

# 卒業研究論文

---

題 目

1000万データベースに対する高速な顔認識

---

第3グループ

---

指導教員 内海 ゆづ子 助教

---

平成 24 年 ( 2012 年 ) 度 卒業

(No. 1090107023 ) 坂野 悠司

---

大阪府立大学工学部知能情報工学科

# 1000 万データベースに対する高速な顔認識

## 第 3 グループ 坂野 悠司

### 1. はじめに

近年、犯罪やテロへの対策として、個人認証を用いたセキュリティが強化されている。個人認証を行う手法の代表的なものとして顔認識が挙げられる。顔認識の利用法の 1 つとして、監視カメラの映像から、犯罪者などを見つけ出すといった用途に利用できる。監視カメラ映像から犯罪者を捜す場合、データベースには大量の犯罪者の顔画像データが登録されるため、データベースの規模は大きくなる。データベースが大規模になった場合、認識に膨大な処理時間がかかると考えられる。また、監視カメラ映像から得られるクエリは低解像度であり、検出できる特徴点の数が少ないため認識率が低下する。

そこで、本論文では、近似最近傍探索によって検索を高速化し、大規模データベースに対しても高速に顔認識を実現する。また、画像から一定数の特徴量を得ることで低解像度の画像でも高精度に認識を行う。1000 万枚のデータベースを用いて実験を行った結果、1000 位累積分類率で認識率 98.5%、処理時間 185 ms で認識が可能であった。また、実際に監視カメラから得られた低解像度画像を用いて実験を行った結果、1 万枚のデータベースに対して、1000 位累積分類率で認識率 93.3%、処理時間 43 ms で認識が可能であった。

### 2. 特徴抽出

#### 2.1 PCA-SIFT

これまでの研究で顔認識に SIFT を用いた場合、良好な認識結果が得られることが報告されている [1]。そこで、本論文では、特徴量に SIFT よりも低次元に圧縮されており、高速にマッチングを行うことが可能である PCA-SIFT を用いる。

#### 2.2 Dense Sampling

PCA-SIFT の特徴点検出では照明などの影響により特徴点が少しか検出されない場合がある。そこで、特徴点の検出アルゴリズムを使用せず、特徴量を抽出する領域を固定し、Dense Sampling により特徴量を抽出する手法も検討する。

### 3. 認識処理

特徴量の検索処理をクエリの特徴点に対してマッチングを行い、最終的な得票数の高い順に順位付けし認識結果とする。特徴量の検索処理では、処理を高速化するために、ハッシュを利用した近似最近傍探索手法である佐藤らの手法 [2] を用いた。

PCA-SIFT の特徴点検出を用いて特徴量を抽出した場合は、特徴点マッチングの際に図 1(a) のようにすべての特徴量を検索の対象とする方法と、図 1(b) のようにクエリの特徴点との座標の差がしきい値以下の特徴点のみを検索の対象とする方法を利用する。Dense Sampling を用いて特徴量を抽出した場合は、図 1(c) のように同じ位置から抽出された特徴点のみを検索の対象とする。

### 4. 実験と結果

データベースとして、1000 万枚と 1 万枚の顔画像を用いた。クエリとして、実際に監視カメラで撮影した画像である 15 人 15 枚の画像と AR Face Database の left

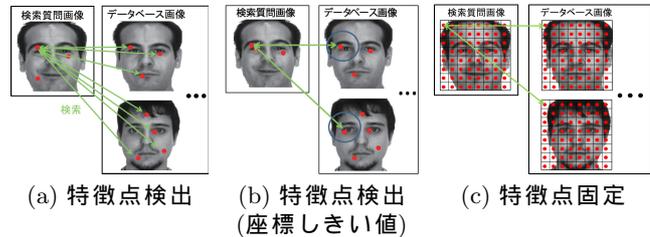


図 1: 特徴抽出手法ごとのマッチング処理

side light on 132 枚を用いた。実験に使用した顔画像は、顔特徴点を自動で検出し、得られた特徴点に基づいて顔の切り出しと顔向き正規化を行っている。また画像は、解像度 512 × 512 [pixel] に正規化し、グレースケール画像に変換した。

クエリに対する 1000 位累積分類率の最も高い認識率と処理時間を調べた。低解像度のクエリを用いた 1 万枚データベースに対する結果を表 1 に、AR Face Database のクエリを用いた 1000 万枚データベースに対する結果を表 2 に示す。監視カメラで撮影した画像では、解像度が低く、特徴点が検出されにくい。そのため、表 1 に示すように、特徴点の数を固定する Dense Sampling を用いた方が高い認識率を示した。1000 万枚データベースに対する実験では、表 2 に示すように特徴点のマッチングの際、検索対象の領域を限定した手法はすべての特徴量を検索の対象とする手法よりも高速に認識することができた。これは、検索対象の領域を限定することで、計算コストを低減することができたからだと考えられる。

表 1: 認識率と処理時間

マッチング処理	特徴点検出	特徴点固定
認識率 [%]	73.3	93.3
処理時間 [ms]	112	43

表 2: 認識率と処理時間

マッチング処理	特徴点検出	特徴点検出 (座標しきい値)
認識率 [%]	98.5	98.5
処理時間 [ms]	216	185

### 5. まとめと今後の課題

本論文では、大規模データベースに対して高速・高精度で認識できること、クエリに低解像度画像を用いた場合でも高精度で認識できることを目指した。今後の課題として、更なる認識率の向上のため、照明条件の変動に対応することが挙げられる。

#### 参考文献

[1] J. Luo et al., "Person-specific SIFT features for face recognition," Proc. of ICASSP 2007, vol.2, pp.593-596, 2007.  
 [2] 佐藤ほか, "概算距離の精度向上による近似最近傍探索の高速化," 信学技報, pp.61-66, 2011.

# 目次

<b>第 1 章 緒論</b>	<b>1</b>
<b>第 2 章 特徴量抽出</b>	<b>3</b>
2.1 特徴点の種類	3
2.1.1 特徴点検出	3
2.1.2 Dense Sampling	4
2.2 特徴量の記述	4
<b>第 3 章 認識処理</b>	<b>7</b>
3.1 投票による認識	7
3.2 近似最近傍探索による検索処理の高速化	7
3.2.1 最近傍候補の選択	8
3.2.2 クエリ適応探索	9
3.2.3 マッチング処理	9
<b>第 4 章 実験</b>	<b>13</b>
4.1 実験条件	13
4.2 1000 万データベースに対する評価	14
4.3 データベースの枚数と処理時間の評価	15
4.4 低解像のクエリに対する評価	15
<b>第 5 章 結論</b>	<b>21</b>
<b>謝辞</b>	<b>23</b>
<b>参考文献</b>	<b>25</b>



