卒 業 研 究 論 文

題　　目

顔向きにロバストな顔検出・顔認識

～ヘッドマウントディスプレイへの応用を目指して～

第 3 グループ

指 導 教 員　　内海 ゆづ子 助教

平 成　23　年（　　2011　年）度 卒 業

（No.　　1080107016　　）　　加藤　祐也

大 阪 府 立 大 学 工 学 部 知 能 情 報 工 学 科

顔向きにロバストな顔検出・顔認識
～ヘッドマウントディスプレイへの応用を目指して～

第3グループ　加藤 祐也

1. はじめに

皆さんは，学会や学校，ビジネスの場などで，1度会った事のある人に，どこかでもう1度会った時に声を掛けられが，名前が出てこず困惑した経験がないだろうか．このように，たくさんの人を覚えるのは難しく，忘れると困ることがある．そのため，出会った人を覚えておくことの出来るシステムを開発すれば非常に便利であると考えられる．また近年，ヘッドマウントディスプレイが普及し始めている．ヘッドマウントディスプレイの利点としては，メガネ感覚で自然に装着可能であり，ハンズフリーで使えることが挙げられる．このヘッドマウントディスプレイとカメラを用いることにより，自然に会話をしながら表示された映像を見ることが可能である．

そこで本研究では，出会った人物を登録しておき，次にその人に出会った場合，人物を認識し，名前をヘッドマウントディスプレイに表示させる記憶補助システムを提案する．本論文では，上記のシステムを実現するための手法のうち，顔検出・顔追跡・顔認識について実験を行った．

2. 顔検出手法

本論文では，顔検出手法として，Haar-Like 特徴量を用いた手法 [1] を使用した．Haar-Like 特徴量を用いた手法では，まず顔画像と顔の含まれていない画像を何枚かずつ予め用意して，Haar-Like 特徴量を抽出する．学習アルゴリズムとして AdaBoost を用いて顔検出器を作成し，画像中から顔を検出する．しかし，Haar-Like 特徴量を用いた顔検出手法では，正面顔しか検出できない欠点がある．そこで，横顔にも対応させるため，本研究では，Mean-shift を用いた顔追跡法 [2] により，顔を検出した後，顔を追跡する．Mean-shift を用いた顔追跡法は，追跡対象のカラーヒストグラムから，ヒストグラム間の類似度関数に従い求められる各ピクセルの重みの分布に対して探索を行う．分布に対して探索を行うため，追跡対象の形状変化や部分的な隠蔽に対して頑健である．

3. 顔認識手法

本論文では，顔認識手法として，Affine Hull を用いた顔認識手法 [3] を使用した．この手法は，認識の際にクエリ・データベース共に，1 枚の画像を用いるのではなく，複数の画像を用いて，Affine Hull を作成し，Affine Hull 同士の距離を使用して，認識を行う手法である．1 人に対して，複数の画像をデータセットとして用いることで，顔向きや表情の変化に対して頑健に認識できる利点がある．

4. 実験と考察

本論文では，ヘッドマウントディスプレイへの応用を前提に，顔検出・顔追跡・顔認識の実験を行った．まず，顔検出の実験として，顔画像 3000 枚，顔の含まれていない画像 3000 枚を用いて，顔検出器を作成した．検出対象画像として，CAS-PEAL の無表情正面顔画像 1000 枚を用いた．検出結果は，再現率が 89.9 %，適合率が 100 %であった．

顔追跡を行った結果を，図 1 で示す．図 1 で示されるように，横顔でもうまく追跡できていた．しかし，追跡対象者が首の開いている服を着ていた場合，追跡結果が首の方に寄っていってしまうケースが見られた．これは，今回使用した Mean-shift を用いた顔追跡手法では，追跡対象のカラーヒストグラムを用いているため，共に肌色である顔と首を区別出来なかったためであると考えられる．

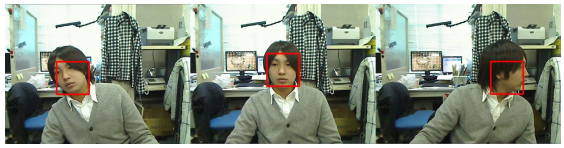

図 1: 顔追跡の結果

表 1: 顔認識の結果 [%]

	1	2	3	4	5	6	7	8	9	10
1	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0
5	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0
6	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0
7	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0
8	8.3	0.0	0.0	0.0	0.0	0.0	0.0	25.0	33.3	33.3
9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0
10	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0

次に，顔認識の実験では，10 人を web カメラで撮影し，顔検出・顔追跡を行い得られた，様々な顔向きの顔画像を，データベース・クエリ共にそれぞれ 200 枚用意して行った．人物ごとの認識率の結果を表 1 に示す．画像サイズがクエリ，データベース共に 15 × 15[pixel] の場合が最も認識率が高く，全体の認識率は 92.5 %であった．8 番のクエリセットでは，データベースには無表情が多いのに対し，クエリでは，横顔や表情の変化が多いため認識に失敗したと考えられる．

5. まとめと今後の課題

本論文では，ヘッドマウントディスプレイを用いた記憶補助システムを作成することを提案し，顔検出・顔追跡・顔認識の実験を行った．実験の結果，顔検出・顔追跡・顔認識共に，システムの実現に実用的な結果が得られた．今後の課題としては，さらに人数を増やして実験を行うことや，追跡・認識精度の向上が挙げられる．また，実際にヘッドマウントディスプレイにシステムを搭載させ，名刺などから文字認識により，自動的に名前を登録させるシステムを実装することも今後の課題である．

参考文献

[1] P. Viola, and M. Jones, “Robust real-time face detection”, IJCV, Vol. 57, no. 2, pp. 137–154, 2004.

[2] D. Comaniciu, V. Ramesh, and P. Meer, “Real-time tracking of non-rigid objects using mean shift”, in Proc. on CVPR, vol.II, pp. 142–149, 2000.

[3] H. Cevikalp and B. Triggs. “Face recognition based on image sets”, in Proc on CVPR, pp. 2567–2573, 2010.

目 次

第 1 章　緒論 **1**

第 2 章　顔検出・顔認識システム **5**

2.1　提案システムの概要 5

2.2　AdaBoost を用いた顔検出手法 5

2.2.1　Haar-like 特徴量 7

2.2.2　AdaBoost 8

2.3　Mean-shift を用いた顔追跡手法 11

2.3.1　Mean-shift 11

2.3.2　Tracking 12

2.4　顔認識法 14

2.4.1　Affine Hull を用いた顔認識手法 14

第 3 章　実験 **17**

3.1　AdaBoost による顔検出 17

3.2　顔追跡 18

3.3　顔認識 20

3.3.1　実験条件 20

3.3.2　画像サイズと認識率 20

3.3.3　クエリ枚数と認識率 21

3.3.4　結果 23

第 4 章　結論 **25**

謝辞 **27**

参考文献 **29**

図 目 次

1.1 ヘッドマウントディスプレイ 2

2.1 提案システムの処理の流れ 6
2.2 Haar-like 特徴量 7
2.3 矩形パターン 8
2.4 Integral Image 9
2.5 AdaBoost アルゴリズム 10
2.6 Tracking アルゴリズム 13

3.1 CAS-PEAL の画像例 18
3.2 検出された画像 18
3.3 追跡成功例 19
3.4 追跡失敗例 19
3.5 提案手法により切り出した画像例 20
3.6 画像サイズと認識率 21
3.7 画像サイズが 15×15 [pixel] の時のクエリ枚数と認識率 21
3.8 画像サイズが 20×20 [pixel] の時のクエリ枚数と認識率 22
3.9 認識に失敗したデータベースの画像例 22
3.10 認識に失敗したクエリの画像例 23

