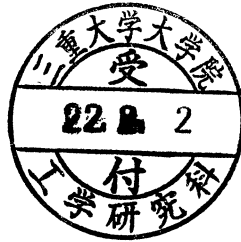


修士論文

チェインコード化 Merlin-Farber 法 を用いた正面顔正中線の自動抽出



平成 21 年度修了
三重大学大学院工学研究科
博士前期課程 情報工学専攻

岡本 大地

はじめに

バイオメトリクス認証に応用できる自動の顔認識・顔認証の実現には，入力画像から顔領域を切り出す顔検出と，切り出された領域が誰の顔であることを認識する顔認識の両方が必要である．顔検出では，人物の顔が1つあるいは複数含まれている画像が入力として与えられる．顔検出で切り出される領域は，衣服や毛髪を含まずに，額とあご，両頬までを含む顔面を正しく捉えている必要がある．

顔画像の認識を高精度に行うためには，入力された画像から対象人物の顔のみを含む領域を正しく切り出す必要がある．切り出す顔領域の決定に，両目，鼻，口などの顔部品の情報を利用することは有用であるが膨大な計算コストがかかってしまう．画像からの顔部品抽出において，処理の高精度化，高速化に利用するために顔正中線（中心線）を抽出する手法として，仲尾らは正中線自動抽出の高速化アルゴリズムを提案した．顔部品の抽出処理の前に，顔の回転角度と中心位置を検出した正中線を利用して補正すれば，顔部品抽出に必要な画像の探索範囲を削減出来るため，計算コストの削減が期待できる．

本研究では，Merlin-Farber ハフ変換 (Merlin-Farber Hough Transformation : MFHT) によって正中線を検出する際に，入力画像をチェーンコード化し，そのコードを参照することで投票空間への投票回数を制限する新しいアルゴリズムを提案する．これにより，正中線抽出に不必要な投票を減らすことが期待できる．このアルゴリズムは，仲尾らの手法に比べより高速，高精度に正抽線を抽出できる．

本研究では正中線自動検出手法の有効性を，FERET データベースから抽出した 2409 画像を用いた実験により評価した．実験の結果，正中線抽出成功率が正中線候補数が 1 本の場合 89.9% から 95.1% に向上した．また，候補数が 10 本の場合は 97.7% から 99.3% に，候補数が 20 本の場合は 98.9% から 99.8% に向上した．これにより，提案手法は従来手法よりも安定して正中線を抽出できることが大規模な画像データベース上での実験により実証された．今後の課題として，(1) 正中線検出性能のさらなる改善，(2) 正中線抽出処理のさらなる高速化，(3) 顔部品抽出，顔検出への応用が挙げられる．

目次

はじめに	i
第 1 章 緒言	1
1.1 研究の背景	1
1.2 本研究の目的	2
1.3 関連する先行研究	2
1.4 論文の構成	3
第 2 章 顔画像に関する従来研究	4
2.1 顔検出と顔認識	4
2.2 顔検出の代表的な手法	5
2.2.1 事前知識に基づく顔検出	6
2.2.2 特徴に基づく顔検出	6
2.2.3 テンプレートマッチングに基づく顔検出	7
2.2.4 見え方に基づく顔検出	7
第 3 章 正中線自動抽出	8
3.1 顔の正中線とは	8
3.2 手法の概要	9
3.3 前処理	10
3.4 Merlin-Farber ハフ変換 (Merlin-Farber Hough Transformation : MFHT)	11
3.5 チェインコード化 MFHT	12
3.5.1 チェインコードの概要	12
3.5.2 エッジ画像のチェインコード化	13
3.5.3 チェインコード化 MFHT による投票	14
3.6 投票空間の次元削減手法	15
3.7 正中線の決定	17
3.8 まとめ	17

第 4 章	評価実験	19
4.1	FERET データベース	19
4.2	正中線検出の正しさの指標	19
4.3	正中線検出におけるパラメータ決定のための予備実験	20
4.4	性能評価実験	21
第 5 章	結言	28
5.1	考察	28
5.2	今後の課題	28
付録 A	プログラムと実験用データ	30
A.1	プログラムソースリスト	30
A.2	画像ファイルリスト	30
付録 B	修士論文発表会で使用したパワーポイント	32
謝辞		33
	参考文献	34

第 1 章

緒言

1.1 研究の背景

バイオメトリクス認証に応用できる全自動の顔認識・顔認証を実現するためには，入力画像から正確に顔領域を切り出す顔検出と，切り出された顔を認識する顔認識の両方が必要である．顔検出とは，人物の顔が一つあるいは複数個含まれている画像から顔領域のみを切り出し，非顔領域と分割する処理である．このとき，顔の大きさは未知である場合がほとんどであり，例えば， 512×768 画素の大きさの入力画像において， 128×128 画素の領域に顔が存在する場合や，ほぼ画像全体の 500×700 画素を顔が占める場合もある．顔として切り出される領域は，衣服や髪を含まず，額とあご，両頬を含む顔面を正しくとらえていることが求められる．顔領域を正確に切り出すことで，後に続く顔認証の精度が向上することが期待できる．

顔検出には目，鼻，口などの目印となる顔部品の位置情報が主に用いられる．そのためこれらの顔部品の位置を正確にとらえることが出来れば，顔の検出と顔領域の抽出は容易かつ正確に行うことができる．

しかし，顔部品の抽出には高い計算コストを要する．顔認識において，最も単純な単一人物の正面顔の画像でさえ，各顔部品の位置，顔の大きさ，顔の回転角度など多くのパラメータの推定を行う必要がある．

この計算コストを削減するために仲尾ら [1] は顔の正中線を抽出し，検出した正中線を顔の傾きと位置を正規化するのに利用した．これにより，顔部品抽出の複雑さを削減できることが分かっている．

1.2 本研究の目的

正中線抽出に要する時間は、従来の相関関係を用いた手法に比べ、Merlin-Farber ハフ変換 (Merlin-FarberHough Transformation:MFHT) を用いた手法は短くなっている。しかし、3次元ハフ空間へ投票を行う処理は無視できる量ではない。そこで、仲尾らは投票空間のパラメータを制限した高速化の手法を提案した [1]。この手法では、正確性やある程度のノイズを容認することで計算時間を削減した。もし、新たなハフ投票空間への投票の削減手法を提案することができれば、正確性を犠牲にすることなく計算コストの削減を行うことができる。

正中線抽出の精度の向上はその後の顔部品抽出の精度に大きな影響をもたらすため、本研究では顔の正中線抽出の精度の向上、さらなる計算コストの削減を目的とする。

本研究では、仲尾らが提案した MFHT を用いた正中線抽出を応用した手法を提案する。提案手法の主な概念は、ハフ投票空間への余分な投票を禁止することで、計算コストの削減、正確性の向上を行うことである。そして、ハフ投票空間への余分な投票を禁止するために、正中線抽出にチェインコードで表現された2値エッジ画像を用いる。チェインコードは隣接する画素対を8方向の系列で表現したものであり、このチェインコードを用いることで余分な投票を禁止することが期待できる。詳しい処理については第3章にて説明する。

本手法は、1個の正面顔が含まれる画像から顔の正中線を検出する。画像中の顔は、しばしば傾いているため、検出手法は顔の傾きを考慮する必要がある。本論文では、顔正中線の自動検出技術と、その検出精度を、FERET データベース [2] に含まれる顔画像を用いた実験によって示す。

1.3 関連する先行研究

顔の正中線を自動検出する手法として、X.Chen らが提案した顔正中線自動検出手法 [3] がある。彼らの手法では、任意の直線の両側の輝度値の差 (Gray Level Differences : GLD) に基づいて、対称性の尺度として定義された Y 値を最大にするような直線を顔の線対称軸として検出する。しかし、この手法は証明環境の変化に影響を受けやすい。また、Y 値が最大となる軸を求めるためには、回転や位置など推定すべきパラメータの組み合わせの全てで Y 値の計算処理が必要となり膨大な計算コストがかかってしまう。

これ以外に、Hiremath ら [4] によって提案された手法がある。この手法では、顔の線対称軸を含めた lines-of-Separability(LS) 顔モデルによって、顔が記述される。この LS 顔モデルを得るためには、顔の線対称軸を得る前に、正面顔画像から両目を抽出する必要が

ある。この手法は本手法とは逆に先に顔部品を抽出してから顔の線対称軸を求める手法である。

他の手法として、Active Appearance Model[5]のような、顔の形や見え方のモデルを適用する手法がある。これらの手法は、入力画像に対して、事前に学習されたモデルやテンプレートを適用するものである。本手法では、正中線の検出にこのような事前学習は必要としない。

1.4 論文の構成

本論文ではまず、第2章にて顔画像に関する研究について紹介する。続いて、第3章で、顔の正中線を検出するための提案手法について述べる。さらに、第4章では、従来手法と提案手法の正中線検出を定量的に評価した実験結果について述べ、第5章にて考察と今後の課題について述べる。

第 2 章

顔画像に関する従来研究

顔画像の自動認識は、建物の入退出管理、銀行 ATM などでの個人識別、ロボットと人間との間のインターフェースなど、様々な応用分野があり、盛んに研究されている。この研究分野の最終的な目標は、実環境下における安定した顔画像の自動認識であるが、これは計算機にとっていまだに難しい課題である。

ここでは、従来の顔画像処理に関する研究について本研究に関連する顔検出を取り扱うものを中心に紹介し、この分野のこれまでの研究動向について概説する。

本章の内容よりも詳しいサーベイ論文 [6][7] が発表されているので、それらも参照されたい。

2.1 顔検出と顔認識

顔を含んだ画像、または画像列は、例えば視覚に基づいた知的マンマシーンインターフェースなどにおいて基本となる情報である。顔認識や顔の追跡、表情の認識など多数の顔画像処理に関する研究が報告されているが、これらのほとんどは入力画像中から適切に顔領域が切り出されていることを仮定としている。

ここでは、顔が蔵書利の分野で特に重要なテーマである顔検出と顔認識について概略を説明する。

入力画像に複数の顔が含まれていた場合、それらの顔を発見し適切な大きさに画像から切り出す処理を顔検出と呼ぶ。顔検出の最終的な目標は、画像中の顔のサイズや向き、形状の変動にかかわらず正しく顔を検出することである。

顔認識とは、入力画像中の顔が、事前に取得されているデータベースに含まれるどの人物の顔かを判定する処理である。この処理では、入力画像とデータベース内の全ての顔が比較される。多くの顔認識手法では、額からあご、左右の頬までの顔面画像を認識対象としている。これは、頭髮や衣服の変化による画像変動が認識結果に与える悪影響を低減す

るためである。つまり、顔認識を目的とした顔検出では、余分な領域を含まない顔面領域のみを正しく切り出す必要がある。顔認識の手法は大きく次の3つに分類することができる。

- 画像全体を用いてマッチングを行う手法
- 特徴に基づいてマッチングを行う手法
- 画像全体と特徴の両方を用いる手法

画像全体を用いたマッチングの代表的な手法としては、主成分分析に基づく固有値顔を用いた手法 [8][9] などがある。特徴に基づいたマッチングの代表的な手法としては、局所特徴の形状に基づく手法 [10], Elastic Bunch Graph Matching(EBGM)[11] などがある。画像全体と特徴の両方を用いた代表的な手法としては、全体の固有値と局所的な固有値を用いた手法 [12] などがある。

顔検出手法の多くは、顔の検出に重点を置いているため、顔の境界線を正確かつ厳密に決定することが困難である。この問題を解決するために、顔部品 (両目, 鼻, 口)などを抽出し, それらの位置情報をもとに統計的モデルなどによって顔の境界線を決定するアプローチが提案されている。

2.2 顔検出の代表的な手法

顔が含まれた画像全体から顔のみを含む領域を検出し切り出す処理を, ここでは顔検出と呼ぶ。この処理を顔の同定 (Face Localization) として顔検出 (Face Detection) と分ける場合もあるが, 本研究ではこれらの区別は行わない。

顔の検出は顔認識における重要な前処理である。ここでは顔検出技術に関するサーベイ論文である文献 [6] で紹介されている研究のうち, 本研究に関連の深いものを紹介する。

自由な環境条件下で撮影された画像から顔を検出する問題は, 以下に示すような困難さを含んでいる。

1. **画像中の顔の姿勢**: 被写体である顔とカメラとの相対的な位置関係の変化により, 画像中の顔の見え方は大きく変動する。位置関係によっては顔の特徴の退部文が隠れる場合もある。
2. **顔構成の変動**: 例えば, 眼鏡や髭などの存在や変化により, 顔の見え方は容易に変化する。
3. **表情の変動**: 表情が変動することにより, 顔の見え方が変化する。
4. **顔の隠れ**: 顔が他の物体で隠れる場合がある。
5. **画像の向き (カメラの回転)**: カメラが光学軸上で回転することにより顔が回転しているように見える場合がある。

6. 画質： 撮影場所の光源環境やカメラの特性により画像の画質が異なる。

これらの問題は、撮影環境を制御することにより軽減できる場合もある。

本研究では、単一のカメラで撮影された顔画像を入力画像としている。また、複数台のカメラで撮影した顔画像、3次元デジタイザで撮影した顔画像を用いた顔検出の研究もさかんに行われている。単純な単一のカメラで撮影された画像を入力画像とすることで、複雑な入力画像を用いた研究の指標となることが期待される。

2.2.1 事前知識に基づく顔検出

このアプローチでは、顔に関する事前知識に基づく規則によって顔検出手法が定義される。例えば、画像中にそれぞれ対称な位置関係で両目がありその中心線上に鼻と口が存在する、のように顔部品とそれらの位置関係を説明することの容易性が、この方法が採用された理由である。主な手順としては、なんらかの処理により顔の候補を複数抽出し、それぞれの候補を顔ルールに従って真に顔であるかどうかを判定する。これにより偽陽性 (False positive) を減少できる。

これらの手法の問題は、多くの場合に対して一般的に有効なルールの作成が極めて困難であることである。具体的には、特定の撮影環境で撮影された画像に対して有効なルールが開発されたとしても、そのルールが想定しない範囲の変動が画像中の顔に存在した場合、顔の検出は失敗しやすくなる。

このアプローチの代表的な手法として Yang らによる Multiresolution rule-based method[13] がある。

2.2.2 特徴に基づく顔検出

事前知識に基づく手法に対して、顔検出に利用できるような顔の不変な特徴、例えば、画像から顔部品や顔の特有な肌色領域を取り出すことで顔を検出する研究が多く行われてきた。これらの特徴が得られれば、統計的モデルによりそれらの位置関係などを検証することで、対象領域の顔らしさを評価することが出来る。

これらの手法の問題点は、このような顔の特徴の画像での見え方が光源環境に容易に悪影響を受けること、顔の特徴の境界が顔そのもののそれよりも弱く表現されることなどである。また、顔の特徴の検出が、顔そのものを検出する問題と同じ困難さをもっていることに注意する必要がある。このアプローチの代表的な手法としては、顔部品を用いる手法 [14] [15][16]、顔テクスチャを解析する手法 [17]、肌の色を利用する手法 [3][4] などがある。

2.2.3 テンプレートマッチングに基づく顔検出

事前に手動で定義，またはパラメータ表現された標準的な顔テンプレートと入力画像とのマッチングにより顔を検出する手法である．入力画像が与えられた時には，例えば，顔の輪郭，両目，鼻，口などでそれぞれ独立に標準テンプレートとの相関を計算し，相関の値に基づいて顔の有無を決定する．この手法の利点は実装が容易であることである．しかし，この手法は顔の大きさ，姿勢，形状の変動に対して十分に対応できない場合がある．これらの変動に対処するために多重分解能，多重スケールテンプレートの導入や，可変形状テンプレートなどが提案されている．