# 目 次

2.1	特徴点の例 . . . . .	6
3.1	近似最近傍探索の概要 . . . . .	8
3.2	概算距離 . . . . .	10
3.3	クエリ適応探索 . . . . .	11
3.4	特徴抽出手法ごとのマッチング処理 . . . . .	12
4.1	AR Face Database の画像例. (a) Set1: neutral, (b) Set2: smile, (c) Set3: anger, (d) Set4: scream, (e) Set5: left side light on, (f) Set6: right side light on, (g) Set7: all side light on. . . . .	14
4.2	特徴点の例 . . . . .	17
4.3	1000 万顔画像データベースに対する認識率と処理時間 . . . . .	18
4.4	データベースの枚数と計算時間 . . . . .	19
4.5	低解像度のクエリを用いた実験の画像の例 . . . . .	20

# 第1章 緒論

近年、防犯意識の高まりから監視カメラを設置する一般家庭が増えてきている。防犯用に監視カメラを設置する目的は、監視していることによる犯罪抑止効果を求めるケースと、犯罪が起きた時の証拠の確保のケースに分かれる。特に、後者の目的で設置された場合、監視カメラで撮影された画像がきっかけで事件やトラブルが解決された例は少なくなき、今後ますます監視カメラ画像が犯罪捜査に用いられると考えられる。

現在の犯罪捜査では、すべての犯罪者を覚えておくことは困難であるため、監視カメラで撮影された膨大な画像の中から犯罪者を目視によって見つけ出さなければならず、多くの時間が必要となっている。犯罪者を見つけ出す際に、自動で犯罪者の候補を絞ることが出来れば犯罪捜査のコストが軽減できると考えられる。そこで、本研究では、犯罪捜査支援のための高速な犯罪者の絞り込みを目標とする。犯罪者を絞り込むためには、多くの犯罪者を認識する必要がある。データベースに登録する犯罪者の数が多くなるため、データベースの規模は大きくなる。データベースが大規模になった場合、認識に膨大な時間がかかると考えられる。そのため、顔認識が高速に行える手法でなければならない。さらに、データベースに登録可能な犯罪者の画像は1人あたり数枚しか得られないことが多いため、データベースの画像が1人あたり1枚でも認識できることが求められる。

そこで本論文では、犯罪捜査支援を行うため高速に犯罪者を絞ることを目的とし、局所特徴量と投票を組み合わせた手法を顔認識に適用する。局所特徴量は画像の局所的な特徴を抽出することができるため、顔全体が写っていないクエリ画像でも特徴量を抽出することができ、精度良く絞り込みを行うことができる。また、近似最近傍探索によって検索を高速化し、大規模データベースに対しても高速に顔認識を実現する。さらに、監視カメラ画像から得られるクエリは解像度が低く、検出できる特徴点の数が少ないため認識率が低下する。そこで、低解像度のクエリに対しては、画像から一定数の特徴量を得ることで高精度に認識を行う。

性能評価を行うため、1000万枚のデータベースを作成し、それらを用いて1000位以内

に正解を絞り込むことを目指し、実験を行った。実験の結果、1000 位累積分類率で認識率 98.5%、処理時間 185 ms で認識が可能であった。また、実際に監視カメラから得られた低解像画像を用いて実験を行った結果、1 万枚のデータベースに対して、1000 位累積分類率で認識率 93.3%、処理時間 43 ms で認識が可能であった。

以降、2 章では特徴量抽出の手法について述べ、3 章では認識処理について説明する。4 章では低解像のクエリに対する実験と 1000 万データベースに対する実験について述べる。最後に 5 章で本論文をまとめる。

## 第2章 特徴量抽出

本章では特徴量抽出について説明する。本論文では、特徴量として、低次元に圧縮されているため SIFT よりも高速にマッチングを行うことが可能な PCA-SIFT [1] を使用する。また、従来研究において、顔認識に SIFT を用いた場合、良好な認識結果が得られることが報告されている [2]。PCA-SIFT の処理は、特徴点の検出と特徴量の記述に分かれている。以下に各処理の詳細を述べる。

### 2.1 特徴点の種類

本論文では、PCA-SIFT を抽出する特徴点として、特徴点検出を用いた場合と Dense Sampling により特徴点を固定した場合の 2 つを用いた。以下、それぞれの特徴点について説明する。

#### 2.1.1 特徴点検出

DoG(Difference-of-Gaussian) 処理を用いてスケールスペースにおける極値探索をすることで、キーポイントの位置とスケールを決定する。キーポイント候補点は、スケールの異なるガウス関数  $G(x, y, \sigma)$  と入力画像  $I(u, v)$  を畳み込んだ平滑化画像  $L(u, v, \sigma)$  の差分 (DoG 画像) から求める。ここで、 $\sigma$  はガウシアンフィルタのスケール、 $u, v$  は注目画素の座標であり、 $x, y$  はそれぞれ注目画素からの垂直方向、水平方向の距離である。平滑化画像  $L(u, v, \sigma)$  とガウス関数  $G(x, y, \sigma)$  は、以下の式により求める。

$$L(u, v, \sigma) = G(x, y, \sigma) * I(u, v) \quad (2.1)$$

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \quad (2.2)$$

DoG の結果の画像を  $D(u, v, \sigma)$  とすると、DoG 画像は次式で求まる。

$$\begin{aligned} D(u, v, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(u, v) \\ &= L(u, v, k\sigma) - L(u, v, \sigma) \end{aligned} \quad (2.3)$$

この処理を、 $\sigma$  の初期値  $\sigma_0$  から  $k$  倍ずつ大きくした異なるスケール間で行い、複数の DoG 画像を求める。得られた DoG 画像から極値を検出し、キーポイントとスケールを決定する。DoG 画像の注目画素と、その周りの 26 近傍を比較し、極値であった場合、その画素をキーポイント候補点として検出する。そして、そのときの DoG 画像のスケールをキーポイントのスケールとする。次に、キーポイントのローカライズを行う。検出されたキーポイント候補点の中には DoG 出力値の小さい点やエッジ上の点が含まれており、これらの点はノイズや開口問題に影響を受けやすいという問題がある。そこで、キーポイント候補点の中から、主曲率とコントラストを用いてノイズや開口問題に強いキーポイントに絞り込む。特徴点検出した例を図 2.1(a) に示す。

### 2.1.2 Dense Sampling

画像の解像度が低い場合には、特徴点検出を用いた場合、検出される特徴点の数が少なく認識率が低下するといった問題がある。そこで、特徴点検出を行わず、特徴量を抽出する領域を固定し、Dense Sampling して一定の特徴量を得ることが可能な手法を用いた。特徴抽出に用いるスケールとして  $\sigma = 2, 4, 6, 8, 10$  を使い、特徴点の数は、1 枚の画像当たり合計 880 領域から PCA-SIFT 特徴を抽出する。Dense Sampling では、特徴点のサンプルの間隔は 40, 80, 120, 160, 200 [pixel] とし、格子状に特徴点を配置した。特徴点の例を図 2.1(b) に示す。