第1章　緒論

近年，人の生活・行為・体験などの思い出をデジタルデータとして記録するライフログが注目を集めており，人の記憶を記録し，表示する研究が行われている [12]．人の記憶には限界があり，日常の全てのことを覚えておくのは困難である．誰しも，一度会ったことのある人に，どこかでもう一度会った時に声を掛けられたが，名前が思い出せず困惑した経験があるだろう．人は日々新しい出会いを経験しており，頻繁に合わない人の名前は忘れてしまう．そこで，一度会ったことのある人に再会した際に，自動で名前を表示するサービスがあれば有用である．そこで，本研究では，出会った人を記録し，記録した人に再会した場合，リアルタイムで名前を表示する記憶補助システムを提案する．

記憶補助システムの結果を表示する装置として，スマートフォンやヘッドマウントディスプレイ (HMD) が考えられる．スマートフォンは広く普及しており，誰でも容易に使用できるといった利点があるが，スマートフォンを用いた場合，機器の操作が必要である問題がある．HMD は，メガネ感覚で自然に装着可能であり，ハンズフリーで使える利点がある．HMD と HMD に取り付けられたカメラを用いることで，自然に会話をしながら液晶中に表示された映像を見ることが可能である．そこで本研究では，システムの結果を表示する装置として，機器の操作を必要とせず自然に結果を見ることができる HMD を用いることを提案する．

記憶補助システムを作成する場合，各個人を認識し，人物を見分ける必要がある．人物を認識する方法として，音声，服装，顔などを認識する手法が考えられる．このうち，音声を用いた場合では，ある程度会話をしないと認識不可能である．服装を用いた場合では日によって服装が変わる可能性があるため，今回のシステムには不適当であるといえる．本研究では，短期的に変化しない同一特徴量が得られ，また機器への接触を必要とせず，離れていても相手の顔を見るだけで認識可能である顔認識を用いる．

提案システムで顔認識をする場合，会話中に認識対象の人物が横を向いたり表情を変えたりすることがある．そこで，認識対象人物の顔向きや表情にロバストな顔認識を行うこ

図 1.1: ヘッドマウントディスプレイ

とができる手法が望ましい．顔向きや表情にロバストな顔認識を行う手法として，1 人に対して複数枚の画像を用いて認識することがあげられる．提案システムでは，HMD に取り付けられたカメラで撮影された動画像を用いるため，同一人物の画像を複数枚得ることが可能である．そこで，HMD に取り付けられたカメラで撮影された動画中から顔を検出し，顔を追跡することで，同一人物の複数枚の画像を得る．また，人に出会った場合には，すぐに名前が表示されることが望ましい．そのためには，顔検出，顔追跡，顔認識，それぞれ高速な手法である必要性がある．

顔検出の手法として代表的なものとして，サポートベクターマシン (SVM) を用いた手法 [9] やテンプレートマッチングを用いた手法 [10]，AdaBoost を用いた手法 [1,2] などがあげられる．SVM を用いた手法は，学習サンプルより各データ点との距離が最大となる分離平面を求めることにより，識別器のパラメータを学習し検出を行う手法である．検出率が高い利点があるが，学習に膨大な時間がかかることや検出速度が遅いことが問題である．テンプレートマッチングを用いた手法は，予め検出対象のテンプレートを用意し，テンプレートを検出したい画像中で移動させながら，比較し相関の大きくなる場所を見つけていく手法である．処理が早い利点があるが，検出精度が低いことが問題であ

る．AdaBoost を用いた手法は，SVM も用いた手法に比べ高速で，テンプレートマッチングを用いた手法に比べ，精度が高い顔検出手法である．この手法では，画像中から特徴量を抽出し，学習を行う．そして，弱識別器の選択と組み合わせにより，１つの検出器を作成する．また，画像特徴量として高速で特徴抽出可能な Haar-like 特徴量 [3] を利用し，カスケード型の検出器を用いることにより，高速な顔検出を実現している．そこで本研究では，精度が高く，高速性のある AdaBoost を用いた顔検出手法を使用した．

顔追跡の手法として代表的なものとして，アピアランスモデルを用いた手法 [11] や Mean-shift を用いた手法 [4,5] などがあげられる．アピアランスモデルを用いた手法は，予め顔画像を学習することにより顔モデルを作成しておき，顔にモデルをフィッティングさせて顔を追跡する手法である．アピアランスモデルを用いた手法では，顔のパーツの位置がわかるといった利点がある．しかし，顔を追跡する際に，顔向きの変化に対応するためには，様々な顔向きで撮影した多数の顔画像で学習を行いモデルを作成する必要がある．また，顔にモデルをフィッティングさせる際に，処理時間がかかる問題がある．Mean-shift を用いた手法では，領域内のヒストグラムの分布に対して探索を行うため，学習が不要であり，形状変化や姿勢変化に強い利点がある．また，単純な計算を行うため，高速である．そこで本研究の顔追跡手法には，顔向きの変化にロバストで，高速な Mean-shift を用いた手法を利用した．

複数の顔画像を用いて顔認識を行う手法で代表的なものは，Eigenfaces[14] や Affine Hull を用いた手法 [6] がある，Eigenfaces は，顔のパーツの位置を合わせた顔画像を主成分分析することにより，固有顔を抽出し，認識を行う手法である．非常に高速である利点があるが，照明の変化に弱く，顔の正規化が必要である問題がある．Affine Hull を用いた顔認識手法は，複数の顔画像より，Affine Hull と呼ばれる凸包を作成し，Affine Hull 同士の距離を求めることによって，認識を行う手法である．Affine Hull を用いた顔認識手法は，データベース・クエリともに複数枚の画像を使用しても高速な認識が可能であり，正規化の必要もない．そこで本研究の顔認識手法には，Affine Hull を用いた手法を利用した．

本論文では，顔検出・顔追跡・顔認識それぞれの手法について実験を行い，提案システムの実現に向けて有効であるか性能評価を行った．実験の結果，顔検出においては，適合率 100%，顔認識においては認識率 92.5%と提案システムの実現に向け有効な手法であることが確認できた．以降 2 章では HMD に搭載するシステムの概要について述べる．3 章では実験結果を示し，4 章で結論を述べる．

第2章　顔検出・顔認識システム

本章では，まずシステムの概要を説明し，続いて，本研究に用いた顔検出手法，顔追跡手法，顔認識手法の概要について説明する．

2.1　提案システムの概要

本節では，提案システムの概要を説明する．提案システムの処理の流れを図 2.1 に示す．まず，出会った人を HMD に取り付けたカメラで撮影する．撮影された画像より顔領域を検出する．そして，顔追跡と顔特徴量の抽出を行い，人物を登録したデータベースから検索する．検索した結果，人物が見つかれば名前を表示する．見つからなければ，初めて会った人とし，名前情報と共に顔特徴量をデータベースに登録する．このシステムにより，出会った人を記録し，記録した人に再会した場合，リアルタイムで名前を表示する．

2.2　AdaBoost を用いた顔検出手法

AdaBoost を用いた顔検出は，Viola，Jones らによって開発され，Rainer らによって改良された [7]．Viola-Jones らの手法では，まず顔画像と顔の含まれていない画像を何枚かずつ予め用意して，Haar-like 特徴量を抽出し，AdaBoost を用いて強識別器を作成する．また，作成した強識別器を Attentinal Cascade と呼ばれる手法により連結する．Attentional Cascade で連結することにより，顔でない画像を早い段階で除去することが可能であり，非常に高速な顔検出を実現している．以下本手法で用いられている Haar-like 特徴量，AdaBoost について述べる．

