フレームで近傍に検出されていない場合には、判別スコアが十分に小さければ候補領域として記録し、次フレームでの連続性判定に利用できる。このように、検出スコアによる信頼度と連続性判定と組み合わせることによって、検出結果の信頼性を向上できると思われる。

3.2 LSI 化による高速化

ハードウェア (LSI) 化による高速化に関する検討は、実利用を考えた場合に必要である。また、LSI 化によってカメラ自体に人物検出機能を搭載することができれば、提案手法がより使いやすい形態となるため、応用範囲は広がると思われる。しかし、提案手法のアルゴリズムを LSI に実装する場合、これまで PC での実装を想定していたリソースは確保することが困難である。逆に、CPU 処理では、表現できない回路も記述可能である。そこで、LSI 化を意識したアルゴリズムの再構築が必要となる。本研究では、LSI 化の前段階として、提案手法を FPGA に実装した。組み込み型 FPGA と PC ボード型 FPGA の 2 種を検討した。前者の組み込み型 FPGA は、カメラへの実装を目的とし、全ハードウェアを FPGA に実装する。後者の PC ボード型 FPGA は、PC をシステムの構成要素に利用できる場合を想定し、人物検出部を高速化することで検出以外の処理にシステムリソースを割り当てたい場合に有効である。

3.2.1 LSI 化を想定したアルゴリズム

今回、ハードウェア化を意識した処理は、以下の処理である。

前処理 (動き検出、低解像度化)

- ・ 四方向面特徴抽出処理
- ・ 線形判別処理
- ・ 後処理

提案手法の実装では、PC上で実施されていたため、高速なCPU、豊富なメモリ、高速画像処理ライブラリなどを用いることができる。しかし、LSI化や組み込み用途を想定した場合、リソースの制約からアルゴリズムの工夫や処理の単純化が必要となる。ハードウェアロジックに対応できるように演算の整数化、扱う画像の低解像度化（画素数低減）、フィルタ回路の縮小等を行った。LSI化を意識し、アルゴリズムの変更点を以下に示す。

- (1) 四方向面特徴を抽出する際に、並列に（効率よく）計算できるように、計算方法を工夫した。
- (2) 四方向面画像のSDRAMへの格納に時間を要する（バスのトラフィックが高いため）。そのため、格納回数を減らした。
- (3) 四方向面特徴抽出において、ガウシアンフィルタ処理の計算コストが大きいため、従来5x5のマスクを20x20に適応した結果と同等なフィルタを作成した。
- (4) 線形判別分析を高速化するため、整数化を図った。
- (5) 組み込み型FPGAにおいては、特徴抽出用と判別の距離計算用の積和演算器を兼ねる構造にし、回路の小型化を実現した。

3.2.2 組み込み型FPGAの開発

組み込み型FPGAの開発にあたって、使用したシステムの構成（実験環境）に関して記述する。画像処理の殆どをLSI化することを想定して、以下の仕様をもつ組み込み型FPGAボードにアルゴリズムを実装することとした。ボードの基本仕様を以下に示す。

- ・ 名称：EPXA10-DEV-ボード
- ・ 10/100イーサネットPHY
- ・ UART2ポート
- ・ 16MバイトFLASHメモリ
- ・ 複数クロックソース
- ・ SDRAM DIMMソケット

本ボードに搭載されているFPGAの基本仕様を以下に示す。

- ・ 名称：Excalibur EPXA10 F 1020 C2
- ・ ゲート規模：38,400ロジックエレメント(100万FPGAゲート相当)
- ・ CPUのスペック：ARM922T, 166MHz

本ボードには、画像キャプチャや、結果の画像表示を行う機能がない。そこで、それらを入力出力するビデオインタフェース基板を新たに開発した。ビデオインタフェースボードの基本仕様を以下に示す

- ・ 映像入力インタフェース：S端子、NTSCコンポジット
- ・ 映像出力インタフェース：DVI出力、VGA
- ・ 信号：YCbCr (16bit)
- ・ CLOCK：27Mhz
- ・ 同期信号：水平、垂直、画素

3.2.3 PCボード型FPGAの開発

PCボード型FPGAの場合、PCのGUIや他のリソースが活用できるため、簡単に現実的なアプリケーションを構成することができる。通常のPCによる画像では、人物抽出、追跡等の処理が計算コストを削減、実時間処理に近い有用な応用の実現が困難である。そこで、PCボードに画像処理を実装したFPGAを載せることにより、高速化を図ることが有効と成る。本研究で用いたPCボードの仕様を以下に示す。

- ・ FPGA (Stratix) 搭載
- ・ コンフィグレーションROM EPC8QC100
- ・ PCI制御LSI PC19054 搭載

- ・ ローカルバスクロック 32bit/33MHz (可変)
- ・ プログラマブル PIO

これに NTSC キャプチャボード KIT1070ICAP をアドオンして動作させる。画像のキャプチャと処理を行う FPGA が直結することで、高速化が期待できる。通常の PC ボードでは、キャプチャと FPGA が別の基板になる場合が多い。その場合、キャプチャした画像を、一度 PCI バス経由で PC のメインメモリに移し、FPGA 処理のために再度、PCI バスを經由した後に処理しなければならない。このような画像データの送受信は、処理時間の増大に繋がる。本研究で利用した PC ボードでは、キャプチャした画像データを直接 FPGA に入力できるため、高速処理が期待できる。搭載された FPGA (Stratics1S25) の仕様を以下に示す。

- ・ 高速動作、最大 250MHz
- ・ 大規模 RAM 2Mbit 内蔵メモリ
- ・ 乗算器、加算／減算専用ブロック (DSP ブロック数 10)
- ・ 高速ディファレンシャル IO (最大 702)
- ・ ロジックエレメント 25,600