## 2.2 特徴量の記述

特徴量の記述の際、特徴点検出を用いる手法では、まず、検出された各キーポイントのオリエンテーションを求める。オリエンテーションは、キーポイント周辺の勾配情報から算出され、キーポイントにおける方向を表す。特徴点を固定する Dense Sampling の特徴点を使用する手法では、オリエンテーションは常に 0 とし、特徴量記述領域の回転は行わ

ない。

キーポイントのオリエンテーションを求めるには、まずキーポイントが検出された平滑化画像  $L(u, v)$  の勾配強度  $m(u, v)$  と勾配方向  $\theta(u, v)$  を以下の式により求める。

$$m(u, v) = \sqrt{f_u(u, v)^2 + f_v(u, v)^2} \quad (2.4)$$

$$\theta(u, v) = \tan^{-1} \frac{f_u(u, v)}{f_v(u, v)} \quad (2.5)$$

$f_u$ ,  $f_v$  は  $u$ ,  $v$  方向の勾配であり、以下のように計算される。

$$f_u(u, v) = L(u + 1, v) - L(u - 1, v) \quad (2.6)$$

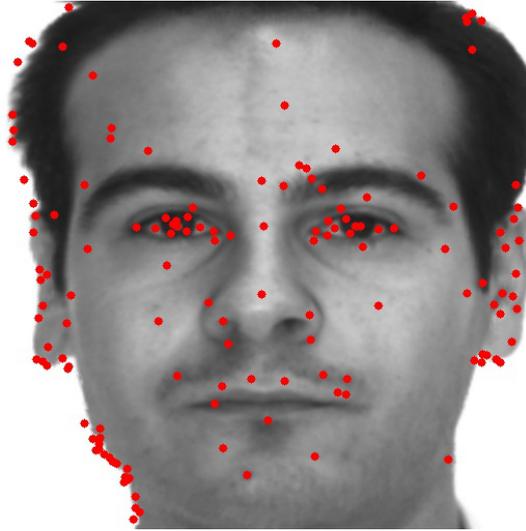
$$f_v(u, v) = L(u, v + 1) - L(u, v - 1) \quad (2.7)$$

局所領域における勾配強度  $m(x, y)$  と勾配方向  $\theta(x, y)$  から重み付方向ヒストグラム  $h$  を以下の式のより作成する。

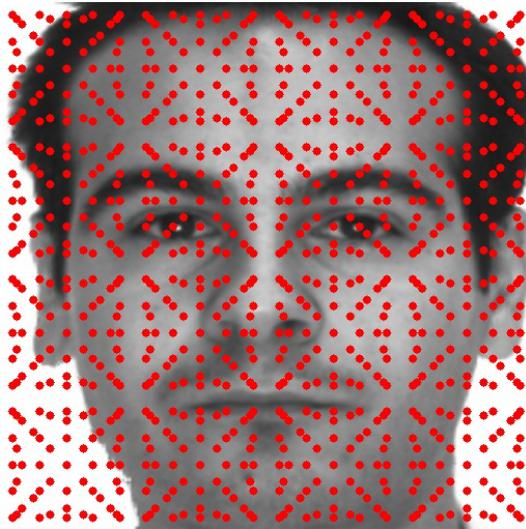
$$h_{\theta'} = \sum_x \sum_y w(x, y) \cdot \delta[\theta', \theta(x, y)] \quad (2.8)$$

$$w(x, y) = G(x, y, \sigma) \cdot m(x, y) \quad (2.9)$$

オリエンテーションは、キーポイント周辺の勾配情報から作成される重み付き方向ヒストグラム  $h_{\theta'}$  により求まる。ここで、 $h_{\theta'}$  は、全方向を 36 方向に量子化したヒストグラムである。この 36 方向のヒストグラムの最大値の 80% 以上となるピークをキーポイントのオリエンテーションとして割り当てる。まず、回転に不変な特徴量を抽出するため、検出したキーポイントを中心とした領域を、オリエンテーションの方向に正規化する。その後、キーポイントのスケールに対応した領域を  $41 \times 41$  のパッチにリサンプリングする。リサンプリングしたパッチの水平・垂直方向の勾配を算出し、 $39 \times 39 \times 2 = 3042$  次元の特徴量を得る。大量の一般画像から抽出された SIFT 特徴量を用いて PCA をかけ 36 次元の基底を計算した。



(a) 特徴点検出



(b) Dense Sampling

図 2.1: 特徴点の例

## 第3章 認識処理

認識処理では、特徴量のマッチングを行い、投票処理により認識を行う。マッチングの処理を高速化するため、近似最近傍探索を適用する。また、マッチングを行う際、誤対応を減らすため、マッチングに用いる特徴点を限定した。以下、投票による認識、近似最近傍探索による検索処理の高速化、マッチングに用いる特徴点の限定について説明する。

### 3.1 投票による認識

検索画像の特徴点ごとにマッチングを行い、検索画像の特徴量から近傍  $k$  点の特徴量の画像 ID に投票する。投票の際に、データベース画像の特徴量と検索画像の特徴量とのユークリッド距離  $d$  の逆比である  $1/d$  の重みをつける。クエリの特徴量とユークリッド距離が最も近いデータベースの特徴量は正解の可能性が高い。そこで、距離が近いものを他の特徴量と比べより多くの投票を行うため、このように重みをつける。

### 3.2 近似最近傍探索による検索処理の高速化

特徴量の検索処理では、データベース内の特徴量すべてと距離計算を行った場合、データベースの規模に比例して処理時間が増加する。そのため、データベースが大規模になると、認識の処理時間が膨大になる。そこで、処理時間を短縮するために近似最近傍探索を用いて検索を行う。ハッシュを用いた一般的な近似最近傍探索の概要を図 3.1 に示す。図 3.1 の緑や赤の線で示すように、データ空間をランダムに生成された基底方向に等間隔に分割することで、空間をバケットと呼ばれる領域に分割してインデクシングを行う。図 3.1 の緑や赤の軸はランダムに生成された基底であり、緑と赤の領域が交差している部分をバケットと呼ぶ。探索時にはクエリと同じバケット内で最近傍探索を実行する。全探索