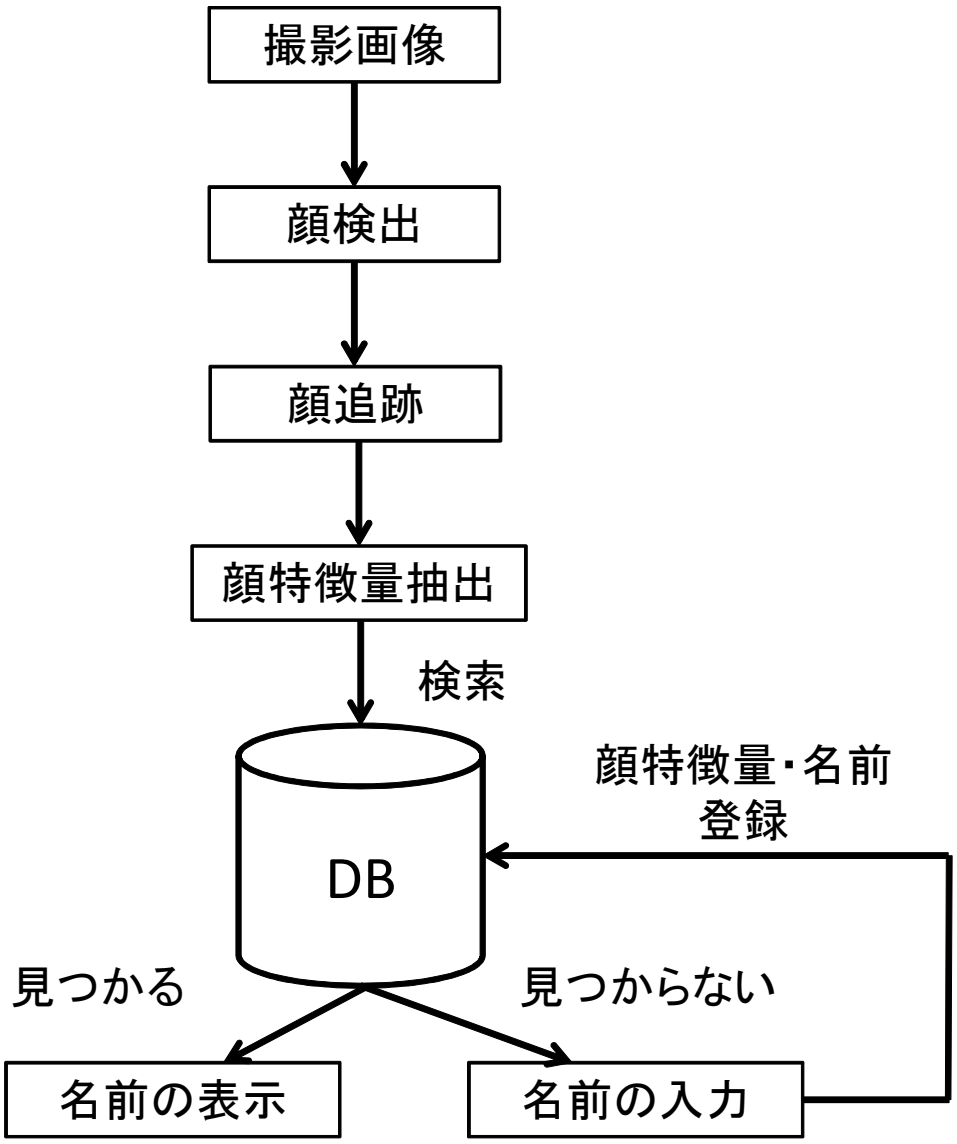

図 2.1: 提案システムの処理の流れ

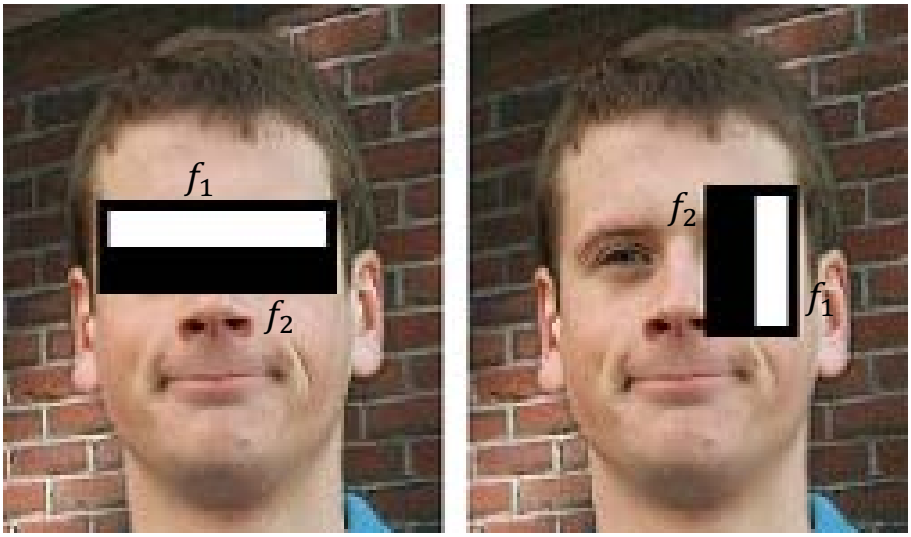

図 2.2: Haar-like 特徴量

2.2.1 Haar-like 特徴量

Haar-like 特徴量とは，図 2.2 のような白の矩形領域 f_1 と黒の矩形領域 f_2 の輝度値の差であり，式 (2.1) のように表すことができる．

$$S(f_1, f_2) = F(f_1) - F(f_2) \tag{2.1}$$

ここで $F(f)$ は，領域 f の輝度値の和を求める関数である．本手法では，矩形の位置と図 2.3 に示すような矩形パターンの組み合わせによって，約 12 万通りの Haar-like 特徴量が用いられる．この特徴パターン 1 つ 1 つで得られた特徴量を用いた識別関数が，以下に述べる AdaBoost の弱識別器となる．Haar-like 特徴量はある輝度値だけを見るのではなく，隣接する矩形内の輝度値の差を利用したものなので，ノイズや照明の変化に対してロバストである．

また，矩形特徴量を計算する際に，Integral Image と呼ばれる手法を使用することで非常に高速に特徴抽出が可能である．ある座標 (x, y) の Integral Image は式 (2.2) で定義できる．

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \tag{2.2}$$

ここで，$ii(x, y)$ はある座標 (x, y) における Integral Image であり，$i(x, y)$ は (x, y) における輝度値である．Integral Image を用いることで，例えば図 2.4 の領域 R 内の輝度値の和は，