このアプローチの代表的な手法としては，形状テンプレート [18]，Active Shape Model(ASM)[20]，Active Appearance Model(AAM)[5]，Flexible Appearance Model(Lanitis)[19] などがある．

2.2.4 見え方に基づく顔検出

見え方に基づく顔検出手法の，テンプレートマッチング手法との大きな違いは，テンプレートが事前に定義されるのではなく，画像から収集された実例の学習によって得られる点である．基本的にこのアプローチは，顔画像と非顔画像との特徴を的確に記述するために，統計的解析と機械学習の手法を用いている．学習された特徴は分布モデルや判別関数の形式で記述され，顔検出に利用される．計算効率と検出性能の向上のために次元削減が行われる．

このアプローチの代表的な手法としては，主成分分析を利用して顔の見かけ特徴を抽出する Eigenface[21]，ニューラルネットワークを用いる手法 [22]，隠れマルコフモデルを用いる手法 [23]，Dynamic link architecture[24] などがある．

第 3 章

正中線自動抽出

3.1 顔の正中線とは

本研究における顔の正中線は、目と目を結ぶ線分（眼間線分）の垂直二等分線と定義される。図 3.1 は入力画像の一例で、角度 θ だけ傾いた頭部を含んでいる。このときの顔の正中線は頭部と同じ傾き θ を持った直線として検出される。画像上から直線を検出することは、その直線が通る一個の点とその直線の傾きを求めることに等しい。正中線の場合、眼間線分上の一点と顔の傾きとを検出することとみなすことができる。図 3.1 の直線 A は、点 $C = (C_x, C_y)$ と角度 θ とを用いて、以下の式で表される。

$$\frac{x - c_x}{\sin \theta} = \frac{y - c_y}{\cos \theta} \quad (3.1)$$

本研究では、正中線を軸として互いに線対称となる 2 点から、2 つのパラメータ c と θ を決定する。点 $p = (p_x, p_y)$ と点 $q = (q_x, q_y)$ とを、互いに線対称とする線対称軸上の点 c は $c = \frac{p+q}{2}$ と表すことができる。そして、角度 θ は、線分 $(q - p)$ に直行する角度として得られる。したがって、式 (3.1) を、 p, q を用いて、以下の式で表すことができる。

$$\frac{x - \left(\frac{p_x + q_x}{2}\right)}{\sin \theta} = \frac{y - \left(\frac{p_y + q_y}{2}\right)}{\cos \theta} \quad (3.2)$$

$$\theta = \tan^{-1}\left(\frac{q_y - p_y}{q_x - p_x}\right) \quad (3.3)$$

正中線検出における課題は、図 3.1 において点 p , 点 q として示されている、お互いに左右対称となる 2 点を検出することである。

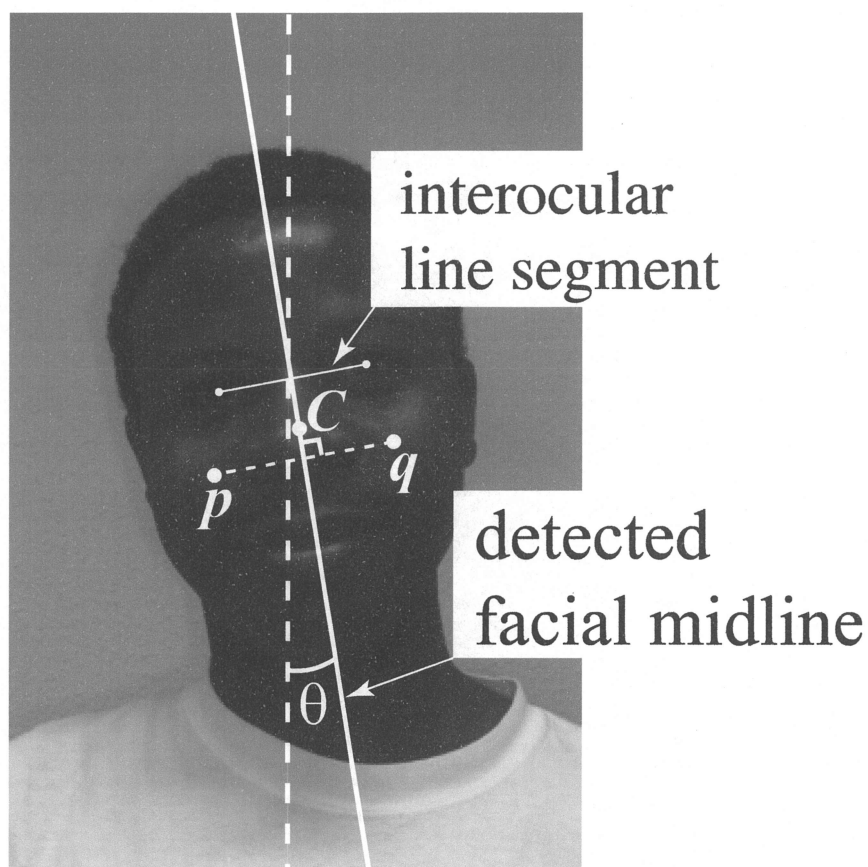


図 3.1: 顔正中線の例

3.2 手法の概要

手法の処理の流れを主に図 3.2 に示す。

本手法の入力画像は、図 3.2(a) に示すような、 512×768 画素のグレイスケール画像で、1 個の正面顔を含む。画像中の顔の大きさ、傾きは任意とする。

まず、エッジ抽出、2 値化、ノイズ除去、境界追跡を含む前処理を行う。本手法は、正面顔は対極的に左右対称であるという仮定に基づいている。しかし、顔に左右のいずれか片方から照明が当たる場合には、顔の見え方の対称性がそこなわれてしまう。このような照明による影響を軽減するために前処理を行う必要がある。前処理を終えた結果画像は MFHT を適用するために十分なエッジ成分を含んでいる。前処理の結果画像の例を図 3.2(b) に示す。

次に、前処理によって得られた輪郭画像に対してチェーンコード化を行う。チェーンコード化によって、輪郭画素のエッジの方向を知ることが出来る。

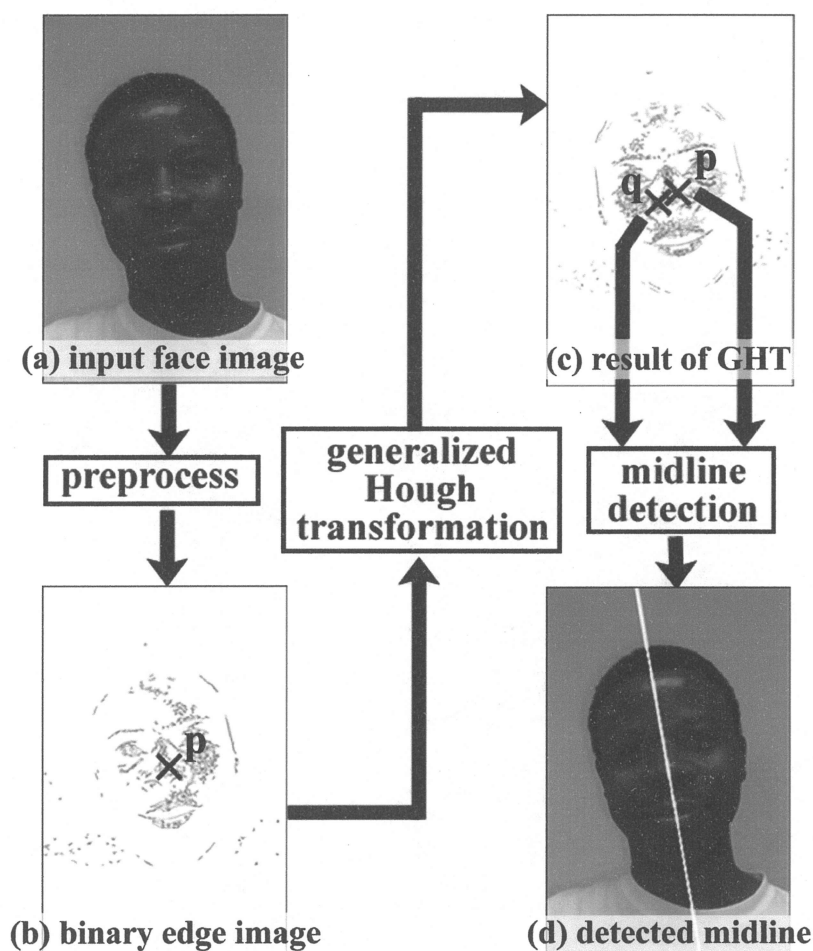


図 3.2: 提案手法における処理

続いて、鏡面テンプレートとのマッチングを MFHT によって行う。MFHT では、適切な実行結果を得るために、妥当な参照点を決める必要がある。参照点の例を、図 3.2(b) の点 p に示す。MFHT は、参照点と左右対称関係にある点を検出する。この検出された点を、本研究では抽出点と呼ぶ。抽出点の例を図 3.2(c) の点 q に示す。

最後に、互いに線対称関係にある点 p 、点 q の座標を用いて、式 3.2 より、顔の正中線を決定する。

それぞれの処理について、次節以降で詳しく述べる。

3.3 前処理

本手法における前処理では、入力画像から 2 値エッジ画像を作成する。以降の処理で実行する MFHT のアルゴリズムは 2 値画像のみを対象とするため、適切な処理結果を得る

ためにも 2 値画像の作成は極めて重要な処理である。例えば、2 値画像に多くのノイズが存在する場合、MFHT の計算コストが増大するだけでなく、結果にも悪影響を与えてしまう。一方、エッジ成分の強度が小さすぎる場合、MFHT の信頼度が低下してしまう。

前処理では、まず最初に、原画像 $I = I(p)$ の画素 p に対して以下の Sobel オペレータ [25] を適用してエッジ強調を行う。

$$\begin{aligned}
 I'(p) &= \sqrt{G_x(p)^2 + G_y(p)^2} \\
 G_x(p) &= I(p_x + 1, p_y - 1) + 2 \times I(p_x + 1, p_y) + I(p_x + 1, p_y + 1) \\
 &\quad - \{I(p_x - 1, p_y - 1) + 2 \times I(p_x - 1, p_y) + I(p_x - 1, p_y + 1)\} \\
 G_y(p) &= I(p_x - 1, p_y + 1) + 2 \times I(p_x, p_y + 1) + I(p_x + 1, p_y + 1) \\
 &\quad - \{I(p_x - 1, p_y - 1) + 2 \times I(p_x, p_y - 1) + I(p_x + 1, p_y - 1)\}
 \end{aligned} \tag{3.4}$$

次にエッジ画像を p -タイル法 [26] で 2 値化する。 p -タイル法では、しきい値 T より大きな画素値を持つ画素の数が、全画素数の $p\%$ となるように T を設定する。従来手法では、 T より小さい輝度値を持つ画素は白、輝度値が T より大きければ黒として 2 値化する。 Sobel オペレータは、原画像のノイズに基づくエッジも強調してしまうため、適切な 2 値化を行ったとしても、結果の 2 値画像にはいくらかのノイズが含まれる。このノイズを削減するために、連結成分数 L_{min} 画素以下の成分を削除する。このとき、連結成分の長さは 8-近傍境界追跡 [27] によって取得する。境界追跡により、各エッジ成分は輪郭線として表される。

3.4 Merlin-Farber ハフ変換 (Merlin-Farber Hough Transformation : MFHT)

従来手法では、輪郭線から顔の正中線を検出するのに Merlin-Farber ハフ変換 (MFHT) を用いる。 MFHT は 2 値の入力画像から、任意の図形を検出する図形検出手法で、一般化 Hough 変換の一種である。一般化 Hough 変換は、図形の欠落やノイズに対して頑健な手法として知られている。 MFHT は 2 値画像を対象とした図形検出のテンプレートマッチングの高速化アルゴリズムとして用いられている。一般に入力画像の図形は、大きさ、位置、回転などの幅広い自由度を伴っているため、テンプレートに対して様々な変換を行いながら、入力画像との適応度を調べる必要がある。この計算コストは膨大である。計算時間を削減するために、MFHT では図形の持つ変動の自由度と等しい次元数のパラメータ空間への投票を行う。

本手法における MFHT の目的は、参照点と左右対称関係にある抽出点を検出することである。人物の顔は左右対称であるという仮定から、エッジ画像も左右対称であると仮定する。前処理で得られた 2 値エッジ画像の鏡面画像をテンプレートとして用いる。つまり

MFHT を用いて、2 値エッジ画像から、それとよく似た図形であると仮定した鏡面画像を検出する。MFHT が図形を検出すれば、左右対称関係にある 2 点は容易に決定できる。

本手法における MFHT の流れを以下に示す。

- (1) **参照点の決定:** MFHT では、参照点を決定する必要がある。参照点の決め方は任意であるが、この参照点は後に続く MFHT の処理に大きな影響を及ぼすため、非常に重要であると言える。佐藤ら [25] は、参照点としてエッジ画素の重心を用いれば、信頼度の高い MFHT の処理結果が得られることを明らかにした。本手法ではこのことより、参照点として 2 値エッジ画像の黒画素 (エッジ画素) の重心を用いる。重心は以下の式によって得られる。

$$p = \frac{\sum_{j=0}^N e_j}{N} = \left(\frac{\sum_{j=0}^N e_j x}{N}, \frac{\sum_{j=0}^N e_j y}{N} \right) \quad (3.5)$$

ここで、 $e_j = (e_{jx}, e_{jy})$ 、 N はそれぞれエッジ画素の位置とエッジ画素の総数を表す。参照点の例を、図 3.3 の点 p に示す。

- (2) **テンプレート画像の生成:** 2 値エッジ画像を垂直軸に対して反転させた鏡面画像をテンプレート画像として用いる。2 値エッジ画像が垂直軸に対して左右対称であるならば、原画像とテンプレートは抽出点において、適切に重なり合うと考えられる。
- (3) **パラメータ空間への投票:** 本手法における MFHT のパラメータ空間は、 q_x 、 q_y 、回転角度 θ の 3 次元からなる。これらは、テンプレートの変換の自由度に相当する。図 3.3 は、MFHT の投票結果は図示したものである。テンプレートの点対称図形の抽出点を、2 値エッジ画像上の全てのエッジ画素に重ね合わせながら走査する。走査処理と同時に、現在のテンプレートのエッジ画素の位置と回転角度に対応するパラメータ空間に、投票を累積していく。
- (4) **抽出点の検出:** パラメータ空間の中から最大の投票度数を持つ点を、テンプレートの位置と回転角度として検出される。

3.5 チェインコード化 MFHT

3.5.1 チェインコードの概要

画像列を座標系列で表す代わりに、隣接する画素対の方向の系列で表したものがチェインコードである。画素列を a_0, a_1, \dots, a_n とすると、 a_0 から a_1 へ向かう方向を図 3.4 に従って符号化する。同様に a_n まで繰り返すことにより、 n 個の符号列からなるチェインコードが求められる。符号化の処理は、終点と開始点が一致する、もしくは隣接する画素が存在しない場合に終了する。