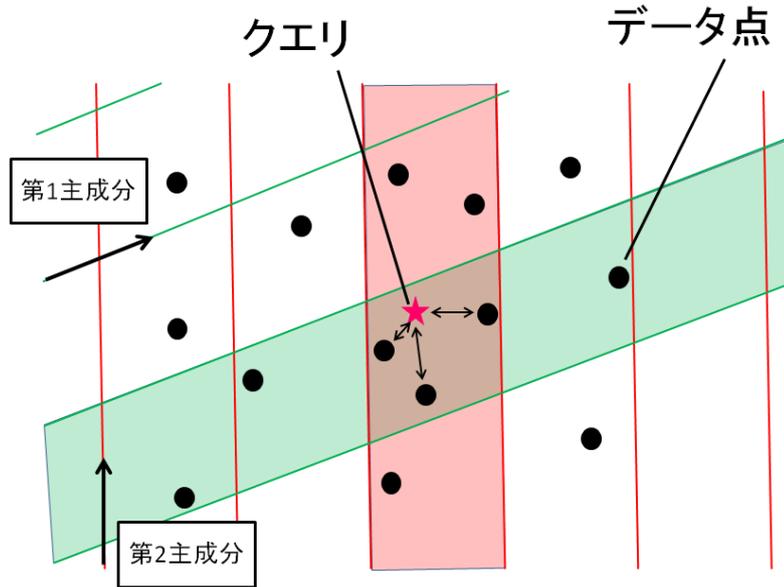


図 3.1: 近似最近傍探索の概要

と比べ、距離計算が削減され、大幅な認識時間の削減が可能である。本論文ではハッシュを利用した近似最近傍探索手法である、佐藤らの手法 [3] を用いた。

佐藤らの手法は、探索領域の大きさではなく最近傍候補の数に制約を設け、最近傍探索をする手法であり、一定の最近傍候補点を得られた上で探索できるという特徴を持つ。クエリのまわりのハッシュ内に含まれるデータ点の密度が異なる場合でも、探索する特徴点の数を一定に保ち、高速かつ高精度に認識ができる。以下、佐藤らの手法の特徴的な部分である最近傍候補の選択方法と、クエリに対応した探索について述べる。

### 3.2.1 最近傍候補の選択

まず、特徴空間において図 3.2(a) のように主成分分析によって求めた上位  $\psi$  主成分基底に対して等間隔で格子状に空間を分割する。図 3.2(a) は、データ点の上位 2 主成分基底を用いた場合で、緑と橙の点線により空間を分割している。そして、クエリと各データが属するバケットの重心までの距離の二乗和を概算距離とし、概算距離がしきい値以下になるバケットを探索範囲として指定し距離計算を行う。クエリと各データが属するバケット

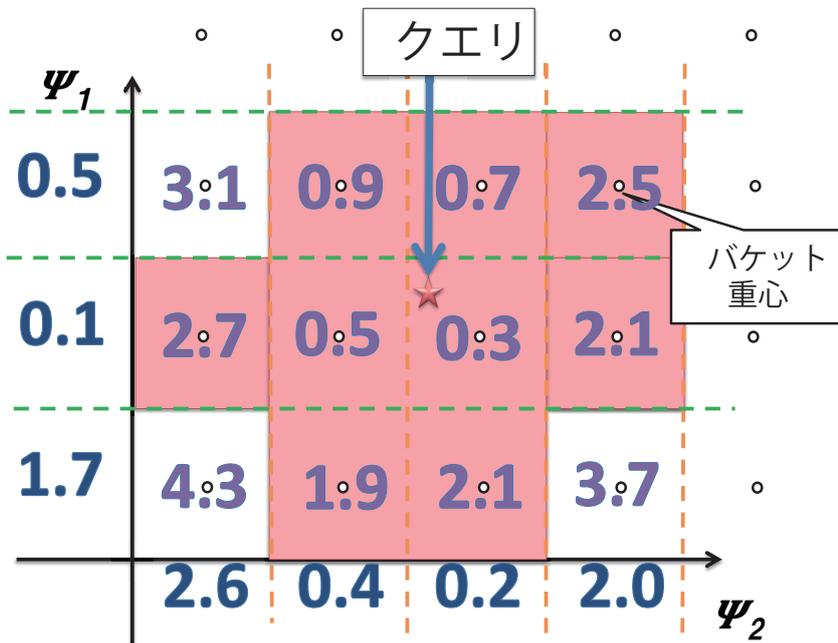
の重心までの概算距離は、各主成分基底に対して、クエリからバケット重心までの二乗距離を計算し、各バケットごとに和をとることで求める。図 3.2(a) はクエリと各バケットへの距離の概算値を示したものである。図 3.2(b) は図 3.2(a) のクエリ周辺を拡大した図であり、主成分基底  $\psi_1$  に対するクエリからバケット重心までの二乗距離が 0.2、主成分基底  $\psi_2$  に対するクエリからバケット重心までの二乗距離が 0.1 である。これより、概算距離は 0.3 となる。しきい値が 3 の場合、図 3.2(a) の赤色で示したクエリ周辺の概算距離が 3 以下の領域となり、これらを探索領域とする。

### 3.2.2 クエリ適応探索

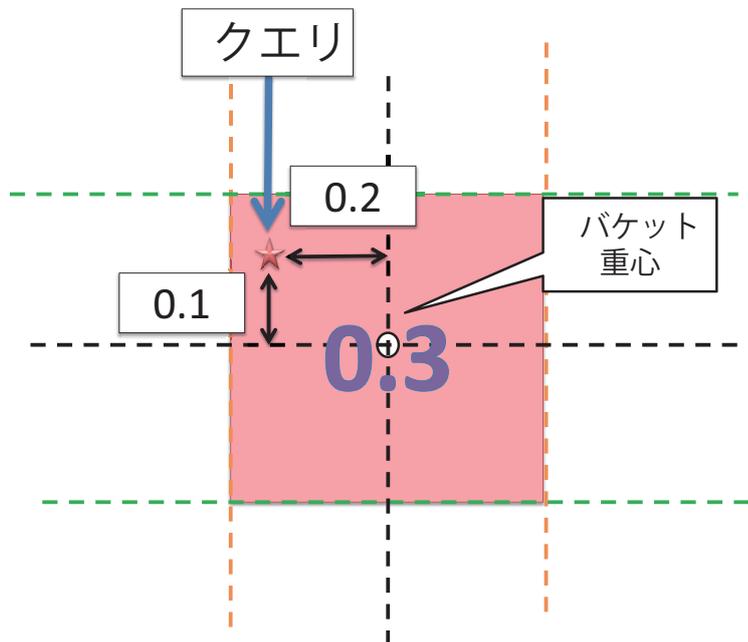
従来のハッシュを用いる手法ではパラメータとして与えられた探索半径に従って探索領域を決定していた。しかし、全てクエリを共通の探索半径で処理すると、クエリ周辺が疎な領域では探索領域が狭すぎて精度が得られず、その一方でクエリ周辺が密な領域では最近傍候補のデータ点数が多すぎて距離計算が過剰になるという現象が起こる。そこで佐藤らの手法では、探索領域の大きさではなく最近傍候補の数に制約を設けることで、クエリごとの処理時間を安定させる。推定距離の小さいところから段階的に探索領域を拡張していき、一定の最近傍候補点を満たした所で探索領域拡張を打ち切る。このアルゴリズムにより、クエリ周辺が疎の場合には図 3.3(a) が示すように、8 つの赤い領域を探索領域とし、広い範囲を探索する。一方、クエリ周辺が密の場合には図 3.3(b) が示すように、4 つの赤い領域を探索範囲とし、狭い範囲を探索することができ、過剰な距離計算を抑えることができる。