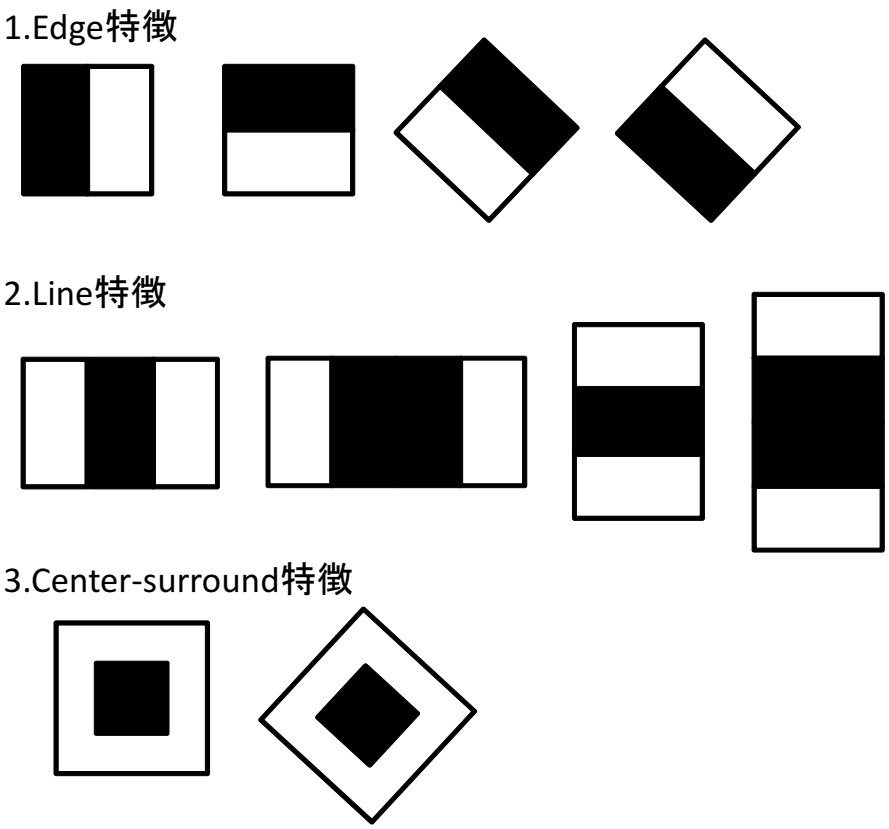

図 2.3: 矩形パターン

図 2.4 における A,B,C,D のそれぞれの点での Integral Image を $ii(A), ii(B), ii(C), ii(D)$ とすると，$ii(D) - ii(B) - ii(C) + ii(A)$ で求められる．これにより矩形内の輝度値の総和を，輝度値をその都度計算することなく高速に求めることができる．

2.2.2 AdaBoost

AdaBoost とは，1 つ 1 つはあまり識別性能の高くない弱識別器を，識別誤り率から求められる信頼度を用いて，複数個線形結合することにより，1 つの強識別器を作成する学習アルゴリズムである．AdaBoost では，容易に識別できる学習サンプルの重みを軽くし，識別が難しい学習サンプルの重みを重くすることで，学習が効率的に行われる．本論文で用いる，弱識別器 $h_j(x)$ は，特徴量 $f_j(x)$，しきい値 θ_j，不等号の向きを決めるパリティ

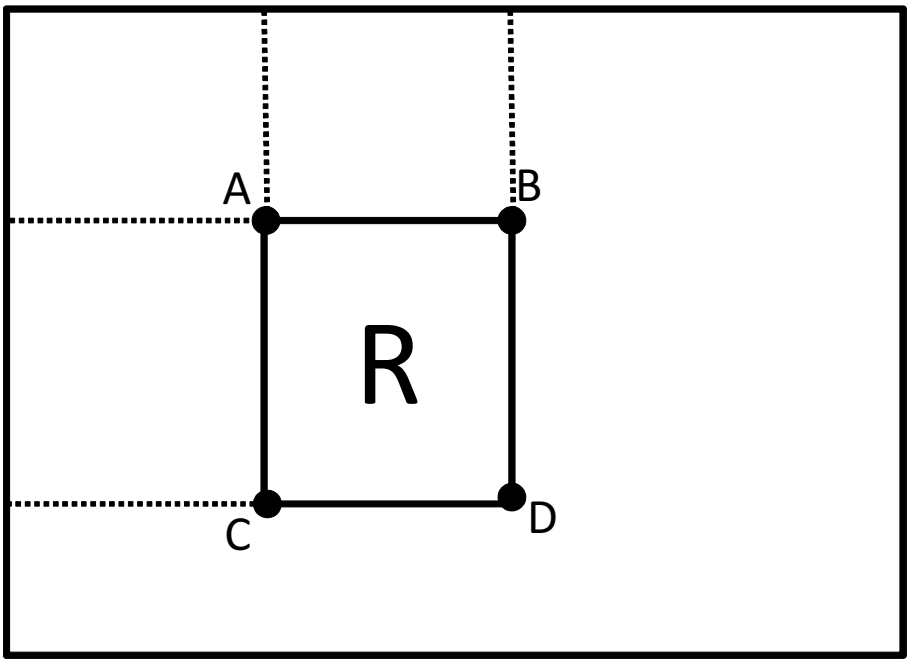

図 2.4: Integral Image

p_j を用いて以下のように定義する．

$$h_j(x) = \begin{cases} 1 & \text{if } p_j f_j(x) < p_j \theta_j \\ 0 & \text{otherwise} \end{cases}$$

今回用いた顔検出手法では，個々の Haar-like 特徴量が 1 つの弱識別器となる．AdaBoost のアルゴリズムの概要を図 2.5 に示す．最終的に得られる強識別器は T 個の弱識別器 $h_t(x)$ の線形結合となり，式 (2.3) のようになる．

$$H(x) = \begin{cases} 1 & \sum_{t=1}^{T} \alpha_t h_t(x) \geq \frac{1}{2}\alpha_t \\ 0 & \text{otherwise} \end{cases} \tag{2.3}$$

ここで $\alpha_t = \log\frac{1}{\beta_t}$ である．

AdaBoost を用いた顔検出手法では，強識別器を Attentional Cascade と呼ばれる方法で複数連結させることにより，非常に高速な検出が可能である．Attentional Cascade では，上流に行くほど false positive rate が高い強識別器 $H(x)$ を置く．上流の強識別器で顔でないと判断された場合その段階で処理を打ち切り識別を終了する．この処理により，顔

- $m + l = n$ 枚の学習サンプル $(x_1, y_1), ..., (x_n, y_n)$ を用意する．ここで顔画像 m 枚，非顔画像 l 枚，x_i は画像，y_i は教師信号である．顔画像には $y_i = 1$，非顔画像には $y_i = 0$ を与える．
- 重み w を初期化する．

$$w_i = \begin{cases} \frac{1}{2m} & (y_i = 1) \\ \frac{1}{2l} & (y_i = 0) \end{cases}$$

- For $t = 1, ..., T$:

  1. 重みを正規化する．

  $$w_{t,i} \leftarrow \frac{w_{t,i}}{\sum_n^{j=1} w_{t,j}}$$

  2. 識別器 h_j を，それぞれの特徴 j を用いて訓練させる．分類を誤った場合には，次式の評価関数で評価する．

  $$w_{t,\epsilon_j} = \sum_i |h_j(x_i) - y_i|$$

  3. ϵ_j が最も低い識別器 h_t を選択する．
  4. 重みを次式で更新する．

  $$w_{t+1,i} = w_{t,i}\beta_t^{1-e_i}$$

  ここで $\beta_t = \frac{\epsilon_t}{1-\epsilon_t}$ であり，もし x_i が正しく分類できれば，$e_i = 0$ とし，できなければ $e_i = 1$ とする．

図 2.5: AdaBoost アルゴリズム

でない確率が高い画像を早い段階で除外することができ，顔検出のリアルタイム処理が可能である．

2.3 Mean-shift を用いた顔追跡手法

Mean-shift Tracking は， Camaniciu らによって提唱されたカーネル関数を用いて物体を追跡する手法である．Mean-shift Tracking は，追跡対象のカラーヒストグラムから，ヒストグラム間の類似度関数に従い求められる各ピクセルの重みの分布に対して探索を行う．分布に対して探索を行うため，追跡対象の形状変化や部分的な隠蔽に対してロバストである．また局所的な重み分布を計算するため，高速な処理が可能である．以下，Mean-shift Tracking の概要を説明する．

2.3.1 Mean-shift

Mean-shift は，統計学においてよく知られるカーネル密度推定を用いたデータ解析手法である．Mean-shift は，1975 年に福永らによって提唱された最頻値探索問題の効率的でロバストな解法 [3] である．福永らの提案した Mean-shift の手順は以下のように表される．