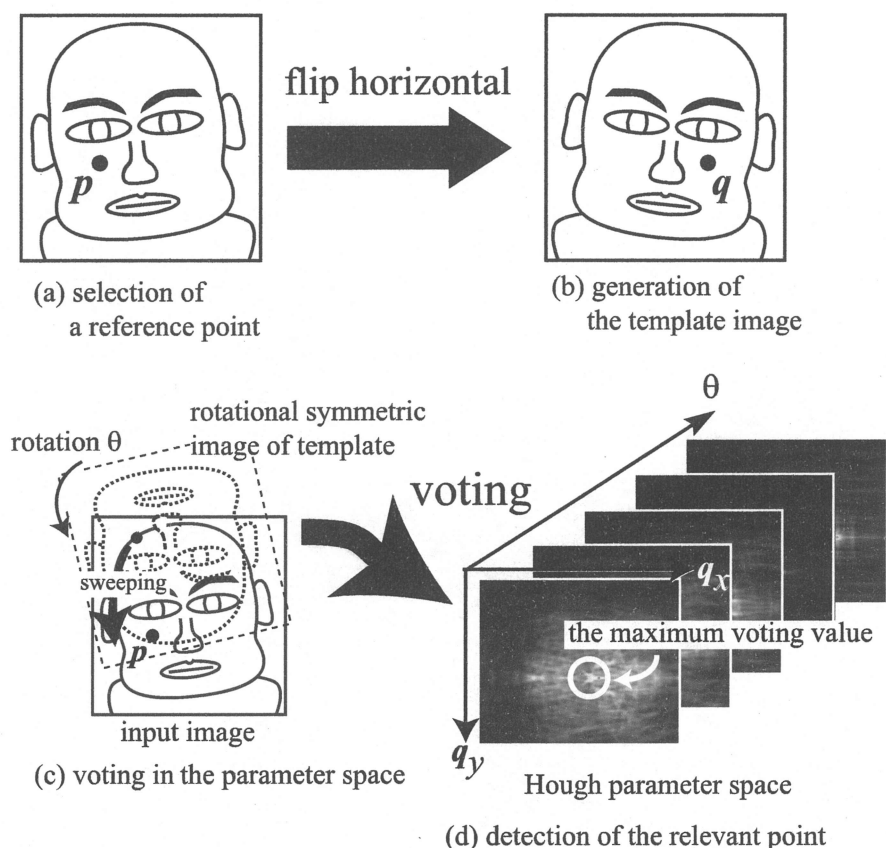


図 3.3: 提案手法における処理

3.5.2 エッジ画像のチェインコード化

円の Hough 変換を行うとき、注目点における輪郭線の接線と同じ方向の円弧成分のみを描くことによって、円の中心を効率よく検出することができる。これと同じように、2 値エッジ画像とテンプレート画像に対してチェインコード化を行う。図 3.5(a) のエッジを例にとり説明する。

画像内を左上から右下に走査して最初に発見された点、図 3.5(a) の点 a_0 を開始点とする。決定された開始点から左回りに連結画素を探索すると、左下方向に次の画素 a_1 が発見される。図 3.4 の規則より、 a_0 は「5」とコード化される。この処理を a_n まで繰り返して行うことで、図 3.5(b) に示すような、チェインコード化されたエッジ画像を生成することができる。

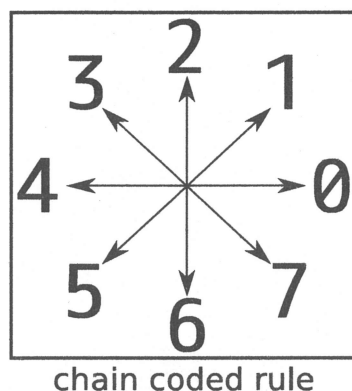


図 3.4: チェインコード化の規則

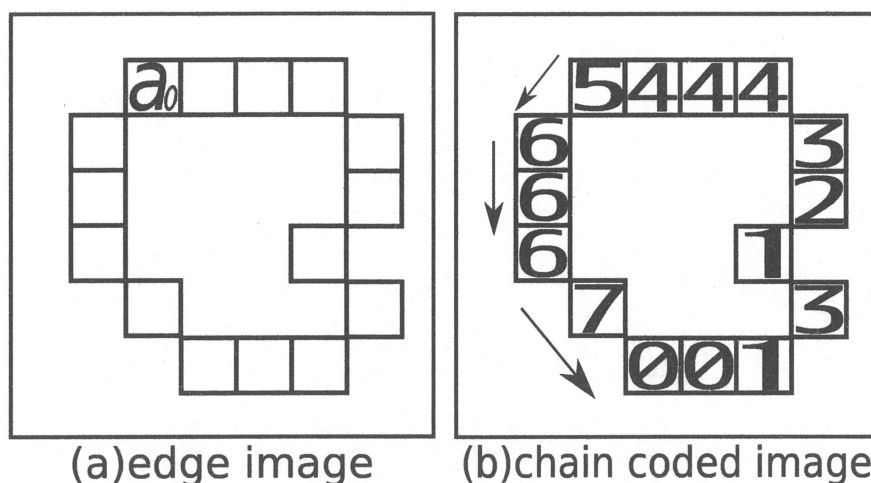


図 3.5: チェインコード化の例

3.5.3 チェインコード化 MFHT による投票

チェインコード化 MFHT と MFHT の投票の違いを図 3.5(a) を例にとり説明する。また、処理の流れを図 3.6 に示す。

3.4 節 (1)(2) と同様の処理、抽出点の決定 (図 3.6(a) の+), テンプレート画像 (図 3.6(b)) の生成を行う。次に、エッジ画像、テンプレート画像を図 3.4 の規則によりコード化を行う (図 3.6(c)(d))。図 3.6(c) のエッジ画素に図 3.6(d) の重心を重ね、それぞれのコードを比較し、一致する座標にのみ投票を行う。これを、図 3.6(c) 全てのエッジに対して行うことで、極大点を取得することができる。チェインコード化 MFHT による投票と、一般化 Hough 変換による投票空間の違いを図 3.7 に示す。輝度値の高い座標ほど投票回数が多いことを示している。(a) と (b) を比較すると、(b) の方が、輝度値の高い点と低い点の差

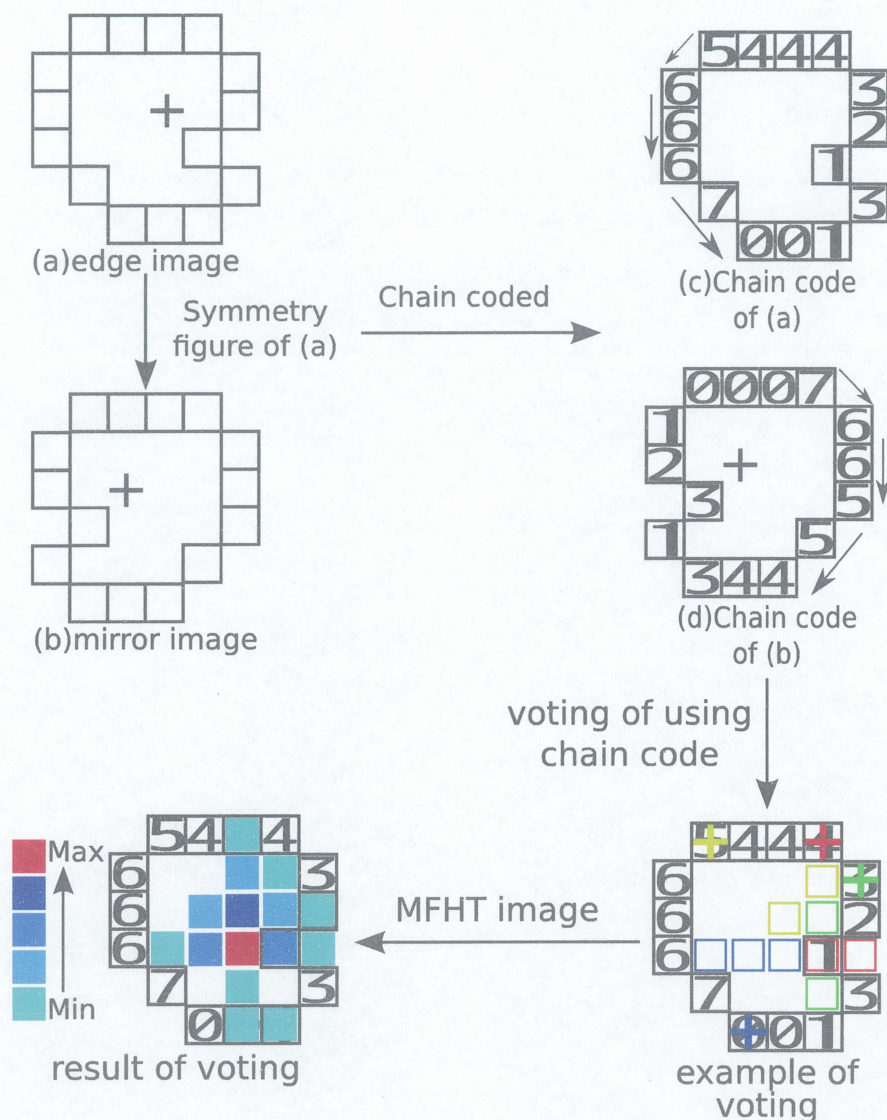


図 3.6: チェインコードによる MFHT

が明確に表れていることが確認できる。一般的に、チェインコードの出現頻度が一樣なら、描画回数は元の $1/8$ になると考えられる。

3.6 投票空間の次元削減手法

上述の処理は、テンプレートを 2 値エッジ画像に適合させるための適切な情報を提供する。しかし、この処理に所要する計算時間は小さくない。特に、3 次元パラメータ空間への投票処理の所要時間が大きい。この処理時間を削減するために、本手法ではパラメータ空間の次元数の削減を行う。人物の顔が垂直軸に対して左右対称に表紙されていれば、顔

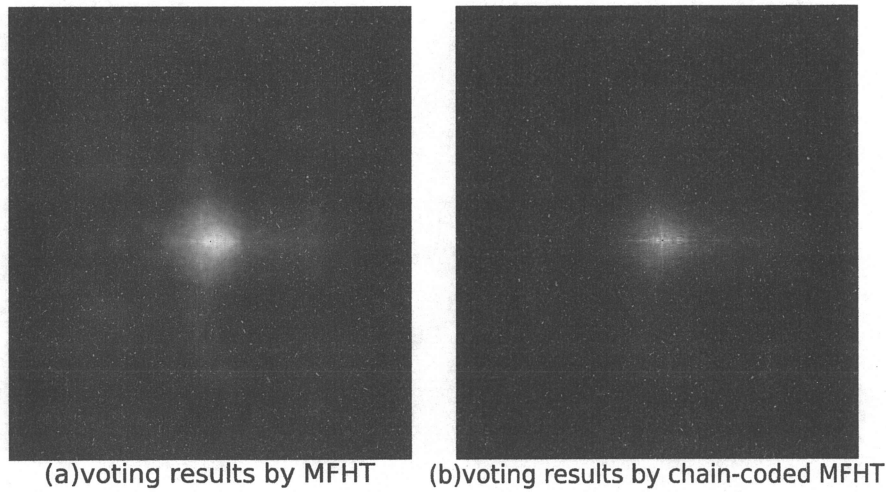


図 3.7: 投票結果の例

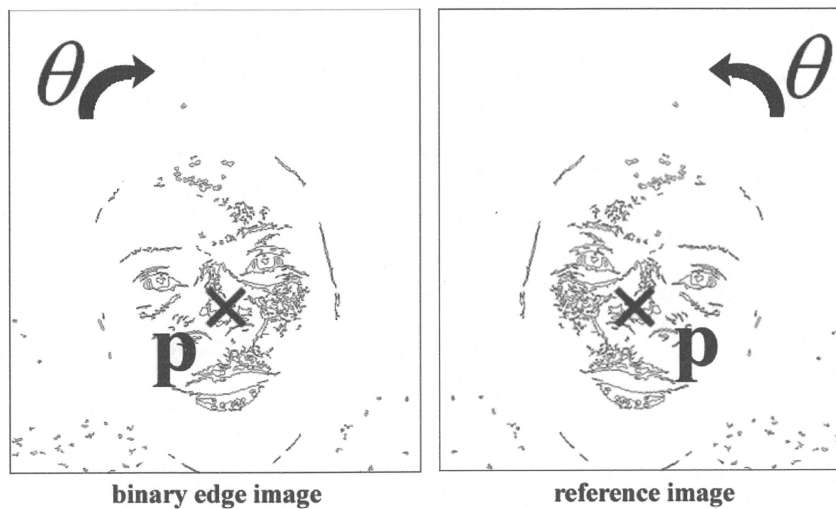


図 3.8: 次元削減手法の概要

は画像中において直立していると言える。これより、参照点と抽出点それぞれの垂直方向の位置 p_y , q_y は、入力画像とテンプレート (鏡面画像) の間で等しくなると考えられる。入力画像とテンプレートが常に鏡像関係を保つように、お互いを逆方向に回転させると、入力画像とテンプレート間の q_y の違いを除去することができる。これにより、パラメータ空間の次元数を q_x と θ の 2 次元に削減できる。図 3.8 は、この手法基本的な概念を図示している。なお、本手法では、人物の頭部の傾きの範囲を $\theta \in [-10^\circ, 10^\circ]$ と仮定している。

3.7 正中線の決定

チェインコード化 MFHT により得られた θ, p, q を式 (3.2) (3.3) に代入して正中線を決定する.

$$\frac{x - (\frac{p_x + q_x}{2})}{\sin \theta} = \frac{y - (\frac{p_y + q_y}{2})}{\cos \theta} \quad (3.6)$$

ここで, 表記を簡単化するために, 以下の変換を行う. 式 (3.6) に $y = 0$ を代入する.

$$\frac{x - (\frac{p_x + q_x}{2})}{\sin \theta} = -\frac{(\frac{p_y + q_y}{2})}{\cos \theta} \quad (3.7)$$

式 (3.7) について解き,

$$x = (\frac{p_x + q_x}{2}) - (\frac{p_y + q_y}{2}) \tan \theta \quad (3.8)$$

を得る.

式 (3.8) は, 検出された正中線の $y = 0$ における値 (x 座標) である. 式 (3.8) の右辺を x_0 とし, 正中線 $L(x_0, \theta)$ を

$$L(x_0, \theta) : \frac{x - x_0}{\sin \theta} = \frac{y}{\cos \theta} \quad (3.9)$$

と, 再定義する.

3.8 まとめ

本手法の処理の流れを図 3.9 にまとめる. 本手法は, グレイスケール画像 (a) を入力とし, 前処理 (3.3 節) によって 2 値エッジ画像を得る. 次に, 2 値エッジ画像 (b) とその鏡像画像であるテンプレート (c) とのマッチングをとるために MFHT (3.4, 3.5 節) を行う. MFHT の結果として (d) が得られる.