### 3.2.3 マッチング処理

特徴点量のマッチング処理において、データベースのすべての特徴量を用いた場合、ある特徴量が他の顔の部位の特徴量と誤対応する可能性がある。そこで、マッチングに用いる特徴点を限定した。特徴点検出を用いて抽出した特徴量を用いる場合は、図 3.4(a) に示すようにすべての特徴量を検索の対象とする。また、図 3.4(b) に示すようにクエリの特徴点との座標の差がしきい値以下の特徴点のみを検索の対象とする方法も検討する。このように、検索対象の領域を限定することで、特徴点の誤対応を減らすことができる。

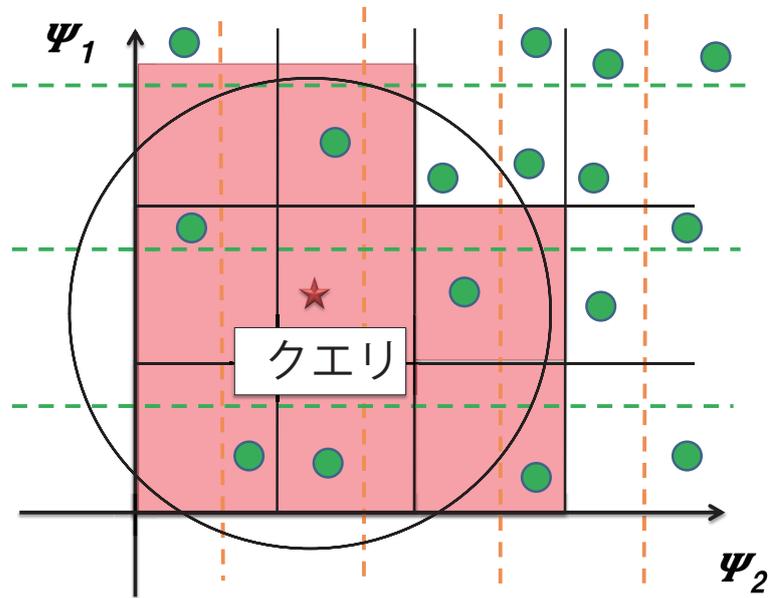


(a) 格子状分割による概算距離

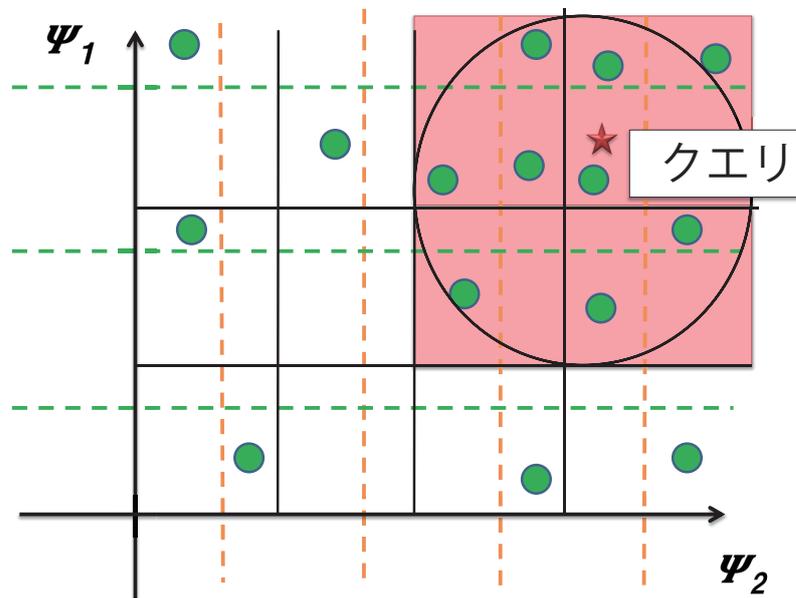


(b) クエリ周辺の拡大

図 3.2: 概算距離



(a) クエリ周辺が疎



(b) クエリ周辺が密

図 3.3: クエリ適応探索

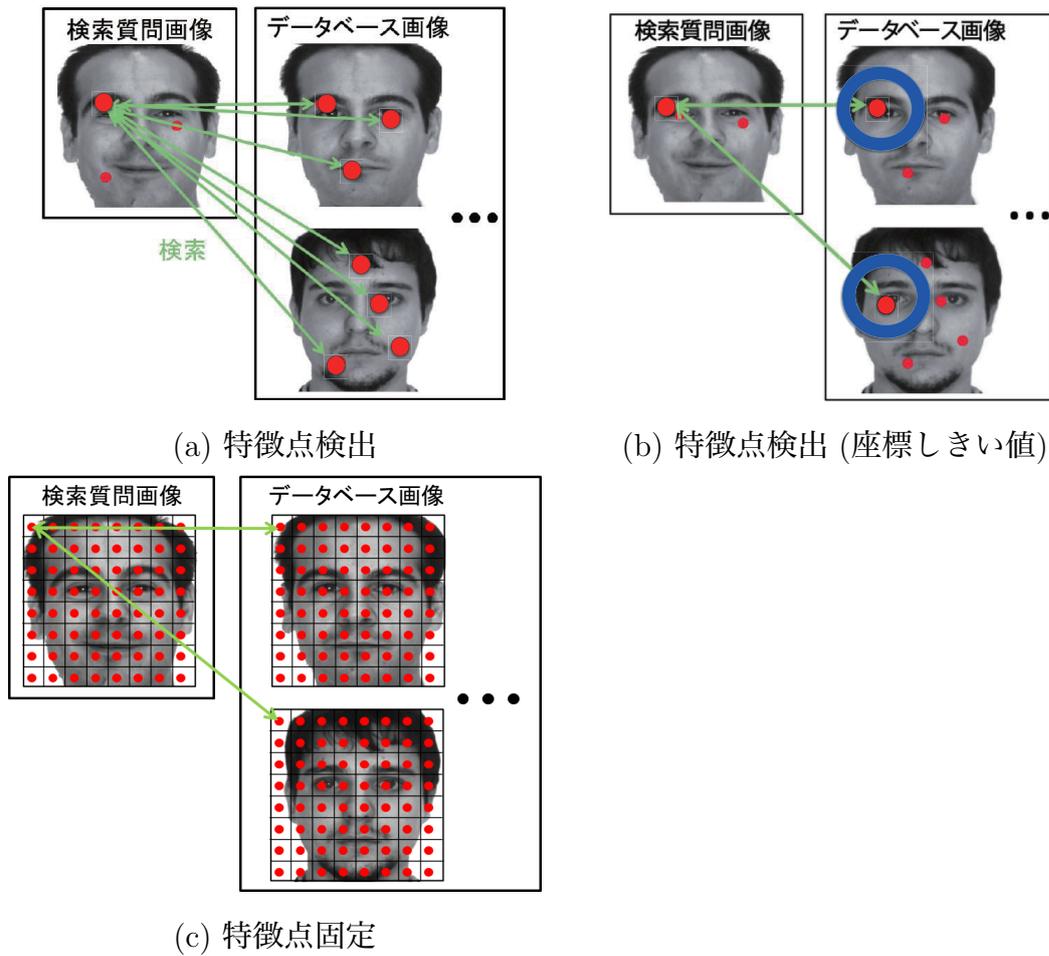


図 3.4: 特徴抽出手法ごとのマッチング処理

一方, Dense Sampling の特徴量の場合は, 図 3.4(c) に示すように検索画像から得た各特徴量に対して, 同じ局所領域から抽出された特徴量のみを検索の対象とする. このように, 検索対象となる特徴点を限定することで, 特徴点の誤対応を減らすことができる.