- サンプル点の集合 $S = \{\boldsymbol{x_i} | i = 1, 2, ...n\}$ が与えられたとする．次に任意の点 μ を中心として半径 h の円領域 $T_h(\mu)$ を考え，この円領域内にあるサンプル点の平均値 $m(\mu)$ を計算する．
- 以下の (a)〜(b) の手続きを収束するまで繰り返す: $j = 0, 1, 2, ...$

  1. 平均値の計算をする．

$$m(\boldsymbol{x_j}) = \frac{1}{|T_n(\boldsymbol{x_j})|} \sum_{\boldsymbol{x} \in T_h(x_j)} x_i$$

  2. 現在位置を更新する．

$$\boldsymbol{x_{j+1}} \leftarrow m(\boldsymbol{x_j}), j \leftarrow j + 1$$

その後，この Mean-shift は 1995 年に Cheng[8] によって改良された．Cheng らが改良した Mean-shift ではカーネルを用いている．

一般的カーネルを $K(\boldsymbol{x}, \mu, h)$，非負値の重み関数を $w(\boldsymbol{x}) : S \to (0, \infty)$ で表すと，サンプル $\boldsymbol{x_i}$ の重み付き平均は式 (2.4) で定義できる．

$$m(\boldsymbol{x}) = \frac{\sum_{i=1}^{n} K(\boldsymbol{x_i}; \boldsymbol{x}, h) w(\boldsymbol{x_i}) \boldsymbol{x_i}}{\sum_{i=1}^{n} K(\boldsymbol{x_i}; \boldsymbol{x}, h) w(\boldsymbol{x_i})} \tag{2.4}$$

そして福永らの Mean-shift と同じ手続きを収束するまで繰り返す．

2.3.2 Tracking

本手法では，$k+1$ 番目のフレーム上で，前フレームの物体位置である $\boldsymbol{x_{k+1,0}}$ を Mean-shift の初期値 $\boldsymbol{y_0}$ と設定し，この Mean-shift での収束点 $\boldsymbol{y_j}$ をターゲット点 $\boldsymbol{x_{k+1}}$ の推定とみなすことを基本とする．両フレームにおける U 個の領域からなるカラーヒストグラム $P_u(\boldsymbol{y})$ と q_u をカーネル関数で平滑化したものを考えると，この 2 つのヒストグラムの類似度 $\rho\ (p_u(\boldsymbol{y}), q_u)$ は，以下の式で定義される Bhattacharyya 係数で表される．

$$\rho\ (p_u(\boldsymbol{y}), q_u) = \sum_{u=1}^{U} \sqrt{p_u(\boldsymbol{y}), q_u} \tag{2.5}$$

式 (2.5) にテーラー展開による近似を施すと，式 (2.6) となる．

$$\rho\ (p_u(\boldsymbol{y}), q_u) \quad \frac{1}{2} \sum_{u=1}^{U} \sqrt{p_u(\boldsymbol{y_0}), q_u} + \frac{C_h}{2} \sum_{i=1}^{n} w_i k \left(\left\| \frac{\boldsymbol{y} - \boldsymbol{x_i}}{h} \right\|^2 \right) \tag{2.6}$$

ここで，$\boldsymbol{y_0}, \boldsymbol{x_i}, y$ は $k+1$ フレームにおける初期地点，その近傍のターゲット候補点，及び Mean-shift 変数である．また，$k(x)$ は幅 h のカーネルプロファイルで，C_h は定数項である．式 (2.6) の右辺第 1 項は変数 $\boldsymbol{y}$ に依存しないので，$\rho\ (p_u(\boldsymbol{y}), q_u)$ の最大値化には，右辺の第 2 項のみを考慮する．この重み付きカーネル和関数を，式 (2.4) により最適化する．ここで，正値の重み関数 w_i は以下のように定義される．

$$w_i = \sum_{u=1}^{U} \delta\ (b(\boldsymbol{x_i}) - u) \sqrt{\frac{q_u}{p_u(\boldsymbol{y_0})}} \tag{2.7}$$

ここで，δ はデルタ関数，$b(\boldsymbol{x})$ は点 x での輝度値を表す．
Tracking アルゴリズムを図 2.6 に示す．

1. 初期値 $\boldsymbol{y_{k,0}}$ を前物体位置 $\boldsymbol{y_{k-1,0}}$ で設定.
2. $\boldsymbol{y_0}$ において $P_u(\boldsymbol{y_0}), q_u$, ρ $(p_u(\boldsymbol{y_0}), q_u)$ の計算.
3. $\boldsymbol{y_0}$ の近傍候補各点で w_i を (2.7) 式より計算.
4. 式 (2.4) で Mean-shift bmy_1 を計算.
5. $\boldsymbol{y_1}$ において ρ $(p_u(\boldsymbol{y_1}), q_u)$ を計算.
6. ρ $(p_u(\boldsymbol{y_1}), q_u) <$ ρ $(p_u(\boldsymbol{y_0}), q_u)$ の場合 $\boldsymbol{y_1} \leftarrow \frac{1}{2}(\boldsymbol{y_0} + \boldsymbol{y_1})$ とする.
7. $\boldsymbol{y_0} \leftarrow \boldsymbol{y_1}$ とし $\boldsymbol{y}$ の収束まで 4〜6 を繰り返す.

図 2.6: Tracking アルゴリズム

2.4 顔認識法

本節では，本研究に用いた手法である Affine Hull を用いた顔認識手法の概要について説明する．本手法は，Cevikalp らによって提案され [6]，顔認識の際に，1 枚の画像を用いるのではなく，複数の画像を用いることで認識を行う手法である．複数の画像を用いることで，顔向きや表情の変化に対してロバストである．

2.4.1 Affine Hull を用いた顔認識手法

Affine Hull を用いた顔認識手法では，Affine Hull と呼ばれる Affine 部分空間同士の距離を求めることで認識を行う手法である．Affine Hull とは，Affine 空間内での凸包のことである．Affine Hull 同士の距離を求める手法の概要を説明する．顔画像サンプルの集合を $\mathbf{x}_{ci} \in \Re^d$ とする．ここで，C 人の顔画像がある場合各個人の画像セットのインデックスを $c = 1, ..., C$ とし，各画像セット c のそれぞれの画像のインデックスを $i = 1, ...n_c$ とする．これらの画像セットを含む最小の Affine 部分空間は，式 (2.8) となる．

$$H_c^{\text{aff}} = \left\{ \mathbf{x} = \sum_{k=1}^{n_c} \alpha_{ck}\mathbf{x}_{ck} \middle| \sum_{k=1}^{n_c} \alpha_{ck} = 1 \right\}, c = 1, ..., C \tag{2.8}$$

式 (2.8) で表される凸包を Affine Hull とする．

この Affine Hull をパラメータで表現するため，適当な基準点をとり μ_c とする．μ_c を用いて，式 (2.8) を書き換えると，式 (2.9) のように表すことができる．

$$H_c^{\text{aff}} = \left\{ \mathbf{x} = \mu_c + \mathbf{U}_c\mathbf{v}_c | \mathbf{v}_c \in \Re^l \right\} \tag{2.9}$$

ここで，$\mathbf{U}_c$ は Affine 部分空間の正規直交基底，$\mathbf{v}_c$ は部分空間の次元を削減した時のパラメータベクトルである．$\mathbf{U}_c$ は $[\mathbf{x}_{c1} - \mu_c, ..., \mathbf{x}_{cn_c} - \mu_c]$ に特異値分解 (SVD) を適用することで得られる．