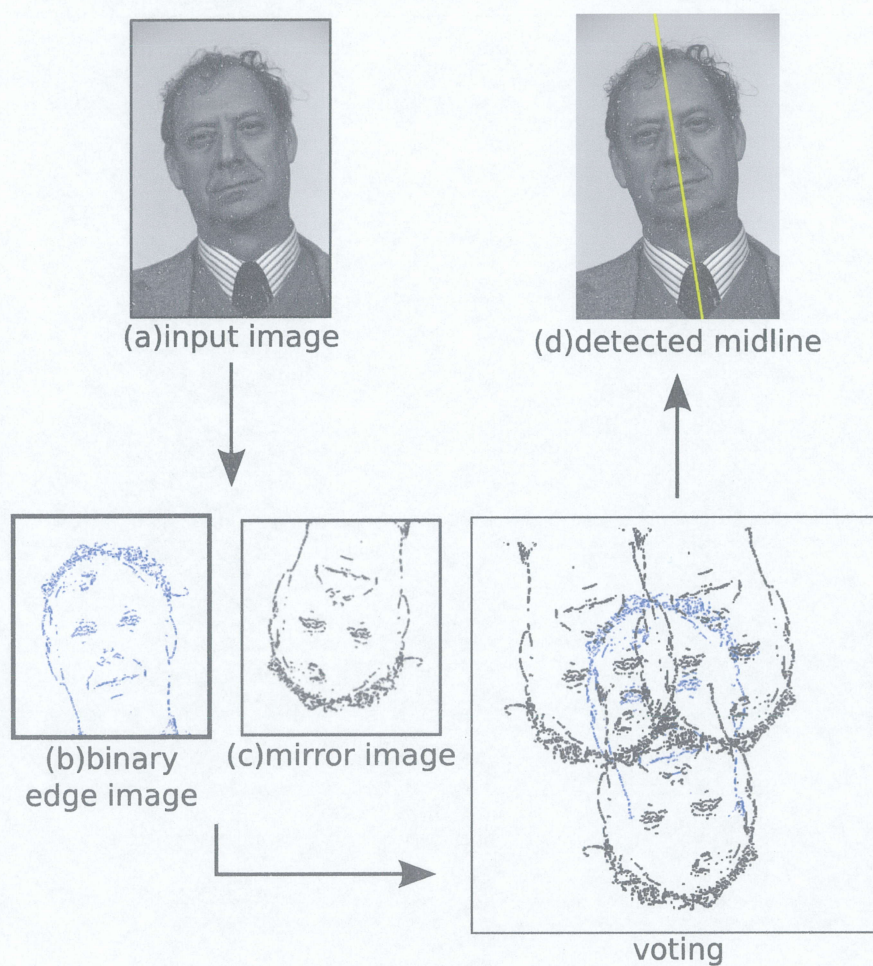


図 3.9: 正中線抽出過程の一例

第 4 章

評価実験

4.1 FERET データベース

The Facial Recognition Technology(FERET) データベース [2] は, 顔認識アルゴリズムの評価を目的に収集されたデータベースである. FERET データベースは, 1000 人を越える個人の顔を収集した大規模データベースであり, 顔認識手法の評価では標準的に用いられている.

本研究で行う評価実験は, FERET データベースに含まれる, 正面顔を撮影した Fa, Fb グループを対象とする. Fa グループは顔認識において主に学習用, 参照用に用いられるグループである. Fb グループは Fa グループと同日に撮影された画像からなるが, Fa グループの画像とは表情が異なる. Fa, Fb グループはそれぞれ 1207 画像, 1202 画像から構成される. 本研究では合計で 2409 画像, 867 人の人物から構成されるデータベースを用いる. なお, 各画像には事前に目視により抽出された顔部品の座標が校正用 (正解) 情報として与えられている. 本研究では, 各パラメータを決定するための予備実験に, Fa, Fb の 2409 枚の画像を用いた.

4.2 正中線検出の正しさの指標

正中線の検出精度を定量的に評価する. 精度評価実験では, 検出した正中線と, 真の正中線を比較する. ここで, 真の正中線とは, 事前に目視抽出されている両目の座標を用いて定めた正中線である. 目視抽出された両目の座標 $(c_{rx}, c_{ry}), (c_{lx}, c_{ly})$ を式 (3.2)(3.3) に代入し, 以下を真の正中線 L_t とする.

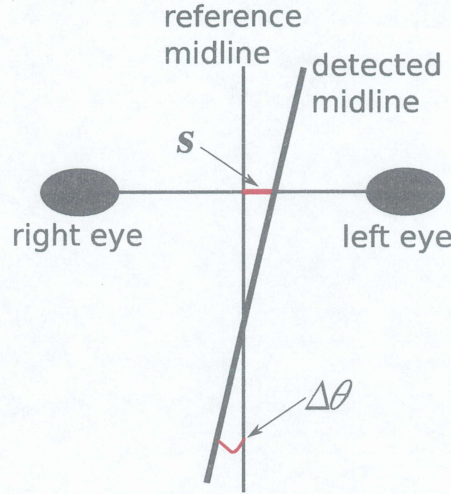


図 4.1: 精度評価のための角度誤差位置誤差

$$\frac{x - (\frac{c_{rx} + c_{lx}}{2})}{\sin \theta} = \frac{y - (\frac{c_{ry} + c_{ly}}{2})}{\cos \theta} \quad (4.1)$$

$$\theta = \tan^{-1} \left(\frac{c_{ry} - c_{ly}}{c_{rx} - c_{lx}} \right) \quad (4.2)$$

文献 [3] では，正中線の検出精度を評価するために，角度誤差 $\Delta\theta$ と，位置誤差 s を用いている．角度誤差 $\Delta\theta$ は，検出した正中線と真の正中線との間の角度の誤差である．また，位置誤差 s は，眼間線分上における 2 つの正中線間の距離を表したものである．図 4.1 は，この 2 つの指標を図示したものである．

4.3 正中線検出におけるパラメータ決定のための予備実験

第 3 章で述べた正中線検出で必要となるパラメータは，前処理における p -タイル法で用いるエッジ画素の割合 p とノイズ除去で用いる連結画素数のしきい値 S_l である．チェインコード化 MFHT に基づく正中線検出手法を，設計用データを入力として，次のパラメータ，

$$p \in \{5, 6, 7, \dots, 20\}[\%]$$

$$S_l \in \{5, 10, 15, \dots, 50\}[pixels]$$

の組み合わせで実行した．それぞれのパラメータの組み合わせにおいて，角度誤差 $\Delta\theta$ ，位置誤差 s に基づいて検出精度の評価を行った．評価実験の結果， $(p, S_l) = (11, 20)$ の組



図 4.2: 提案手法による正中線検出結果

み合わせがもっとも高い検出成功率 95.14%を与えたので、この組み合わせをパラメータとして決定する。

4.4 性能評価実験

提案手法の有用性を示すために、FERET データベースの画像を用いて評価実験を行った。入力画像は、2409 枚の正面顔画像を用いている。

3 節で述べた顔の正中線検出手法を実験した。検出した正中線の例を図 4.2 に示す。各画像中の黄色の線は、検出された正中線である。提案手法は、様々な顔のサイズや頭部の回転角度、眼鏡や髭の有無に関わらず、正しく正中線を検出することができる。

次に、従来手法である MFHT に基づいて正中線を検出した結果と、提案手法のチェーンコード化 MFHT に基づいて正中線を検出した結果を図 4.3 に示す。図中の赤色と黄色の線はそれぞれ、MFHT とチェーンコード化 MFHT による結果である。MFHT の結果では、顔に大して非対称な照明環境な画像、エッジ成分が得られにくい画像において正しく正中線を検出できていない。一方でチェーンコード化 MFHT の結果では、このような条件においても正しく検出されている。

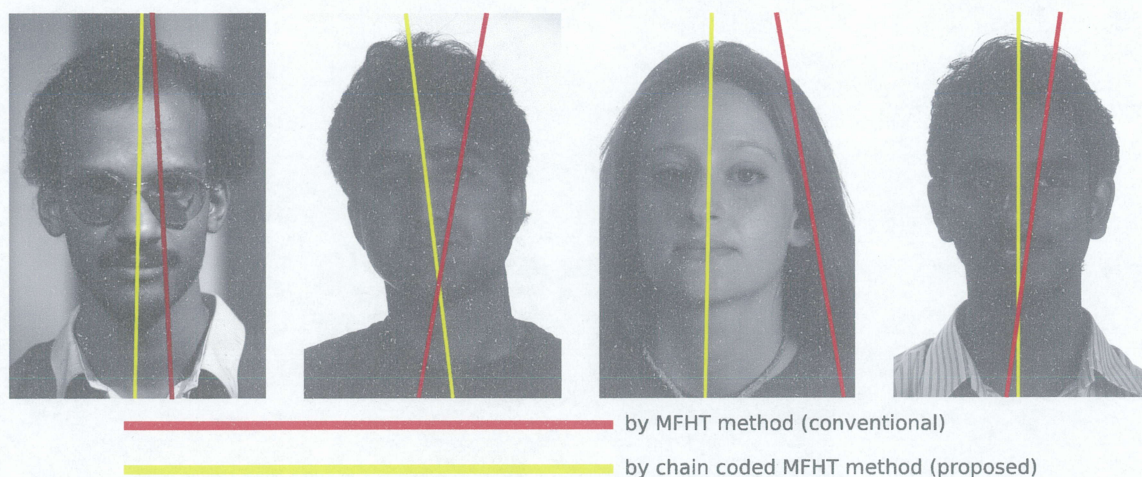


図 4.3: 提案手法と従来手法の比較結果の例

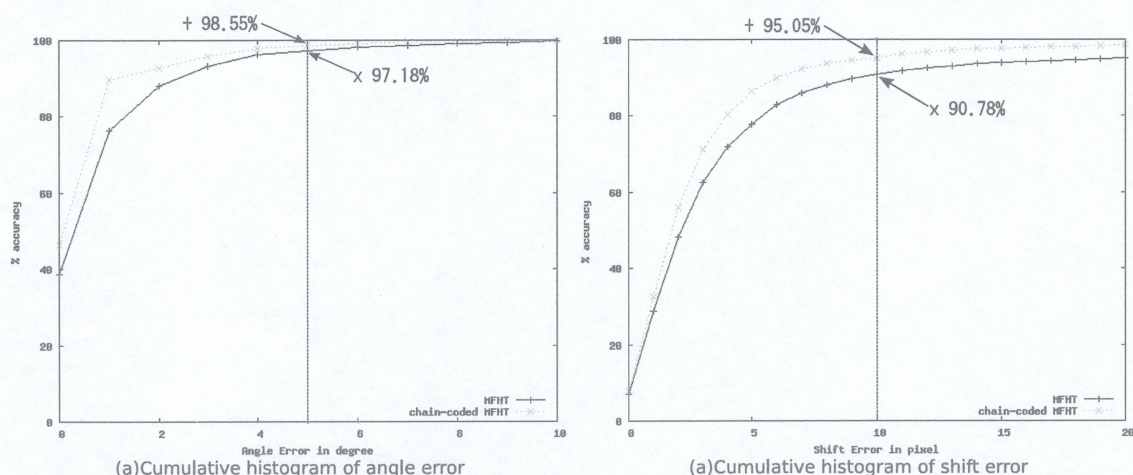


図 4.4: 検出精度の累積ヒストグラム

提案手法の優位性を示すために、提案手法と仲尾ら [1] の従来手法のそれぞれについて、角度誤差と位置誤差を調査した。図 4.4 は、提案手法と従来手法の検出精度を角度誤差と位置誤差の累積ヒストグラムで評価したものである。

図 4.4 より、検出した正中線の 98.55% の正中線は角度の誤差が 5° 以下であることがわかる。これは、提案手法によって入力画像の 98.55% の顔の回転角度が正しく推定できたことを意味する。また、検出した正中線の 95.05% の正中線は位置の誤差が 10 画素以下であることから、入力画像のうち 95.05% の正中線の位置を正しく検出できたと言える。これらの結果から、提案手法は FERET データベース上で仲尾ら [1] の従来手法よりもよい結果を残すことができたと言える。提案手法は正中線を検出するための手法として良好な結果を得られることが分かった。2.66GHz Intel Core2 Quad CPU 上で提案手法を実行し

たところ、1枚の画像から正中線を検出するのに要する所要時間は、従来手法では0.36[s]であったが、提案手法では0.23[s]となり計算時間が削減されていることが分かる。また、ハフ投票空間への投票回数を比較すると、従来手法では10,890,000[回]だったものが、提案手法では2,100,000[回]と大幅な削減に成功した。

図4.5は、これらの結果をグラフにしたものである。(a)と(b)は、それぞれ角度誤差が $\pm 5^\circ$ 以下、位置誤差が ± 10 画素以下の画像の割合である。つまり、正中線検出の成功率である。(c)は、角度と位置の誤差のどちらの条件も満たすときの割合である。これらの結果は、正中線検出の精度が提案手法によって改善されたことを示している。(d)はそれぞれの手法における計算時間の比較である。(e)は1枚の画像を処理する際にかかったハフ投票空間への投票回数を示している。

ここまでの結果は、1枚の画像から1本の正中線のみを検出し、その成功率を評価したものである。もちろん1個の顔には1本の正中線しか存在しないが、検出された正中線を顔部品抽出などに応用する場合、複数の候補を検出してのちの処理に利用の方が都合がいい場合がある。このような場合には複数候補に対する評価が必要である。そこで、図3.3にあるハフ投票空間から、極大点を複数個取り出し、正中線の候補とした。この極大点の数を以降は R_{ml} と表記する。図4.6は、横軸に R_{ml} 、縦軸には図4.7(c)と同様に、角度と位置の誤差のどちらの条件も満たすときの数をプロットしたものである。例えば、 $R_{ml} = 10$ のときは、正中線の候補を10本取り出したときに、その中に検出成功となる正中線が含まれているかどうかを表している。正中線を顔部品抽出へと用いる場合には、角度の誤差が大きく影響するため、許容誤差範囲をさらに厳しくした $\pm 2^\circ$ 以下で成功率を調査した。結果を図4.7に示す。

続いて、提案手法の失敗例について考察する。図4.8は、正中線の誤検出例とその内訳を表している。画像の上段は正中線検出結果画像で、下段は2値エッジ画像である。 $R_{ml} = 1$ の場合の検出エラーは全体の4.86%(117画像)であった。誤検出のうち、もっとも多い失敗例は(A)のグループで、検出の精度が不十分であるため失敗となる。次に多いグループ(B)は、前処理で生成した2値エッジ画像に含まれるノイズが、MFHTに悪影響を及ぼした例である。これに対してグループ(C)は、顔部品のエッジ成分を消失したグループである。提案手法では、顔の左右対称性を利用しているため、顔部品のエッジ成分を積極的に利用し、毛髪や衣服、照明による鏡面ハイライトに相当するエッジはノイズとして除去すべきである。しかし、提案手法は固定パラメータを用いて2値エッジ画像を生成しているため、グループ(B)(C)のようにMFHTにとって不適切な2値エッジ画像が生成されてしまう場合がある。