## 第4章 実験

1000万データベースに対する性能を評価するため実験を行った。また、低解像度のクエリに対する性能の評価も行った。以下に、実験条件、性能評価について詳しく述べる。

### 4.1 実験条件

データベースとして、公開されている顔画像データベースとウェブ上からダウンロードした画像を合わせ、1000万枚の顔画像データベースを作成した。使用した公開されている顔画像データベースは、AR Face Database [4], CAS-PEAL [5], FERET [6], The ORL Database of Faces [7], Georgia Tech Face Database [8], Surveillance Cameras Face Database [9], Indian Face Database [10]である。実験に使用した顔画像は、顔を検出 [11]した後、目や鼻などの顔特徴点14点を自動で検出する [12]。得られた特徴点に基づいて顔の切り出しと顔向き正規化を行う [13]。顔の切り出しや正規化に失敗した画像も含まれている。データベースにはAR Face DatabaseのSet 1の画像のうち、正規化に失敗した2枚の画像を除いた132人132枚の画像が含まれており、正規化後の画像は8 bits グレースケールで、解像度は $512 \times 512$  [pixel]である。

クエリとして、AR Face DatabaseのSet 2–Set 7の画像792枚(132枚 $\times$ 6セット)を用いた。AR Face DatabaseのSet 1–Set 7は図4.1に示すように、neutral, smile, anger, scream, left side light on, right side light on, all side light onの画像で構成されている。犯罪捜査支援を行う際に、自動で犯罪者の候補の絞り込みを行うことを目指すために、正解画像を上位に絞り込むことを目的とし、1000位累積分類率を評価値として用いた。実験に使用した計算機は、CPUがAMD Opteron (tm) Processor 6174 (2.2GHz)、メモリは256GBである。処理時間は特徴量の検索にかかった時間のみを測定し、画像の正規化や特徴抽出の時間は含まない。予備実験により、投票に用いる近傍点を $K = 30$ 点と決め、投票処理を行った。

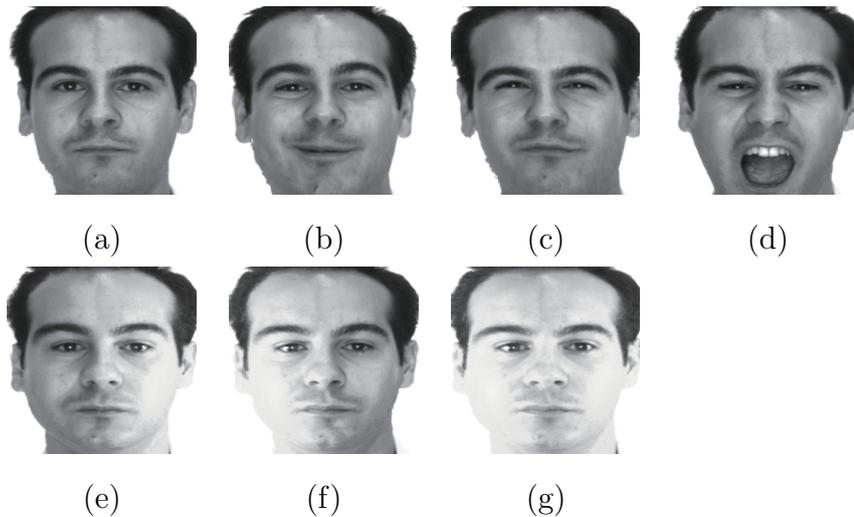


図 4.1: AR Face Database の画像例. (a) Set1: neutral, (b) Set2: smile, (c) Set3: anger, (d) Set4: scream, (e) Set5: left side light on, (f) Set6: right side light on, (g) Set7: all side light on.

## 4.2 1000万データベースに対する評価

近似最近傍探索を行う際の、ハッシュの次元数と探索点数を様々に変化させ、各クエリに対する 1000 位累積分類率の最も高い認識率で最速の処理時間を調べた。AR Face Database のクエリを用いたときの、特徴点検出を用いた場合の、1000 万データベースに対する結果を表 4.1, 表 4.2 に示す。また、パラメータを変化させたときのグラフを図 4.3 に示す。表 4.1, 表 4.2 に示すように、scream と all side light on 以外では、97%以上の高い認識率が得られた。left side light on に対しては、認識率 98.5%, 処理時間 185 ms と高速に認識した。また、特徴点検出により検出された特徴点を用い、座標しきい値を取り入れマッチングした手法 (以下, Detector-coord) と通常の特徴点検出を使用した手法 (Detector) を比較すると、表 4.1, 表 4.2 に示すように、Detector-coord は Detector より高い認識率が得られた。これは、Detector-coord はクエリの特徴点との座標の差がしきい値以下の特徴点のみを検索の対象とするので、検索対象の領域を限定することで、特徴点の誤対応を減らすことができたからだと考えられる。smile, anger では 1000 万顔画像データベースであっても 98%以上の認識率が得られたが、scream, all side light on では認識率が低かった。この原因として、scream では、図 4.2(a), (b) に示すように、クエリの表情変化が大きく、得られた特徴量が対応するデータベース画像の特徴量から大きく変化してしまった

表 4.1: 1000 万顔画像データベースに対する認識率と処理時間 (特徴点検出)

クエリの種類	smile	anger	scream	left side light on	right side light on	all side light on
認識率 [%]	95.5	99.2	30.3	98.5	97.0	21.2
処理時間 [ms]	191	180	120	216	195	247

表 4.2: 1000 万顔画像データベースに対する認識率と処理時間 (特徴点検出:しきい値あり)

クエリの種類	smile	anger	scream	left side light on	right side light on	all side light on
認識率 [%]	98.5	99.2	39.4	98.5	97.7	28.8
処理時間 [ms]	193	106	230	185	228	214

こと, all side light on では, 図 4.2(a), (c) に示すように, 照明により画像全体が白くなっており, 得られた特徴点の数が減少したことが挙げられる.

### 4.3 データベースの枚数と処理時間の評価

また, データベースの枚数と処理時間の関係を調べた. データベースの枚数を 10 万枚, 20 万枚, 50 万枚, 100 万枚, 200 万枚, 500 万枚, 700 万枚, 1000 万枚まで変化させた処理時間を測定した. 特徴点マッチング手法として Detector-coord を用い, AR Face Database の left side light on に対して 1000 位累積分類率で 98.5%以上の認識率で最速の処理時間を測定した. データベースの枚数を変化させたときの処理時間をプロットした結果を図 4.4 に示す. 図 4.4 に示すように, 700 万枚まではデータベースの数が増加しても, 処理時間はあまり増えなかったが 1000 万枚になると処理時間が大幅に増えた. この原因としては, 近似最近傍探索のパラメータを最適に選べなかったことが考えられる. しかし, 1000 万枚という大規模なデータベースでも 200 ms と高速に認識できたことがわかる.