2 つの交差しない Affine Hull$\{\mathbf{U}_i\mathbf{v}_i + \mu_i\}$，$\{\mathbf{U}_j\mathbf{v}_j + \mu_j\}$ が与えられた時，2 つの Affine Hull の最も近い点同士の距離は式 (2.10) から求められる．

$$\min_{\mathbf{v}_i, \mathbf{v}_j} \|(\mathbf{U}_i\mathbf{v}_i + \mu_i) - (\mathbf{U}_j\mathbf{v}_j + \mu_j)\|^2 \tag{2.10}$$

ここで，$\mathbf{U} \equiv (\mathbf{U}_i - \mathbf{U}_j)$ と定義し，$\mathbf{v} \equiv \begin{pmatrix} \mathbf{v}_i \\ \mathbf{v}_j \end{pmatrix}$ と定義すると式 (2.10) は以下のように表

せる．

$$\min_{\mathbf{v}} \|(\mathbf{U}\mathbf{v}) - (\mu_j - \mu_i)\|^2 \tag{2.11}$$

式 (2.11) を解くと，$\mathbf{v} = (\mathbf{U}^{\mathrm{T}}\mathbf{U})^{-1}\mathbf{U}^{\mathrm{T}}(\mu_j - \mu_i)$ となり，2 つの Affine Hull 同士の距離は以下の式で書き換え可能である．

$$D(H_i^{\mathrm{aff}}, H_j^{\mathrm{aff}}) = \|(\mathbf{I} - \mathbf{P})(\mu_i - \mu_j)\| \tag{2.12}$$

ここで，$\mathbf{P} = \mathbf{U}(\mathbf{U}^{\mathrm{T}}\mathbf{U})^{-1}\mathbf{U}^{\mathrm{T}}$ であり，2 つの部分空間の直交射影である．また，$\mathbf{I} - \mathbf{P}$ は，$\mathbf{P}$ に対応する直交補空間への射影子である．

式 (2.12) より，2 つの Affine Hull 同士の距離を求められる．クエリ画像セットの Affine Hull を H_q^{aff} とすると，認識結果 R は式 (2.13) で表される．

$$R = \arg\min_{c} D(H_c^{\mathrm{aff}}, H_q^{\mathrm{aff}}) \tag{2.13}$$

第3章　実験

本章では，提案システムを実現するための手法について評価実験を行った．3.1 節では，AdaBoost を用いた顔検出，3.2 節では，Mean-shift を用いた顔追跡，3.3 節では，Affine Hull を用いた顔認識について評価を行った．

3.1　AdaBoost による顔検出

web 上より顔画像 3000 枚，顔が含まれていない画像 3000 枚を取得し，AdaBoost により顔検出器を作成した．検出対象画像は，CAS-PEAL[13] の無表情正面顔画像 1000 枚を用いた．検出対象画像の画像サイズは，360×480 [pixel] でグレースケールのものを用いた．検出対象画像の一例を図 3.1 に示す．実験に用いた計算機は，Opteron 2.2 GHz，メモリ 256 GB である．

顔検出を行った結果を表 3.1 に示し，検出された画像の一例を図 3.2 に示す．処理時間は 1 枚の画像あたり，382 [ms] であった．適合率は 100%であり，再現率は 89.9%と 10%程度検出できない場合があった．提案システムでは，入力が動画であることを前提にしている．そのため，あるフレームで顔が検出できなくても，次のフレームで検出できる機会がある．再現率が 89.9%でも提案システムの実現には十分であるといえる．また，検出時間に関しては，1 枚あたり 382 [ms] であり，動画中の毎フレームで顔検出を行う場合，リアルタイム性があるとはいえない．しかし，顔検出を行うのは，追跡を行う前の初期の数フレームのみであり，その後は顔追跡を行うので，382 [ms] でも提案システムの実現には問題ないといえる．

図 3.1: CAS-PEAL の画像例

表 3.1: 顔検出の実験結果

検出数	899
顔	899
背景	0
再現率	89.9%
適合率	100%

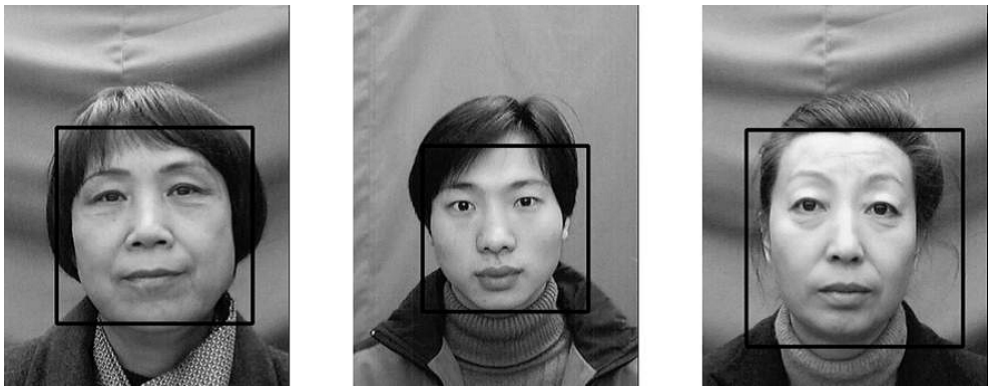

図 3.2: 検出された画像

3.2 顔追跡

Mean-shift を用いた顔追跡手法により，web カメラにより撮影した映像を使用して，顔追跡を行った．使用した計算機は，core i5 3.3GHz，メモリ 8GB，使用した web カメラは Logicool Qcam Pro 9000 (15 [fps]) である．顔追跡の結果を図 3.3，図 3.4 に示す．図 3.3 のから横顔も追跡できていることがわかる．しかし，図 3.4 に示すように，追跡対象者が

図 3.3: 追跡成功例

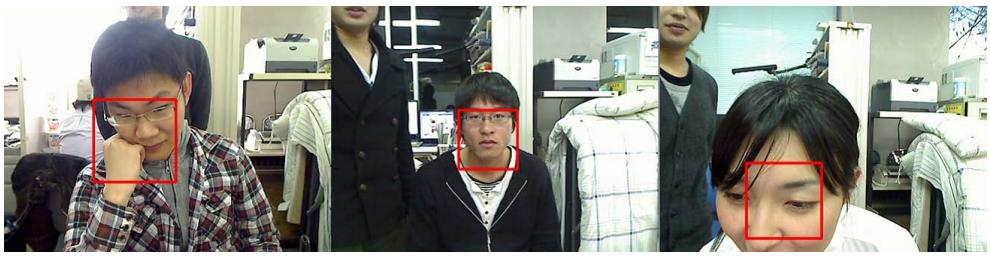

図 3.4: 追跡失敗例

首の開いている服を着ていた場合や手を近づけた場合，追跡結果が首や腕の方に寄った場合が見られた．この原因として，Mean-shift を用いた顔追跡手法では，追跡対象のカラーヒストグラムを用いていることがあげられる．したがって，追跡の初期領域に顔を設定すると肌色が多く含まれているので，Mean-shift を求めると肌色が多い方向に移動ベクトルが求められる．よって，共に肌色である首や腕を顔を誤って追跡したことが考えられる．1 フレームあたりの処理時間は 56 [ms] であった．今回用いた動画像のフレームレートである 15 [fps] においては，リアルタイムで追跡可能であるといえる．

また，顔追跡精度を評価するために，web カメラにより撮影し，顔追跡を行った画像 30 枚を用いて追跡精度を評価する実験を行った．追跡精度は，式 (3.1) で定義する．

$$(\text{追跡精度}) = \frac{(\text{顔領域のの画素数の和})}{(\text{追跡矩形内の顔領域の画素数の和})} \tag{3.1}$$