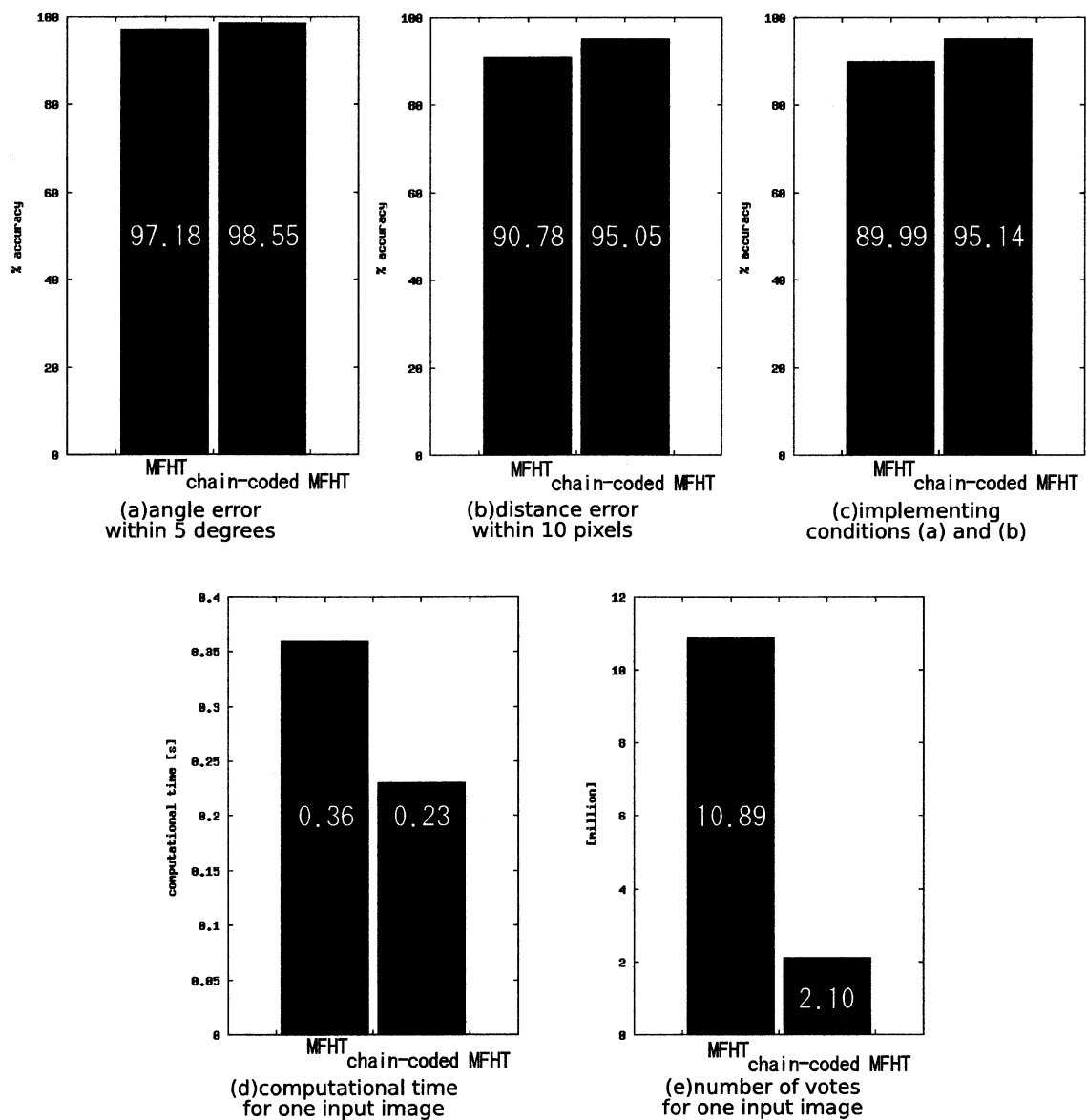
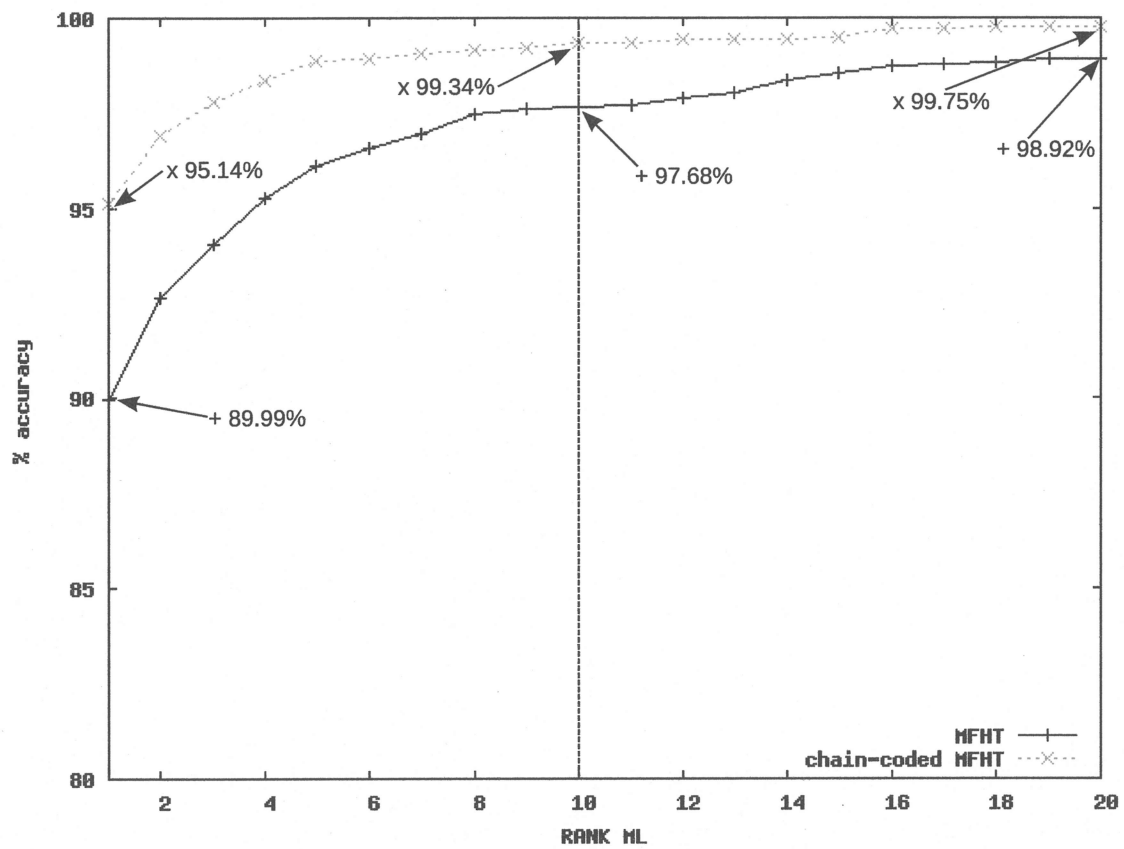


図 4.5: 提案手法と従来手法の検出成功率, 実行時間, 投票回数

図 4.6: 顔正中線検出成功率 ($\Delta\theta = 5^\circ, s = 10$)

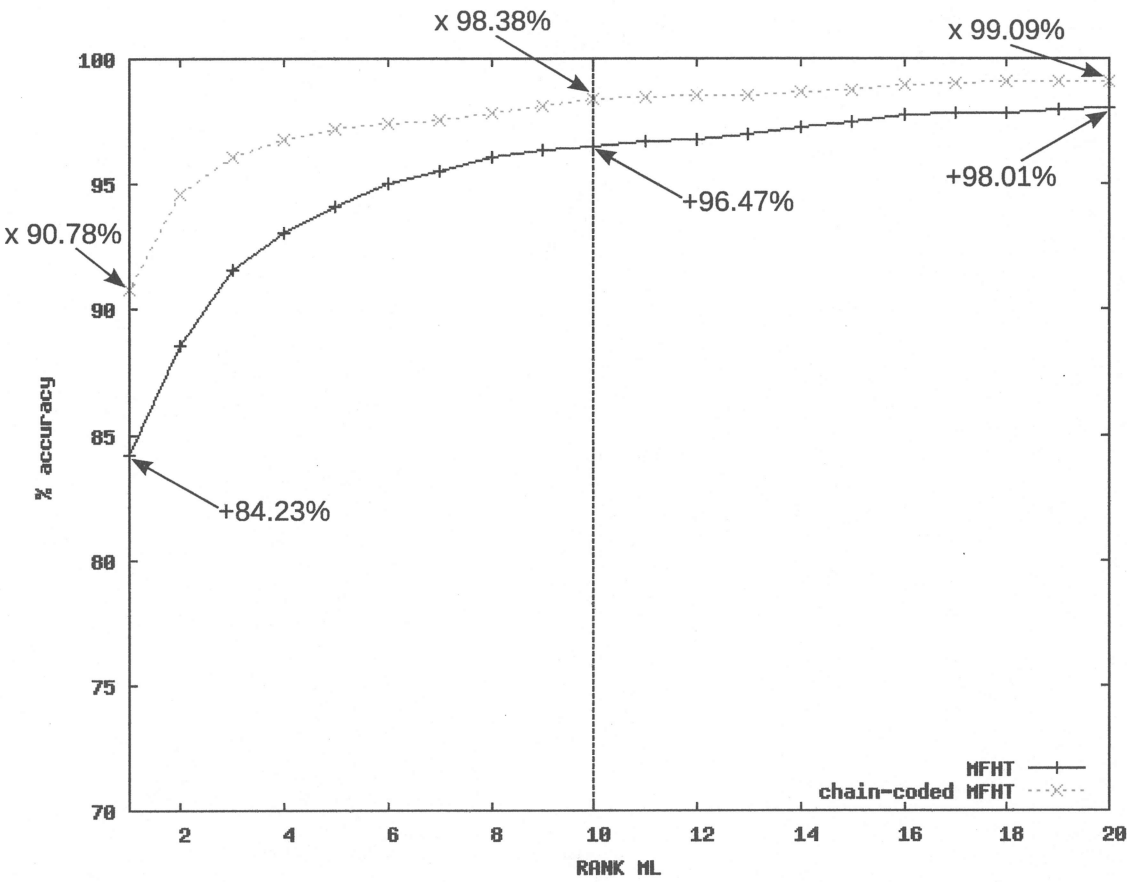


図 4.7: 顔正中線検出成功率 ($\Delta\theta = 2^\circ, s = 10$)

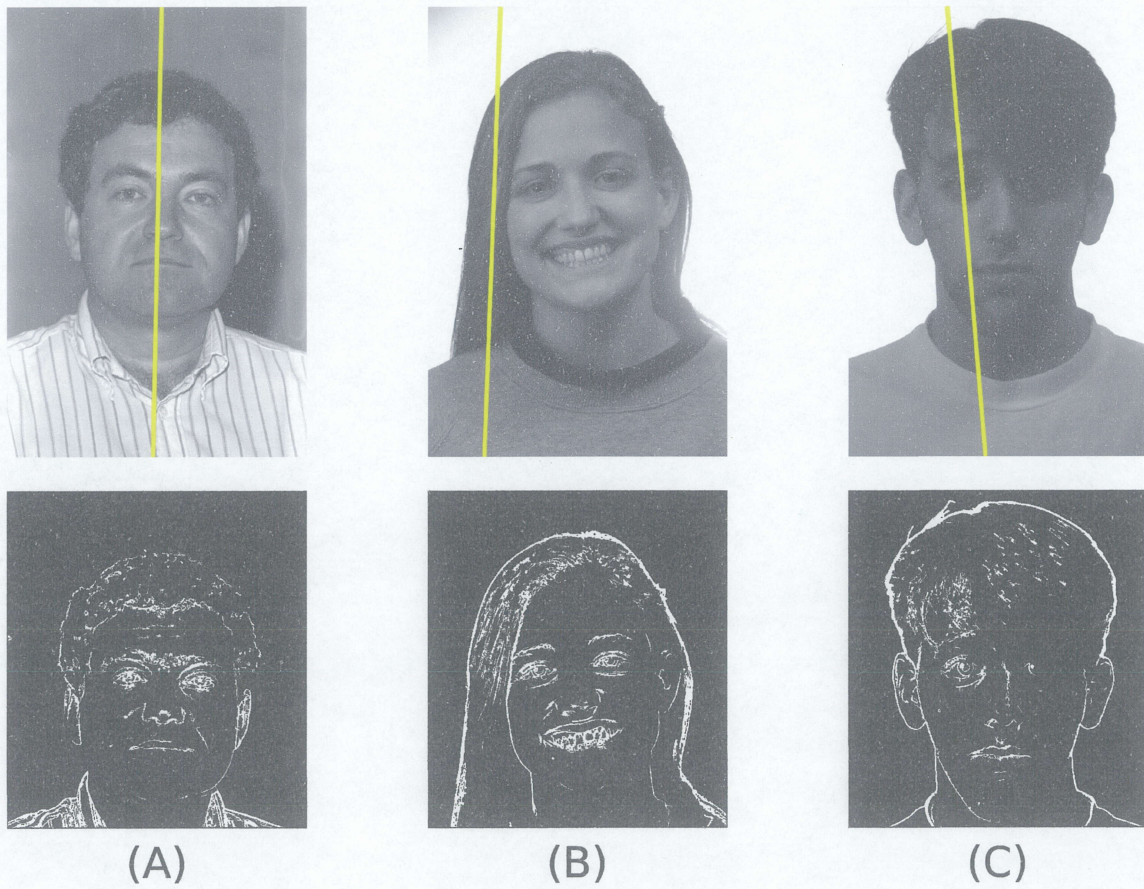


図 4.8: 正中線検出の失敗例

第 5 章

結言

5.1 考察

実験結果より，同じ条件下で従来手法と提案手法を実験した場合，提案手法の方が， $R_{ml} = 1, 10, 20$ においてそれぞれ 5.15%，1.66%，1.08% 向上した． $R_{ml} = 1$ における 1 枚の画像を処理するのに所要する時間は約 36% 削減された．角度誤差，位置誤差で検出精度を評価した場合，それぞれ 1.37%，4.27% の向上がみられた．検出成功率の向上理由としては，チェインコード化を行い，余分な投票を禁止することで，ノイズ削減，正しい極大点の推定が行われたと考えられる．また，投票回数 (図 4.5(e)) の制限が計算時間の短縮に繋がったと言える．

提案手法は，正中線の検出成功率，処理時間共に従来手法を上回る結果となった．これにより，提案手法の有能性が確認された．

提案手法の有能性を証明する実験は，無地または単調な背景の画像を多く含む FERET データベースを用いて行われた．もし，複雑で自由な背景を持つ画像に提案手法を適用すると，複雑な背景のエッジ成分は，MFHT による左右対称性検出に悪影響を及ぼす．そのような場合には，既存の顔検出手法を利用して，背景を取り除く必要になる．

5.2 今後の課題

- (1) 顔正中線検出性能のさらなる改善
Shift error が全体の検出精度を下げているため，位置誤差推定を改善することで性能向上を目指す
- (2) 顔正中線検出処理の高速化
チェインコード化 MFHT とそれ以外の処理に用する計算時間が，それぞれ 0.04[s]，0.19[s] という結果が出た．今後は MFHT 以外の処理に対して高速化を検討する必

要がある.

- (3) 抽出した顔正中線を用いた顔部品, 顔検出
- (4) 複雑な背景を持つ画像に対する正中線検出

が挙げられる.

付録 A

プログラムと実験用データ

A.1 プログラムソースリスト

作成したプログラムを以下のディレクトリに置く。 **okamoto/Kenkyu/**

画像から顔の正中線を検出するメインプログラムを **chain_code/**に置く。また、仲尾らによる正中線を検出するプログラムを **MidlineDetection/**に置く。詳細はスクリプトファイルを参照されたい。

```
| -chain_code
|   | -detect_ML          #正中線を検出する
|   | -do_estimate.sh    #正中線の検出成功率を求める
|   | -do_MAP.sh         #一般化 Hough 変換画像を出力 (calculate を使用)
|   \ -do_pict_MLs       #検出した正中線を描いた画像を出力
|
| \ -MidlineDetection
|   \ -calculate
|     \ -calculate
```

Kenkyu/内の他のディレクトリ及び、**MidlineDetection/**内のファイルについては [1] により詳しい付録があるので参照されたい。

A.2 画像ファイルリスト

しゅうしろ本研究で用いた画像ファイルを **isis** のローカル **/home/FERET/fafb/**

に置く。なお、核実験で用いた画像リストは以下の **okamoto/Kenkyu/chain_code/**に置く。

```
| -chain_code
```

		-fab2409.dat	#本研究で用いた全画像リスト
		\-200.dat	#テンプレート生成用画像リスト

付録 B

修士論文発表会で使用したパワーポイント

謝辞

本研究に関し，研究に対する基本姿勢を御教授下さった木村文隆教授，いつも丁寧なご指導をして下さった若林哲史准教授，計算機の使用や研究の方向性など数多くの適切な助言をして頂いた大山航助教，ディスカッションでたくさんの貴重な意見を出して下さい下さった三宅康二名誉教授に深く感謝いたします。また，日頃いろいろと御世話になった田中みゆき事務官に感謝いたします。

そして，研究に関するたくさんのアドバイスを頂いた研究室の先輩方，充実した楽しい研究室での生活を与えていただいたヒューマンインタフェース研究室の皆さんに感謝致します。

最後になりましたが，長きにわたる私の学生生活を支えてくれた両親，家族に今一度の感謝の意を表して，本論文の結びといたします。

Acknowledgements

Portions of the research in this paper use the FERET database of facial images collected under the FERET program, sponsored by the DOD counterdrug Technology Development Program Office.