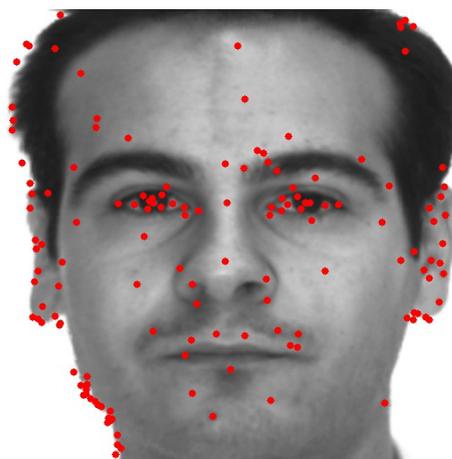
### 4.4 低解像度のクエリに対する評価

実際に監視カメラで撮影した画像を用いて性能評価を行った. 高解像度のデータベース画像と低解像度のクエリ画像を各 15 人分撮影した. データベース画像は目の前に Logicool 社製 C270 を置き正面向きの顔を撮影した. 正規化前の解像度は  $640 \times 480$ [pixel] である.

表 4.3: 認識率と処理時間

マッチング処理	特徴点検出	特徴点固定
認識率 [%]	73.3	93.3
処理時間 [ms]	112	43

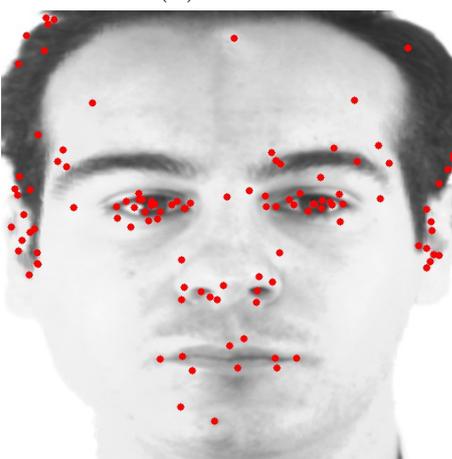
クエリ画像は高さ 2.8[m] に *Panasonic* 社製 *BB-HCM715* を置き、正面向きの顔を撮影した。正規化前の解像度は  $1280 \times 960$ [pixel] である。データベース画像とクエリ画像の例を図 4.5 に示す。データベース画像は、撮影した 15 枚に加えて 1000 万データベースの一部を追加し 1 万枚のデータベースを用いた。特徴点に Dense Sampling と特徴点検出を用いて実験を行った結果を表 4.3 に示す。監視カメラで撮影した画像では、解像度が低く、特徴点が検出されにくい。特徴点検出で得られた特徴点の数は 1 枚あたり平均 104 点であり、Dense Sampling で得られた特徴点の数である 880 点よりも大幅に少ない。そのため、表 4.3 に示すように、特徴点の数を固定する Dense Sampling を用いた方が高い認識率を示した。