顔領域がすべて追跡矩形内に入っている場合，追跡精度は 100% となる．なお，顔領域は手作業で切り出しを行い，画素数の和を求めた．30 枚の画像の追跡精度の平均は，82.8%であった．約 18% 顔領域を追跡できていない原因としては，顔は楕円型であるが，今回追跡に用いた領域が正方形であったため，顔領域が全て入り切らなかったことがあげられる．また，上記で述べた，共に肌色である首や腕を顔と誤って追跡したことも原因である．

図 3.5: 提案手法により切り出した画像例

3.3 顔認識

本節では，Affine Hull を用いた顔認識手法の評価実験を行った．用いる画像サイズ，クエリ枚数が認識率に与える影響について実験を行った．

3.3.1 実験条件

顔画像クエリ・データベースともに，web カメラを用いて撮影し，2.1 節で示した提案手法により，顔検出・顔追跡を行い切り出した．使用した計算機，web カメラは 3.2 節で用いたものと同じものである．切り出した画像を図 3.5 に示す．また実験では，画像は全てグレースケールのものを使用した．顔向き，表情の変化のある 10 人の顔画像を 1 人あたり 200 枚使用した．今回，クエリの人物はデータベース中のいずれかの人物とし，該当しないものはないとした．

3.3.2 画像サイズと認識率

画像サイズによる認識率の変化を調べるため，クエリ，データベースの画像サイズを変化させて実験を行った．画像のリサイズには，バイキュービック法を利用した．画像サイズを変化させた時の認識率を図 3.6 に示す．実験の結果より，15×15 [pixel] の画像を用いた時が最も認識率が高いことがわかった．15×15 [pixel] が最も認識率が高かった理由としては，15×15 [pixel] よりも画像のサイズが小さいと認識に必要な情報が十分得られず，認識率が低下したためである．また，画像のサイズが大きくなっても，次元数が大きくなったことによりロバスト性が失われ，認識率は低下した．

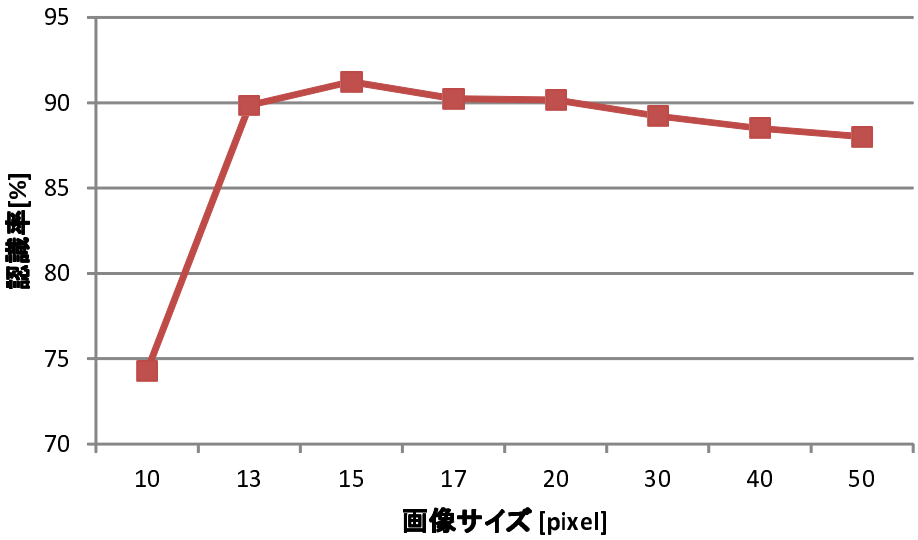

図 3.6: 画像サイズと認識率

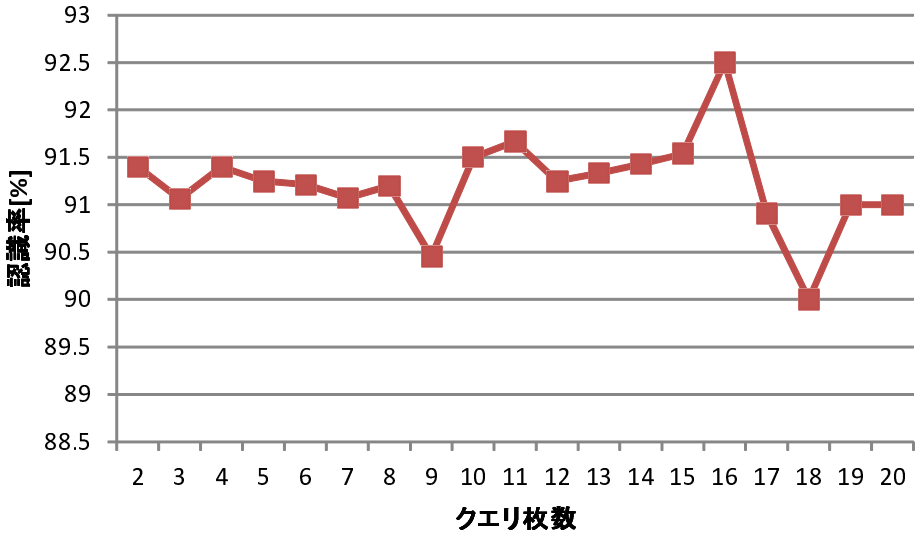

図 3.7: 画像サイズが 15 × 15 [pixel] の時のクエリ枚数と認識率

3.3.3 クエリ枚数と認識率

クエリの枚数による認識率の変化を調べるため，データベースの枚数は 200 枚で固定し，クエリの枚数を 2,3,...,20 枚まで変化させて実験を行った．画像サイズが 15 × 15 [pixel],

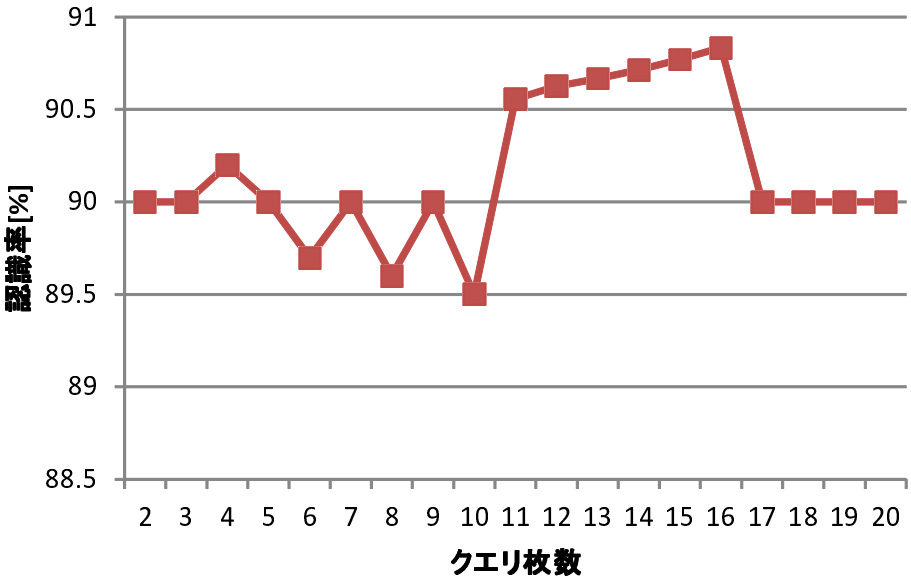

図 3.8: 画像サイズが 20 × 20 [pixel] の時のクエリ枚数と認識率

図 3.9: 認識に失敗したデータベースの画像例

20 × 20 [pixel] でのクエリ枚数を変化させた時の認識率をそれぞれ図 3.7，図 3.8 に示す．実験の結果，画像サイズが 15 × 15 [pixel]，20 × 20 [pixel] のいずれの場合においても，クエリ枚数が 16 枚の時が最も認識率が高いことがわかった．16 枚が最も認識率が高かった理由については，画像に含まれる表情や顔向きの種類にもよると考えられるため，さらに画像や人の数を増やして検証を行う必要がある．