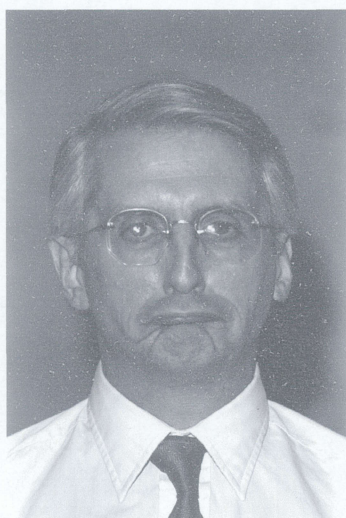
参考文献

- [1] Nozomi NAKAO, Wataru OHYAMA, Tetsushi WAKABAYASHI, and Fumitaka KIMURA: Automatic Detection of Facial Midline And Its Contributions To Facial Feature Extraction. *Electronic Letters on Computer Vision And Image Analysis (ELCVIA)*, Special Issue on Vision & Multimedia Processing, Vol.6, No.3, pp.55-66.
- [2] P.J.Phillips, H.Moon, S.A.Rizvi, and P.J.Rauss: The feret evaluation methodology for face recognition algorithms. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.22, pp.1090-1104, 2000.
- [3] X.Chen, P.J.Flynn, and K.W.Bowyer: Fully automated facial symmetry axis detection in frontal color images. In *4-th IEEE Workshop on Automatic Identification Advanced Technologies*, pp.106-111, 2005.
- [4] P.S.Hiremath, and Ajit Danti: Detection of multiple faces in an image using skin color information and lines-of-separability face model. *International Journal of Pattern Recognition and Artificial Intelligence*, Vol.20, No.1, pp.39-61, 2006.
- [5] G.Dedeoğlu, T.Kanade, and S Baker: The asymmetry of image registration and its application to face tracking. *IEEE Transaction of Pattern Analysis and Machine Intelligence*, Vol.29, No.5, pp.807-823, 2007.
- [6] M.-H.Yang, D.J.Kriegman, and N.Ahuja: Detecting faces in images: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.24, No.1, pp34-58, 2002.
- [7] W.Zhao, R.Chellappa, P.Phillips, and A.Rosenfeld: Face recognition: A literature survey. *ACM Computing Surveys*, Vol.35, No.4, pp.399-458, 2003.
- [8] M.Kirby, and L.Sirovich: Application of the Karhunen-Loeve procedure for the characterization of human faces. *IEEE Trans. PAMI*, Vol.12, 1990.
- [9] L.Sirovich and M.Kirby: Low-dimensional procedure for the characterization of human face. *J.Opt.Soc.Am.* Vol4, pp.519-524, 1987.
- [10] T.Kanade: Computer recognition of human faces. Birkhauser, Basel, Switzerland, and

- Stuttgart, Germany, 1973.
- [11] L.Wiskott, J.M.Fellous, and C.VON DER Malsburg: Face recognition by elastic bunch graph matching. *IEEE Trans.PAMI*, Vol.19, pp.775-779, 1997.
 - [12] A.Pentland, B.Moghaddam, and T.Starner: View-based and modular eigenspaces for face recognition. *IEEE Conference on Computer Vision And Pattern Recognition*.
 - [13] Guangzheng Yang, and Thomas S.Huang: Human face detection in a complex background. *Pattern Recognition*, Vol.27, No.1, pp.53-63, 1994.
 - [14] K.C.You, and R.Cipolla: Feature-based human face detection. *Image and Vision Computing*, Vol.15, No.9, pp.713-735, 1997.
 - [15] M.Hamouz, J.Kittler, J.-K.Kamarainen, P.Paalanen, H.Kälviäinen, and J.Mates: Feature-based affine-invariant localization of faces. *IEEE trans. On Pattern Analysis an Machine Intelligence*, Vol.27, No.9, pp.1490-1495, 2005.
 - [16] O.Jesorsky, K.J.Kirchberg, and R.W.Frischholz: Robust face detection using the hausdorff distance. In *Proc Int'l Conf. Audio- and Video-Based Biometric Person Authentication*, pp.90-95, 2001.
 - [17] Ying Dai, and Yasuaki Nakano: Face-texture model based on sgld and its application in face detection in a color scene. *Pattern Recognition*, Vol.29, No.6, pp.1007-1017, 1996.
 - [18] I.Craw, D.Tock, and A.Bennett: Finding face features. In *Proceedings of Second European Conf. Computer Vision*, pp.92-96, 1992.
 - [19] A.Lanitis, C.J.Taylor, and T.F.Cootes: An automatic face identification system using flexible appearance models. *Image and Vision Computing*, Vol.13, No.5, pp.393-401, 1995.
 - [20] T.F.Cootes, C.J.Taylor, D.Cooper, and J.Graham: Active Shape Models -Their Training and Application. *Computer Vision and Image Understanding*, Vol.61, No.1, pp.38-59, Jan.1995.
 - [21] M.Turk, and A.Pentland: Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, Vol.3, No.1, pp.71-86, 1991.
 - [22] H.A.Rowley, S.Baluja, and T.Kanabe: Neural network-based face detection. *IEEE Transaction of Pattern Analysis and Machine Intelligence*, Vol.20, No.1, pp.23-38, 1998.
 - [23] A.V.Nefian, and M.H.Hayes: Hidden Markov models for face recognition *Proceeding, International Conference on Acoustics, Speech and Signal Processing*, 2721-2724.
 - [24] M.Lades, J.Vorbruggen, J.Buhmann, J.Lange, C.Malsburg, R.Wurtz, W.Konen: Distortion invariant object recognition in the dynamic linc architecture. *IEEE Trans. Comput*, Vol.42, pp300-311.

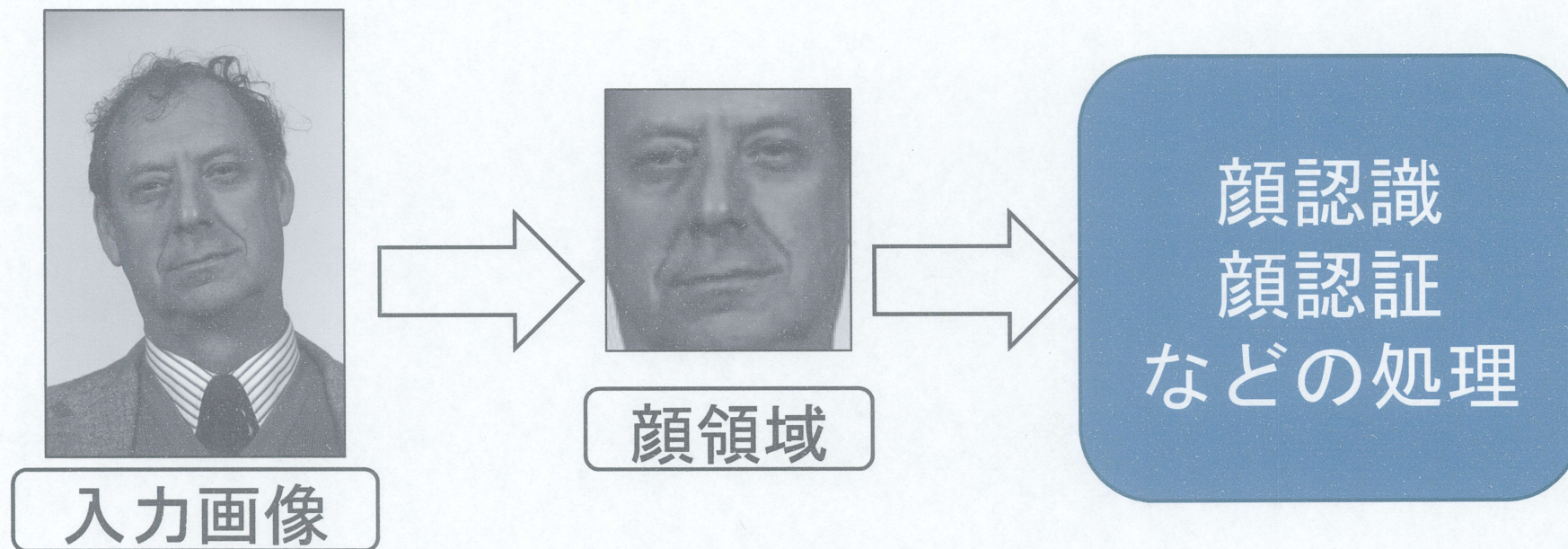
- [25] M.Sato, and H.Ogawa: Optimization of the generalized hough transform. In *Proceeding on 4-th International Conference on Image Analysis and Processing*, 1987.
- [26] 田村秀行（編）：コンピュータ画像処理, pp.182-199. オーム社, 2002.
- [27] 田村秀行（編）：コンピュータ画像処理, pp.138-142. オーム社, 2002.
- [28] 田村秀行（編）：コンピュータ画像処理, pp.151-152. オーム社, 2002.

チェインコード化Merlin-Farber法を用いた正面顔正中線の自動抽出



ヒューマンインタフェース研究室
岡本 大地

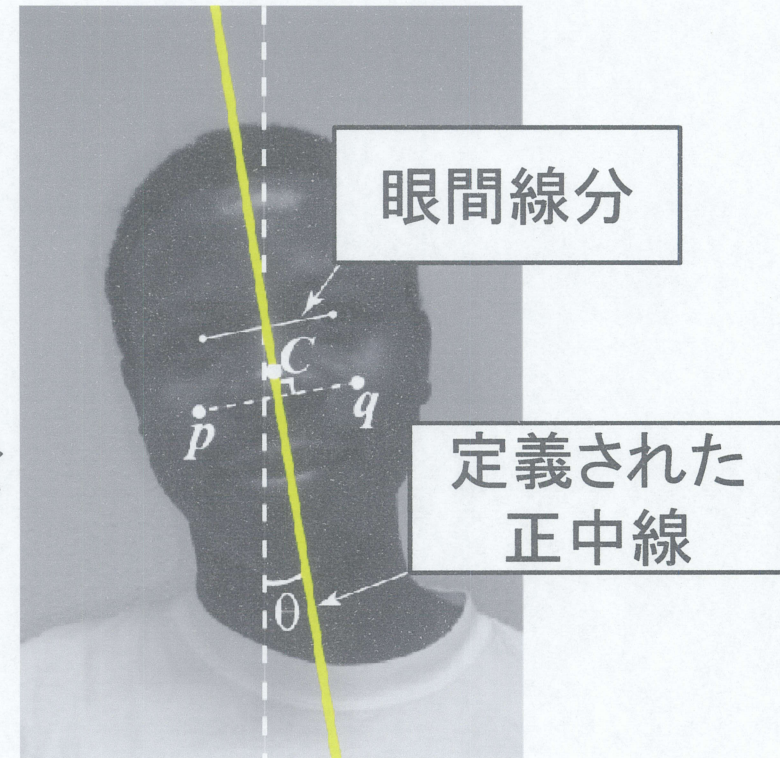
研究背景



- ・ 顔領域だけを正確に抽出する必要がある.
- ・ 顔検出・顔抽出を含む前処理が必要.
- ・ 前処理の簡単化に顔の正中線を利用する.

正中線とは

- ▶ 正中線
 - ▶ 眼間線分の垂直2等分線.
 - ▶ 顔の線対称軸として検出.
- ▶ 正中線は対称な2点 p, q を検出することで求めることができる.



正中線の応用

- ▶ 顔部品抽出に要する計算コストの削減
- ▶ 左右対称な顔による顔認証[1]
- ▶ 明暗のある顔の再構成
- ▶ 顔の対称性の評価

[1] P. Quintiliano et al: "Practical procedures to improve face recognition based on eigenfaces and principal component analysis", In *The 5th international Conference on Pattern Recognition and Image Analysis : New information Technologies*, Vol.11, pp 372-375, 2001

正中線抽出への取り組み

- ▶ 人物の顔は左右対称であるという仮定を用いる.
- ▶ Merlin-Farberハフ変換(MFHT)[2]を用いることで顔画像中の対称な2点を検出する.
- ▶ 仲尾らの正中線検出手法[3]では、抽出成功率90.0%, 処理時間0.36(s)を実現した.

[2] P. Merlin and D. Farber: "A parallel mechanism for detecting curves in pictures", *IEEE Transactions on Computers*, 100(24):96-98, 1975

[3] N. Nakao et al: "Automatic Detection of Facial Midline And Its Contributions To Facial Extraction", *Electronic Letters on Computer Vision And Image Analysis*, Vol.6, No.3, pp55-66, 2007

目的

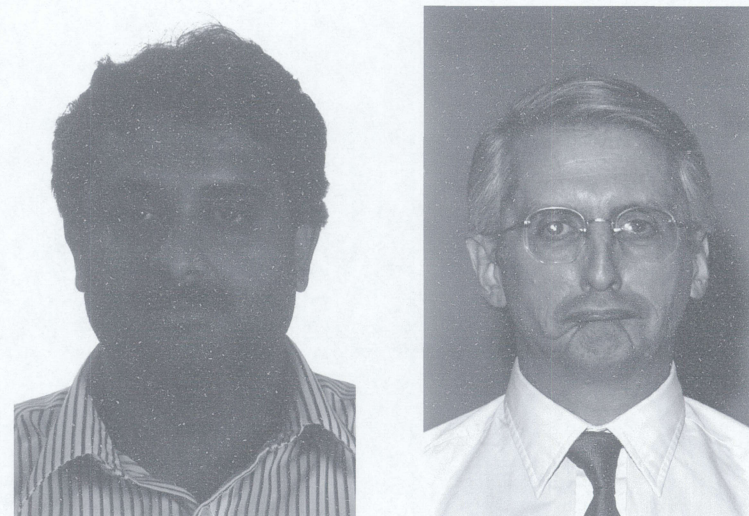
- チェインコード化を用いた新しい正中線抽出のアルゴリズムを提案.
- 従来手法である, 仲尾らの正中線検出手法[3]に比べより高速, 高精度に正中線を抽出することを目指す.