4 結果

実験では、室内や屋外にカメラを設置してデータ収集した実験によって評価を行った。また、提案手法を PC への実装を行った。

4.1 実験用データ

実験に使用した画像データは研究室や廊下、階段、エレベータ前、屋外等で複数の人物が撮影された約 5 秒間の動画を 26 シーン収集した。三脚を用いて約 2.2m の高さに DV カメラを設置した。取得した動画画像から 10fps でサンプリングを行い、640x480 の解像度でテスト用データを得た。

検出対象とする人物は、テスト用データの画像から手作業によって設定した。取得画像には、5 秒間の撮影時間の中にほとんど動いていない人物が含まれていたが、提案手法では動き領域によって探索領域を設定するため、そのような人物は検出対象外とした。

4.2 実験結果

検出結果の例を図 9 に示す。図中では、顔として検出された領域は破線、頭部として検出された領域は実線の矩形で示す。この結果から、人物がカメラ方向に顔を向けた映像に対する顔検出と同時に、横を向いた人物の頭部を検出できていることが分かる。照明条件が悪く、顔領域の濃淡変化が小さい場合にも頭部が検出されている (図 9(b), (i))。これは、顔の検出が困難な照明下において、輪郭形状を用いて頭部を検出したことを示す。検出されていない人物 (図 9(c), (d), (i)) は、検出対象よりも低解像度で撮影されていることが原因と考えられる。

本研究では、検出結果を人物検知率と検出信頼度によって評価した。人物検知率は、1 シーンに検出された人数を出現した人数で除した値である。検出信頼度は、全ての検出数に対する正検出数であり、検出の信頼性を示す。テスト用データに対する実験において、人物検知率は 83.6% (46/55) であり、検出信頼度は 84.6% (1048/1239) であった。検知できなかった延べ 9 人は、いずれも後頭部が撮影されていた。後頭部は、頭部の学習サンプルを $\pm 90^\circ$ の範囲内としたため学習サンプルには含まれていないが、検出できた例もあった (図 9(d), (e))。これは、多方向の頭部画像を 1 クラスで学習することにより、共通な特徴である楕円状の頭部輪郭形状が学習されたためと考えられる。

4.3 PC への実装

Intel 社の高速演算ライブラリである Integrated Performance Primitive (IPP) を使い、PC (Pentium4, 3GHz, Dual) 上に提案手法を実装した。動き検出を前処理として用いているため、シーンによってフレームレートは異なるが、入力画像の解像度を 640x480 とした場合にも約 10fps で動作している。3 フレーム間差分を用いているため、実際の入力画像に対して結果画像は 1 フレーム遅れるが、10fps で処理した場合、遅れは 0.1 秒となるので実環境における長時間



図 9 検出結果の例

の運用では、問題ないと考えている。

4.4 FPGA への実装

4.4.1 組み込み型 FPGA

組み込み型 FPGA に実装した結果、4fps 程度の処理速度であった。顔および頭部検出精度については、PC でソフトウェアにより処理する場合と比較してやや低下した。この精度低下は、ハードウェアリソースの制限からおこなった処理の簡略化によるものと考えられる。

処理速度は、4 fps であったが、これは、FPGA を 12MHz で駆動したためと、ARM のキャッシュをオフにしないと DPRAM を通してデータの受け渡しができないというハードウェアの制約による。この制約がなければ、15fps 程度の処理は達成できると考える。また、NTSC キャプチャボードと FPGA のやり取りに時間を要し、画像がバッファリングされ、約 1 秒遅れて表示されるという弊害も確認された。

処理速度を回路の観点から検討すると、ガウシアンフィルタ等のフィルタ演算処理の負荷がボトルネックとなる。この点を高速化するためには、FPGA の構造から更に踏み込んだカスタム仕様の回路も考えられる。

これらの結果から四方向面特徴と線形判別分析を用いた顔と頭部の検出手法をハードウェアにて動作させる可能性を示すことが出来たと判断できる。図 10 に接続した FPGA ボードとビデオインタフェースボードを示す。また、カメラとモニタを接続した状態を図 11 に示す。

4.4.2 PC ボード型 FPGA

本研究で用いた PC ボード型 FPGA は、ゲート規模、メモリも大きく、積和演算も内蔵して



図 10 FPGA ボードとインタフェースボード



図 11 カメラ、モニタと接続した状態



図 12 PC ボード型 FPGA

いるが、目標とした処理をすべて実装することは困難であった。そこで、線形判別分析等の後半の処理は実装せず、主に低解像度化、ガウシアンフィルタ、四方向面特徴抽出処理のみを実装することにした。

FPGA に実装した結果、13.5fps 程度の画像処理速度を達成することができた。この数値は、キャプチャボードの性能が 15fps であったこと等を考慮すると、実用的な速度であると思われる。図 12 に今回用いた PC ボード型 FPGA を示す。

フェーズ III

今後の取り組み

顔や頭部の検出による人物検出は、その後の処理を円滑に行うために、非常に重要な要素技術である。その応用例は、マンマシンインタフェースやアミューズメント、セキュリティ等が考えられる。本研究では、実装した PC ボード型 FPGA による顔検出機能を用いて、パンチルトズームカメラを用いた監視作業支援システムを試作した。本システムでは、カメラ制御を常にユーザが行う必要が無く、検出した人物を追跡するようにカメラを自動的に制御する。これによって、このシステムの実用化によって、半自動で不審人物を追跡できるため、監視員の負担が軽減できると思われる。しかしながら、プロトタイプを試作や実験等から、実利月のためには、検出精度や頑健性の向上が必要であるという知見が得られた。そのためには、学習手法の検討や判別器の構成法等の検討に今後取り組む必要があると思われる。