表 3.2: 各個人ごとの認識率 [%]

		認識結果									
		1	2	3	4	5	6	7	8	9	10
クエリ	1	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	2	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	3	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	4	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0
	5	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0
	6	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0
	7	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0
	8	8.3	0.0	0.0	0.0	0.0	0.0	0.0	25.0	33.3	33.3
	9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0
	10	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0

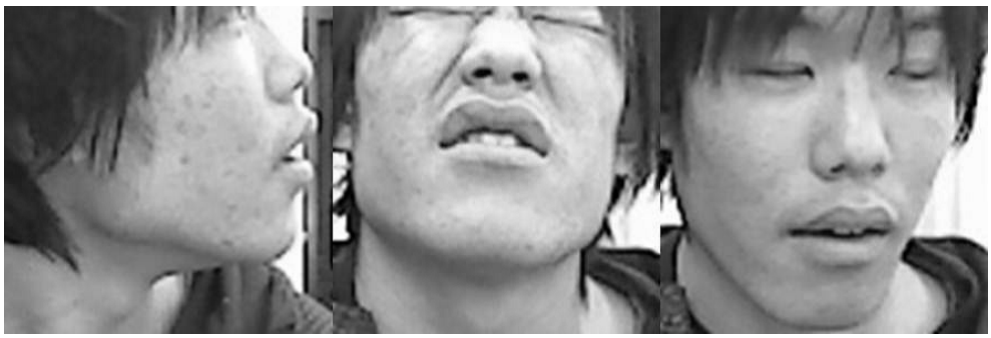

図 3.10: 認識に失敗したクエリの画像例

3.3.4 結果

3.3.2 節，3.3.3 節の結果から，画像サイズを 15×15 [pixel]，クエリ枚数を 16 枚に固定し認識の評価を行った．各人物ごとの認識率の結果を表 3.2 に示す．表 3.2 の 1 から 10 までの番号は，それぞれの人物の ID を表し，縦の番号がクエリの ID，横の番号がそれぞれのクエリの認識結果を表す．また 1 クエリセットあたりの処理時間は，412 [ms] であった．表 3.2 から，8 番のクエリセット以外は 100%認識できていることがわかる．8 番のクエリセットでは，図 3.9 で示すようにデータベースには無表情の画像が多いのに対し，クエリ

では，図 3.10 のように横顔や表情の変化の大きいものが非常に多いため認識に失敗した．また，1 つのクエリセット 16 枚の顔画像を得るためには，今回用いた 15 [fps] の場合であると約 1 秒必要である．処理時間は，412 [ms] であるので，次のクエリセットを撮影している間に認識可能であるので，処理時間は提案システムの実現に十分な速度であるといえる．

第4章　結論

本論文では，HMD を用いて，出会った人を記録しておき，その人に再会した場合，人物を認識し，リアルタイムで名前を表示する記憶補助システムを提案した．また上記システムを実現するためには，顔向きにロバストな顔検出，顔追跡，顔認識の必要性があるため，それぞれの手法について実験を行い提案システムへの有効性を示した．AdaBoost を用いた顔検出手法で顔を検出した結果適合率は 100%であった．Mean-shift による顔追跡手法では横顔も追跡可能であることが示された．また，顔認識においても，顔向き・表情の変化があっても，１枚の画像を用いるのではなく，クエリ・データベースともに複数の画像を用いることで，92.5%と高い認識率が得られた．処理時間に関しては，顔検出においては 382 [ms]，顔追跡においては 56 [ms]，顔認識においては 412 [ms] と提案システムの実現には十分な処理時間であることが示された．

今後の課題として，以下のことがあげられる．1つ目は，さらなる認識率の向上である．この課題を実現するためには，さらに人数を増やして実験を行い，その影響について評価することや，顔追跡において，首や手まで顔として追跡されたことの改善があげられる．次に，顔認識において，データベースに登録されていない人は，該当しないとすることである．さらに，HMD とカメラを組み合わせたシステムを実際に作成し，実用性を評価することもあげられる．また現在は，顔認識は PC で行い HMD は表示装置として使用することを考えているが，HMD 自体にプログラムを搭載し HMD だけで動作可能であるか検証し，評価することも今後の課題である．最後に，ビジネスの場では，名刺を交換することが想定されるので，名刺から文字認識をするなどの名前を自動的に取得するシステムを実装することもあげられる．

謝辞

本研究の進行にあたり，日頃より御指摘，御助言を頂いた黄瀬浩一教授，ならびに，岩村雅一准教授に深く感謝致します．また，研究内容について直接のご指導を頂いたほか，実験，論文，発表においても多くの御配慮，御助言をして下さった，内海ゆづ子助教に深い謝意を表します．また，前川敬介氏をはじめ，様々な御支援，御助力を下さった知能メディア処理研究室の諸氏に感謝致します．

2012 年 3 月 9 日

参考文献

[1] Y. Freund, and R. E. Schapire, “A Decision-Theoretic Generalization of on-Line Learning and an Application to Boosting”, Proceedings of the Second European Conference on Computational Learning Theory, pp. 23–37, 1995.

[2] P. Viola, and M. Jones, “Robust real-time face detection”, International Journal of Computer Vision, vol. 57, no. 2, pp. 137–154, 2004.

[3] P. Viola, and M. Jones, “Rapid Object Detection using a Boosted Cascade of Simple Features”, Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 511–518, 2001.

[4] K. Fukunaga, and L. D. Hostetler, “The estimation of the gradient of a density function, with applications in pattern recognition”, IEEE Transaction on Information Theory, vol. 21, pp. 32–40, 1975.

[5] D. Comaniciu, V. Ramesh, and P. Meer, “Real-time tracking of non-rigid objects using mean shift”, Proceedings IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 142–149, 2000.

[6] H. Cevikalp and B. Triggs. “Face recognition based on image sets”, Proceedings on Computer Vision and Pattern Recognition, pp. 2567–2573, 2010.

[7] R. Lienhart and J. Maydt, “An Extended Set of Haar-like Features for Rapid Object Detection”, Proceedings of 2002 International Conference on Image Processing , vol. 1, pp. 900–903, Sep. 2002.

[8] Y. Cheng, “Mean Shift, Mode Seeking, and Clustering”, IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 17, pp. 790–799, 1995.

[9] E. Osuna, R. Freund, and F. Girosi. “Training support vector machines: an application to face detection”, Proceedings of Computer Vision and Pattern Recognition, pp. 130–136, 1997.

[10] J. Miao, B. C. Yin, K. Q. Wang, et al. “A hierarchical multiscale and multiangle system for human face detection in a complex background using gravity-center template”, Pattern Recognition, vol. 32, no. 7, pp. 1237–1248, 1999.

[11] F. Dornaika, and F. Davoine, “On Appearance Based Face and Facial Action Tracking”, IEEE Transaction on Circuits And Systems For Video Technology, vol. 16, pp. 1107–1124, 2006.

[12] M. Lamming, and M. Flynn, “Forget-me-not: Intimate computing in support of human memory”, Proceedings of Future Personal Information Environment Development, pp. 125–128, 1994.

[13] W. Gao, B. Cao, S. Shan, et al. “The CAS-PEAL Larege-Scale Chinese Face Database and Evaluation Protocols”, Technical Report, Joint Research and Development Laboratory, CAS, 2004.

[14] M. Turk and A. Pentland, “Eigenfaces for recognition”, Journal of Cognitive Neuroscience, vol. 3, no. 3, 71–86, 1991.