[3] N. Nakao et al: "Automatic Detection of Facial Midline And Its Contributions To Facial Extraction", Electronic Letters on Computer Vision And Image Analysis, Vol.6, No.3, pp55-66, 2007

正中線抽出の従来手法

対象画像

- ▶ 1 個の正面顔を含む
- ▶ グレイスケール画像
- ▶ 顔の位置，大きさは任意



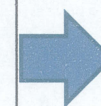
処理の流れ



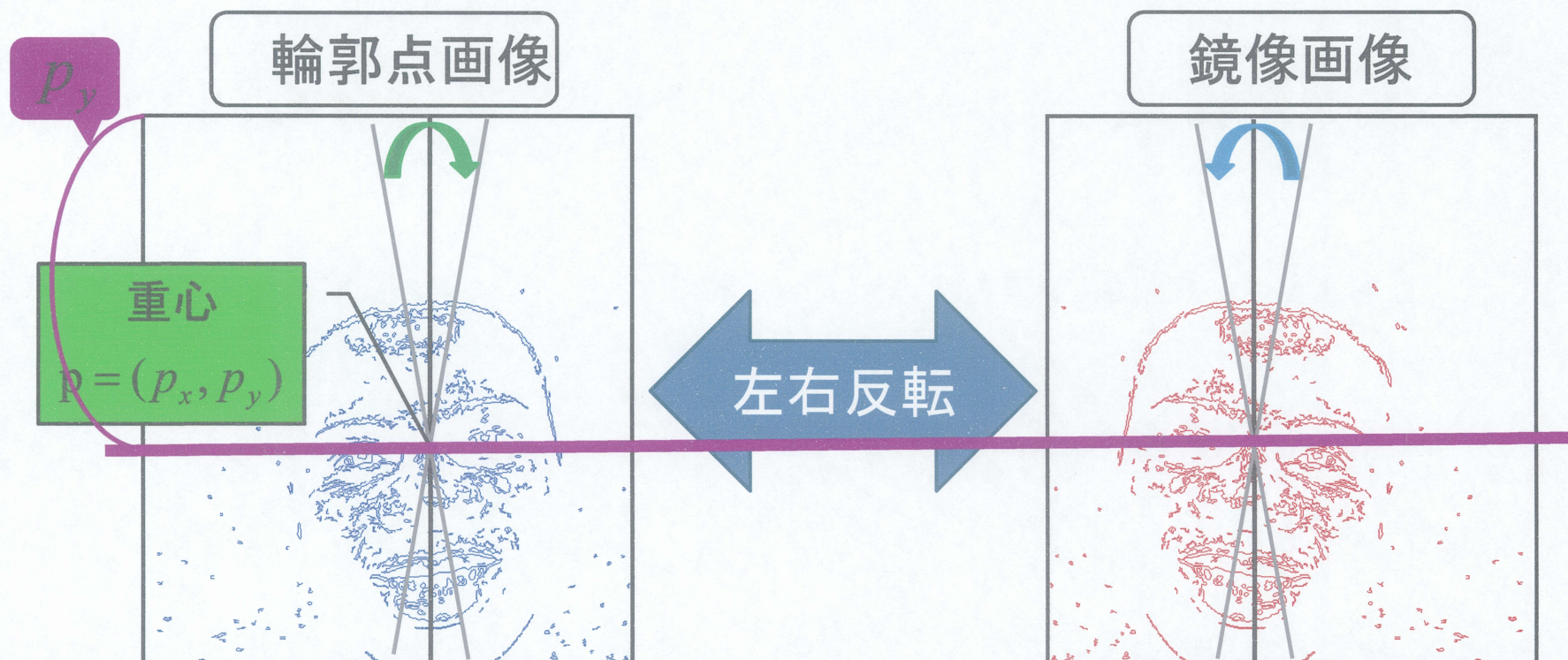
STEP1 前処理



- ① 画像下部1/4を除去
- ② エッジ抽出
(3×3 ソーベルフィルタ)
- ③ 2値化
(p_タイル法: 12%)
- ④ 輪郭線抽出
(ノイズ除去)

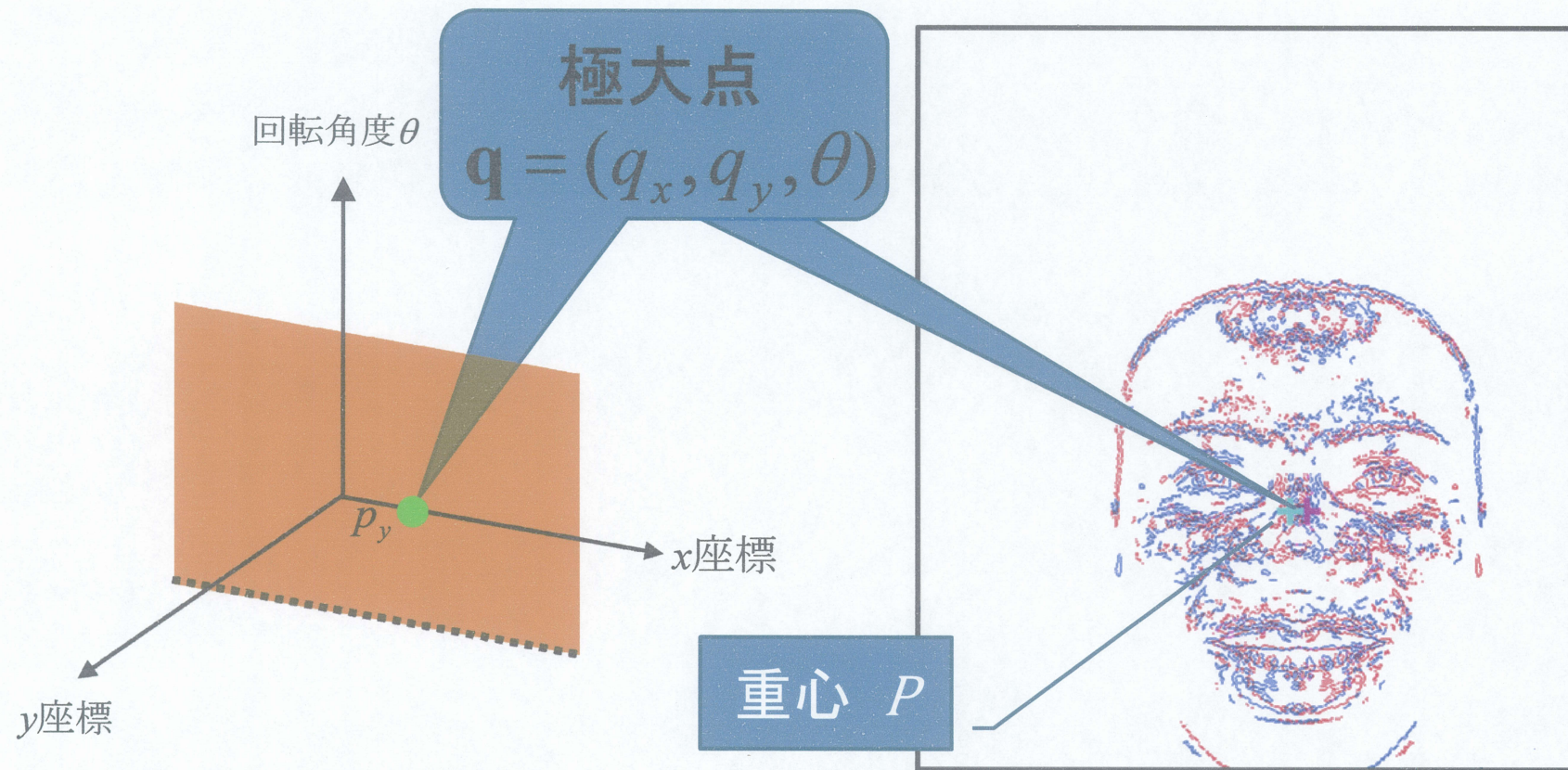


STEP2 正中線検出



MFHTにより回転させながら2枚の画像が最も重なり合う点を求める

STEP2. 1 MFHTによる極大点の検出



STEP2. 2 正中線の決定

- 次式より正中線を決定する

$$\cos \theta \left\{ x - \left(\frac{p_x + q_x}{2} \right) \right\} - \sin \theta \left\{ y - \left(\frac{p_y + q_y}{2} \right) \right\} = 0$$

重心: $p = (p_x, p_y, 0)$
極大点: $q = (q_x, q_y, \theta)$
ただし $q_y = p_y$



正中線抽出画像

チェインコードの利用

▶ 従来手法の問題点

- ▶ 正中線抽出に時間がかかる.
- ▶ 不必要な投票による極大点の誤検出.

チェインコードの利用

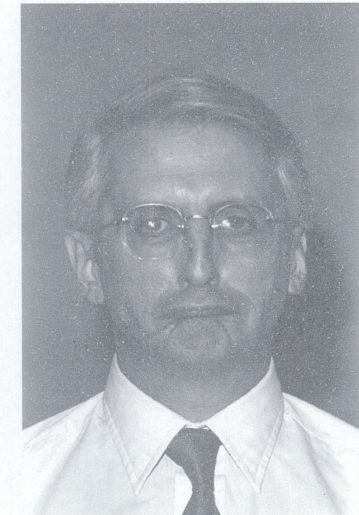


- ▶ コードの一致する点のみに投票を行うので,
 - ① Merlin-Farberハフ変換が高速化される.
 - ② 正中線抽出の精度が向上する.

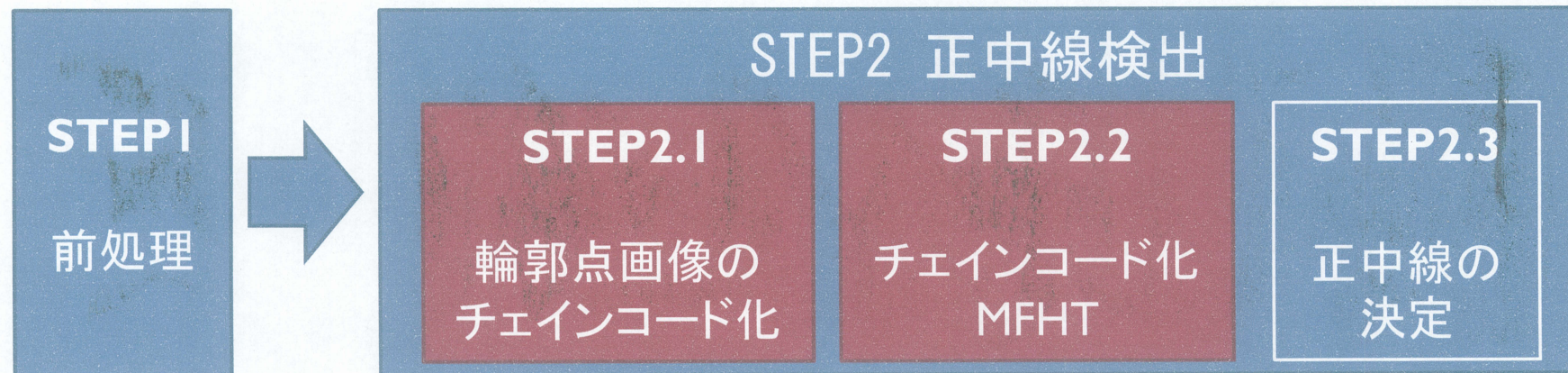
提案手法による正中線抽出

対象画像

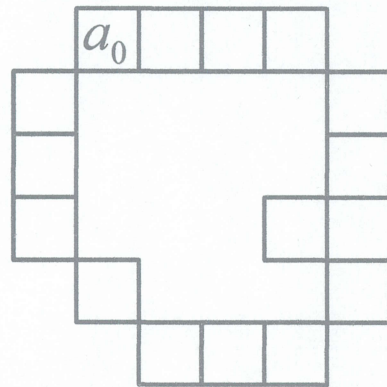
- ▶ 1個の正面顔を含む
- ▶ グレイスケール画像
- ▶ 顔の位置, 大きさは任意



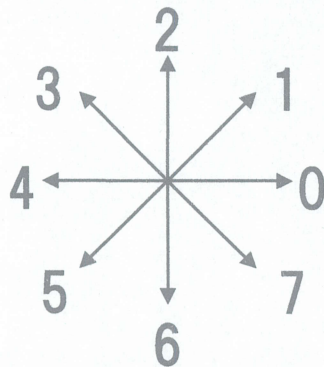
処理の流れ



STEP2.1 輪郭点画像のチェーンコード化

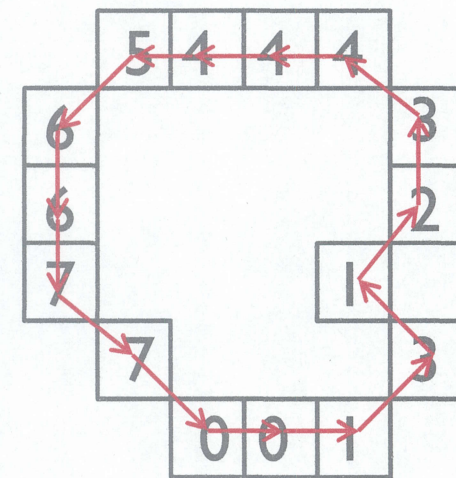


輪郭点画像

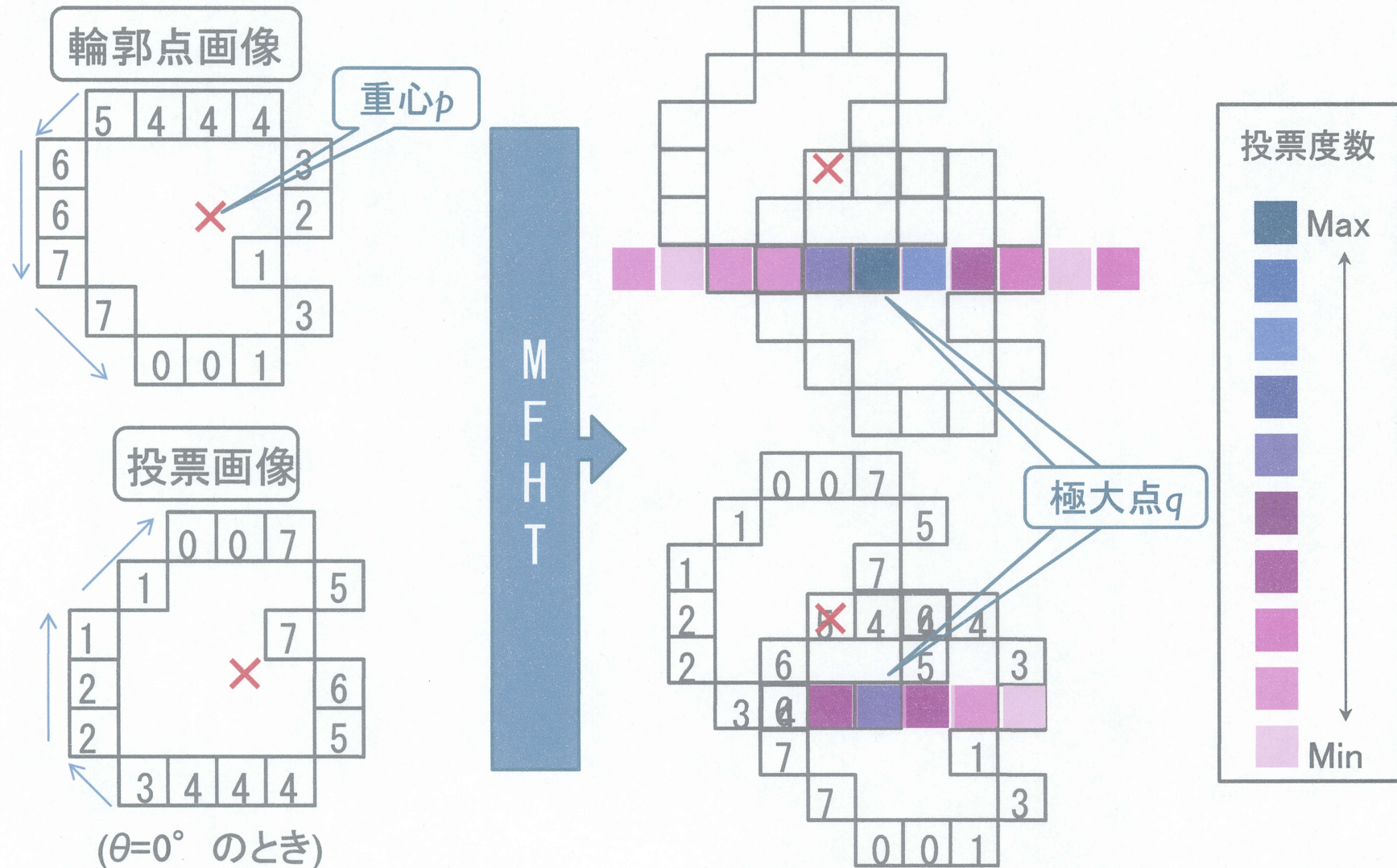


コード化の規則

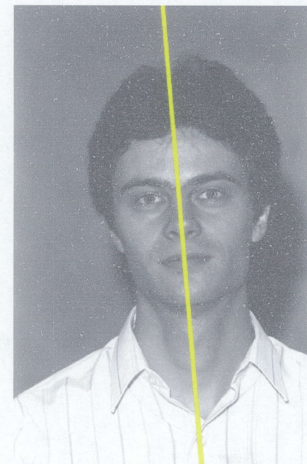
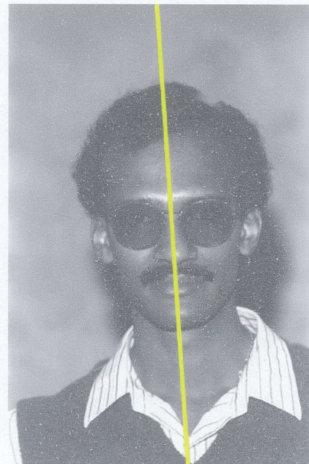
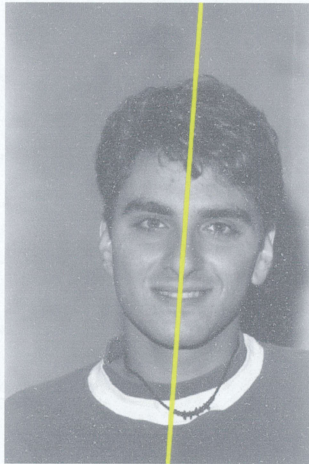
チェーンコード化