(a) Neutral



(b) scream



(b) all side light on

図 4.2: 特徴点の例

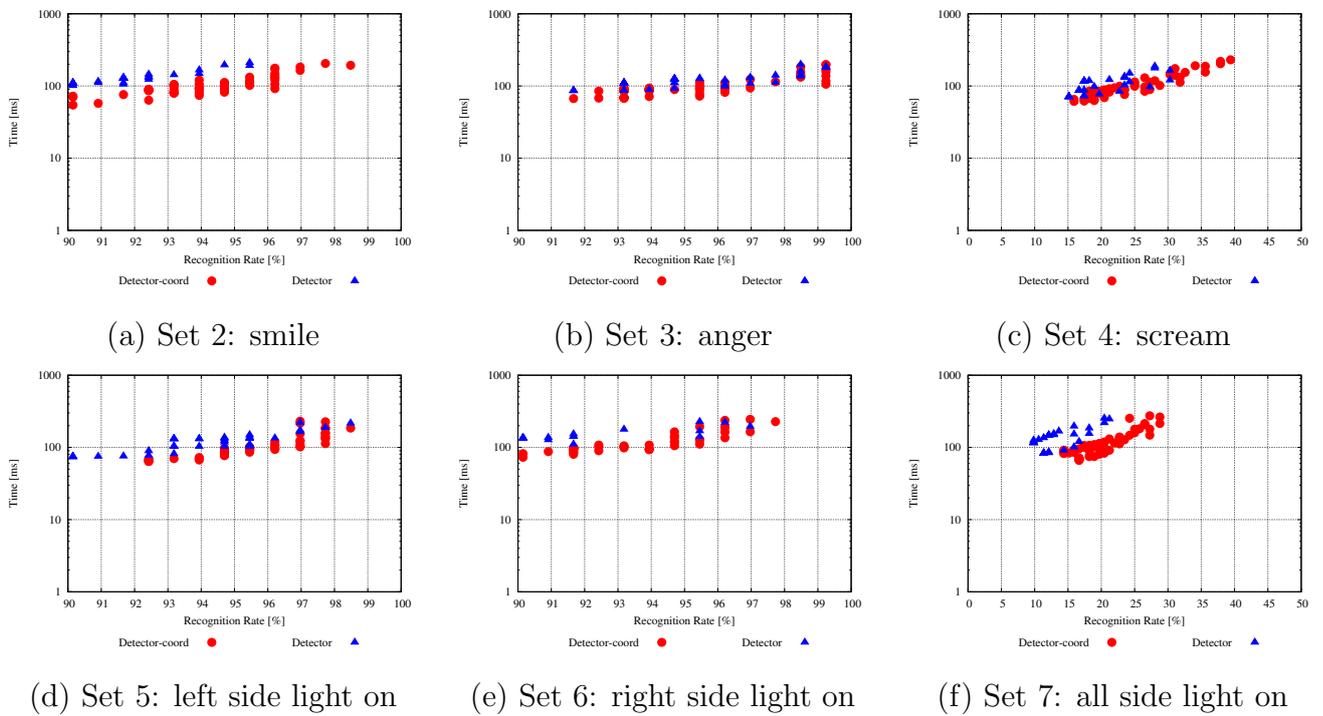


図 4.3: 1000 万顔画像データベースに対する認識率と処理時間

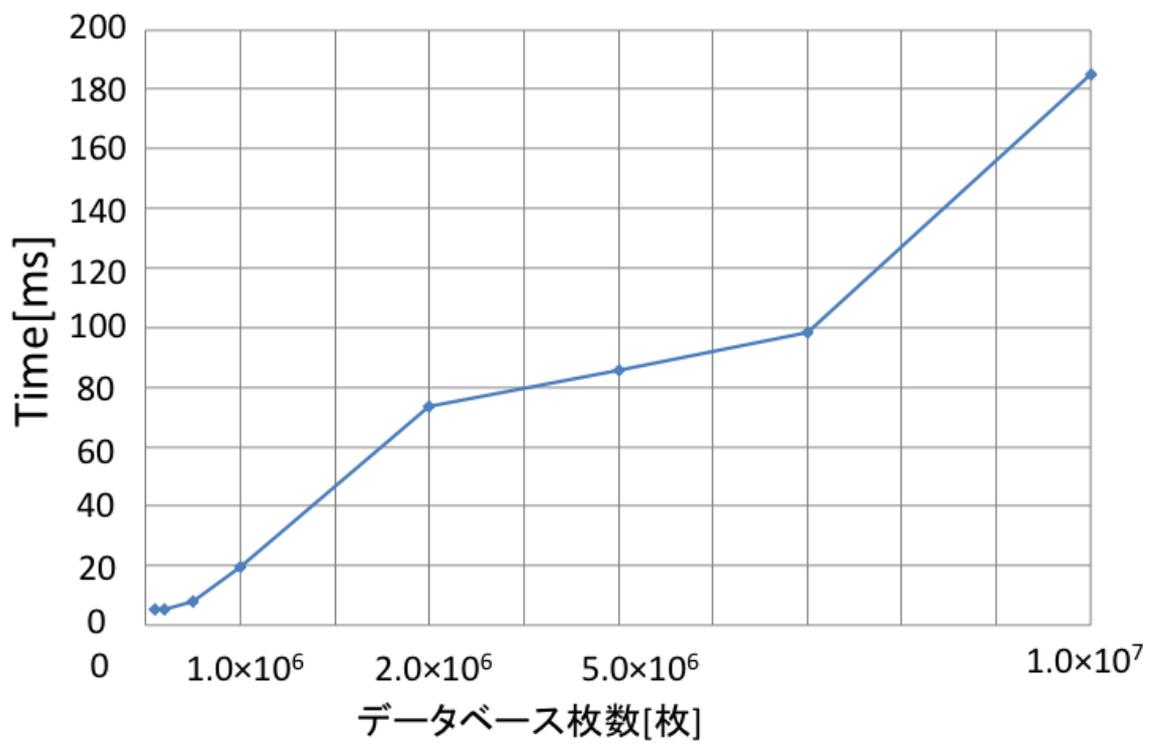
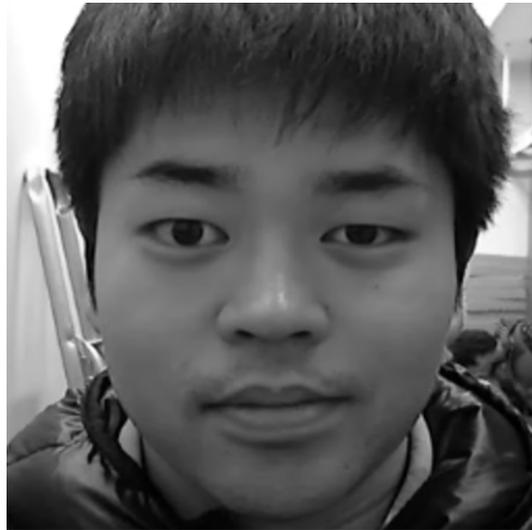


図 4.4: データベースの枚数と計算時間



(a) データベース画像



(b) クエリ画像

図 4.5: 低解像度のクエリを用いた実験の画像の例

## 第5章 結論

本論文では、犯罪捜査支援を目的とし、大規模データベースに対して高速・高精度で認識できること、クエリに低解像度画像を用いた場合でも高精度で認識できることを目指した。1000万枚の顔画像データベースを用いた実験の結果、AR Face Databaseのleft side light onに対して認識精度が98.5%で185msで認識が可能であった。また、特徴点マッチング手法のDetectorとDetector-coordを比較すると、Detector-coordはDetectorに比べ認識率が高く、高速に認識することができた。実際に監視カメラから得られた画像を用いて認識を行った結果、1万枚の顔画像データベースに対して、認識精度が93.3%、処理時間43msで認識が可能であった。また、Dense Samplingで一定の特徴量を得ることで低解像度画像に対しても高精度に認識を行うことが可能であった。今後の課題として、更なる認識率の向上のため、照明条件の変動に対応することが挙げられる。



## 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、岩田基助教に深く感謝いたします。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、加藤祐也氏をはじめ、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いたします。

2013年3月8日



## 参考文献

- [1] Y. Ke and R. Sukthankar, "Pca-sift: A more distinctive representation for local image descriptors," CVPR'04, vol.2, pp.506–513, 2004.
- [2] J. Luo, Y. Ma, E. Takikawa, S. Lao, M. Kawade, and B.L. Lu, "Person-specific SIFT features for face recognition," ICASSP 2007, vol.2, pp.593–596, 2007.
- [3] 佐藤智一, 岩村雅一, 黄瀬浩一, "概算距離の精度向上による近似最近傍探索の高速化," 電子情報通信学会技術研究報告, pp.61–66, 2011.
- [4] A. Martínez and R. Benavente, "The AR face database," Technical Report 24, Computer Vision Center, Bellatera, 1998.
- [5] W. Gao, B. Cao, S. Shan, D. Zhou, X. Zhang, D. Zhao, and S.S.E. Al, "The CAS-PEAL large-scale Chinese face database and evaluation protocols," Technical report, Joint Research & Development Laboratory, CAS, 2004.
- [6] J.P. Phillips, H. Moon, S.A. Rizvi, and P.J. Rauss, "The feret evaluation methodology for face-recognition algorithms," IEEE Trans. on PAMI, vol.22, no.10, pp.1090–1104, 2000.
- [7] "ORL Face Database," <http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>
- [8] "Georgea Tech Face Database," [http://www.anefian.com/research/face\\_reco.htm](http://www.anefian.com/research/face_reco.htm)
- [9] M. Grgic, K. Delac, and S. Grgic, "Scface - surveillance cameras face database," Multimedia Tools and Applications, pp.1–17, 2009.

- 
- [10] V. Jain and A. Mukherjee, “The indian face database,” 2002. <http://vis-www.cs.umass.edu/~vidit/IndianFaceDatabase/>
- [11] T. Mita, T. Kaneko, B. Stenger, and O. Hori, “Discriminative feature co-occurrence selection for object detection,” *IEEE Trans. on PAMI*, vol.30, pp.1257–1269, 2008.
- [12] M. Yuasa, T. Kozakaya, and O. Yamaguchi, “An efficient 3D geometrical consistency criterion for detection of a set of facial feature points,” *IEICE - Trans. Inf. Syst.*, vol.E91-D, pp.1871–1877, 2008.
- [13] T. Kozakaya and O. Yamaguchi, “Face recognition by projection-based 3d normalization and shading subspace orthogonalization,” *FGR '06*, pp.163–168, 2006.