STEP3 チェインコード化MFHT



正中線抽出結果



MFHT

提案手法

抽出精度評価実験

- 評価用画像

➤ FERETデータベース [4] の正面顔画像2409枚

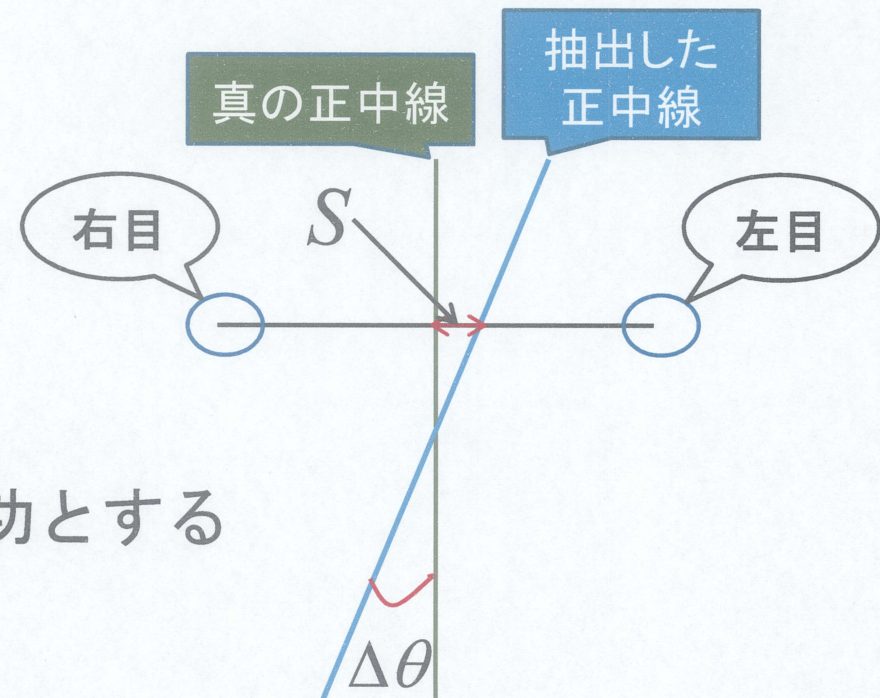
- 評価基準

$$-5^{\circ} \leq \Delta\theta \leq +5^{\circ}$$

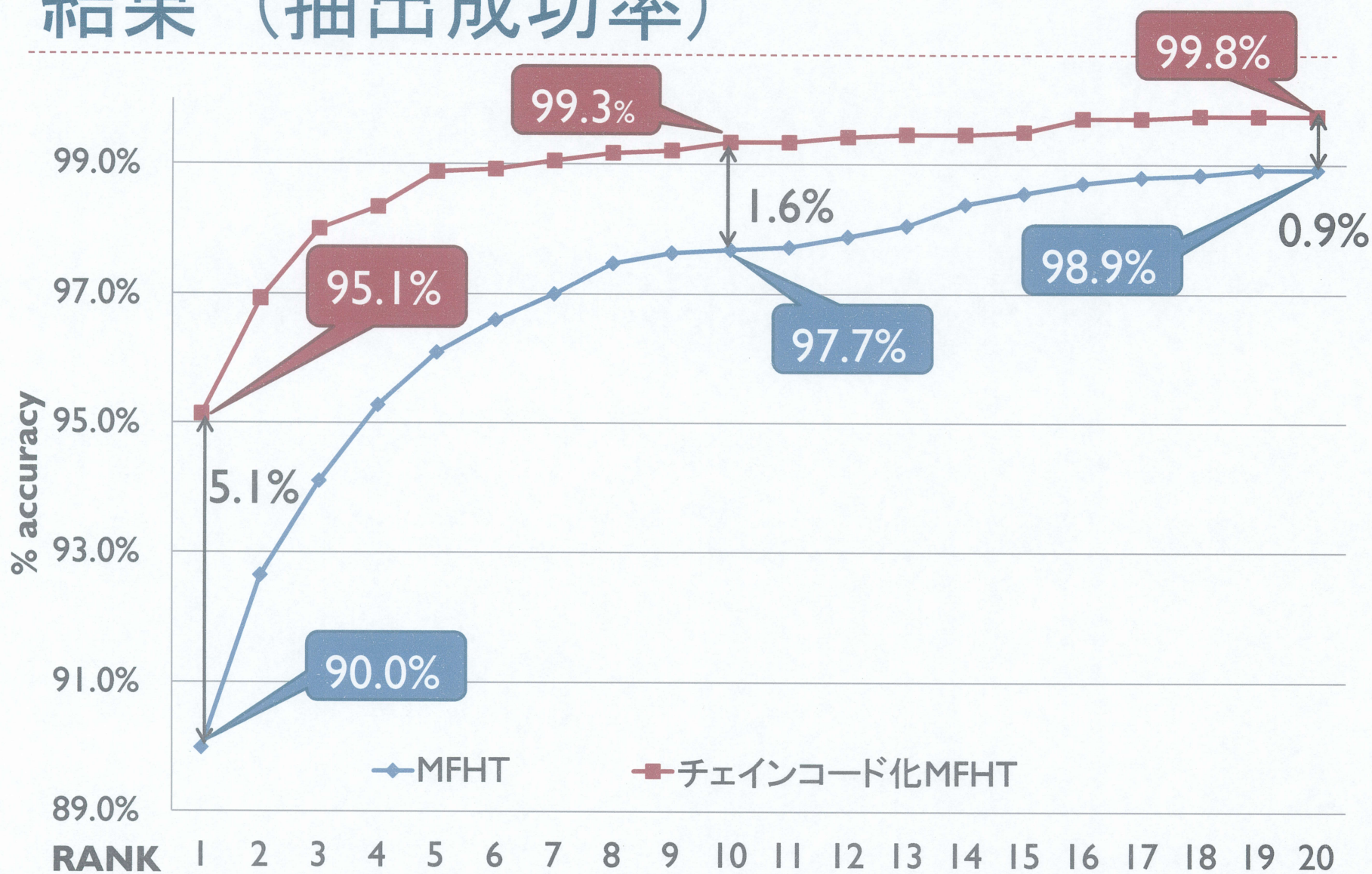
$$-10\text{画素} \leq S \leq +10\text{画素}$$

2つの条件を満たす場合を成功とする

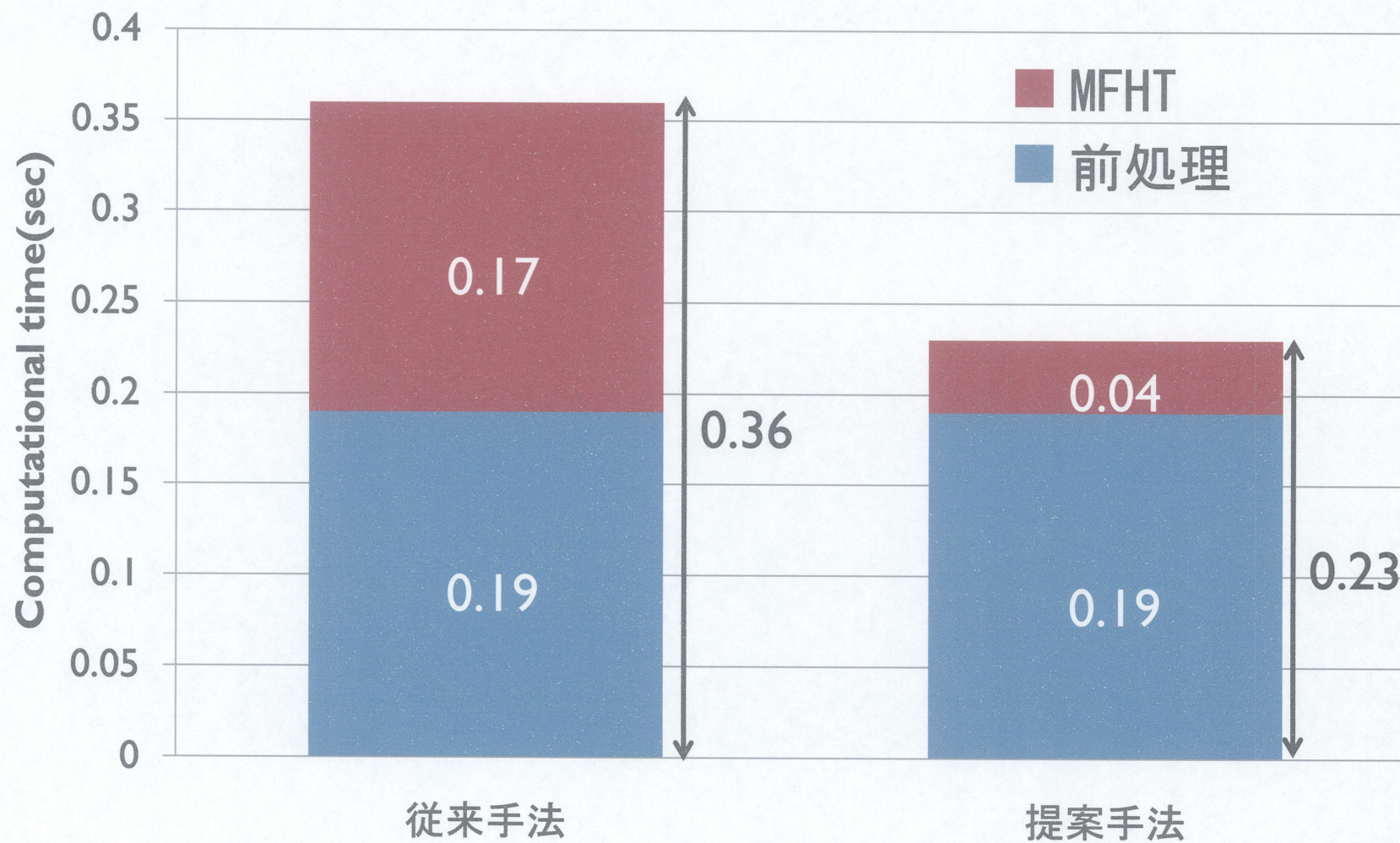
[4] P.J.Phillips et al: "The feret evaluation methodology for face recognition algorithms", IEEE Trans.PAMI, Vol.22, pp.1090-1104, 2000.



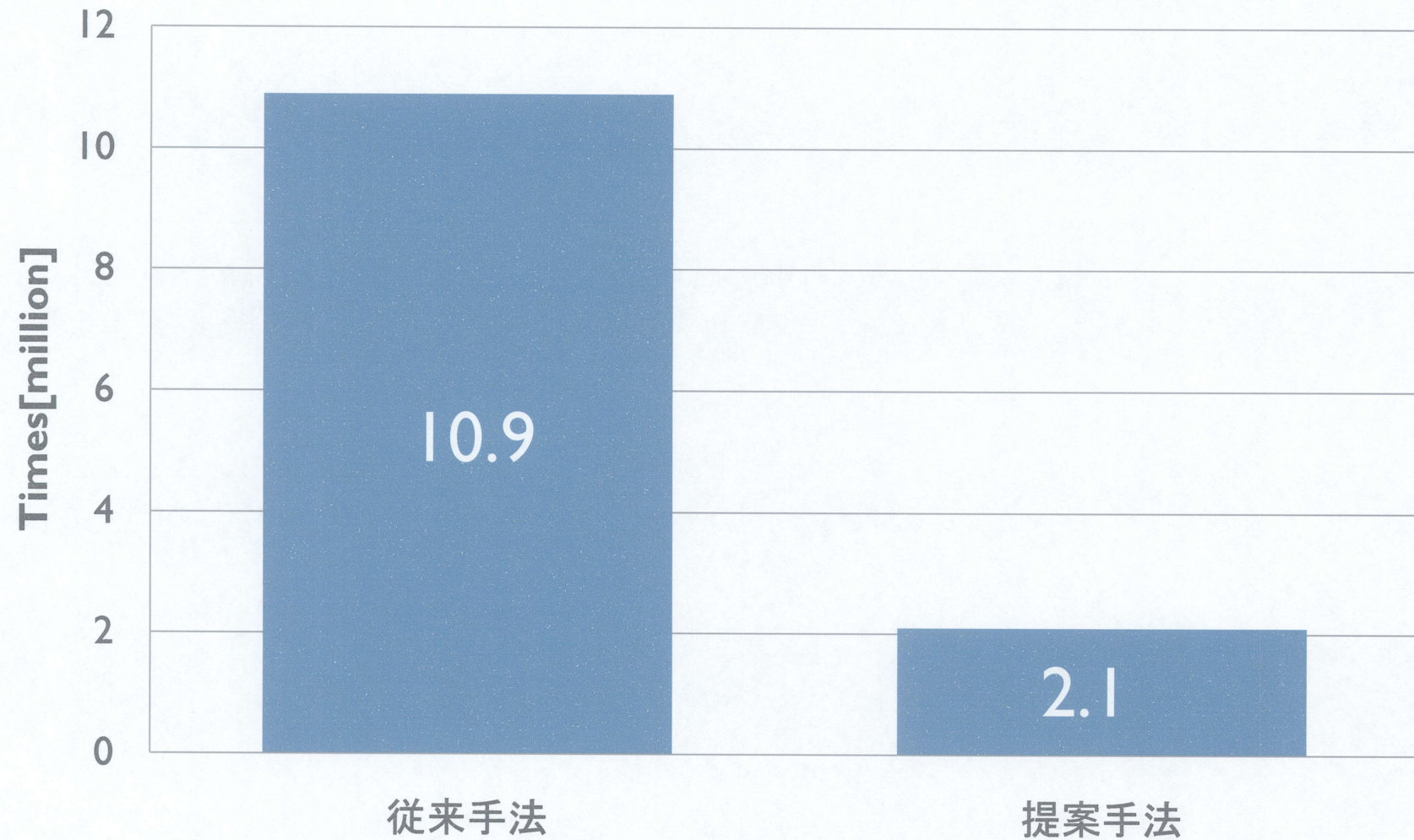
結果（抽出成功率）



結果(処理時間)



結果（パラメータ空間への投票回数）



まとめ

- ▶ Merlin-Farber法による正中線抽出手法に対し、チェインコードを利用した高性能化を行った.
- ▶ 抽出成功率は**5.1%(RANK=1)**, **0.9%(RANK=20)**向上した.
- ▶ パラメータ空間への投票回数は、平均約**1/5**に削減され、1枚あたりの処理時間は平均**36%短縮**された.

今後の課題

- ▶ 顔正中線検出性能のさらなる改善
 - ▶ 複数の正中線候補から真の正中線を求める方法の導入
 - ▶ 適用可能な撮影条件をふやす（複雑な背景、複数顔）
- ▶ 顔正中線検出処理の高速化
- ▶ 顔部品抽出，顔検出への応用

ご清聴ありがとうございました

Mie University
Human Interface Laboratory
Daichi Okamoto