

## 近似最近傍探索の多段階化による高速特定物認識

野口 和人<sup>†</sup>      黄瀬 浩一<sup>†a)</sup>      岩村 雅一<sup>†b)</sup>

Efficient Recognition of Specific Objects by Cascading Approximate Nearest Neighbor Searchers

Kazuto NOGUCHI<sup>†</sup>, Koichi KISE<sup>†a)</sup>, and Masakazu IWAMURA<sup>†b)</sup>

あらまし SIFT などの局所特徴量の最近傍探索によって特定物体認識を行う場合には、1 画像当りの特徴ベクトルの数が膨大になるため、最近傍探索の効率が重要となる。本論文では、「正しい認識を可能とする最近傍探索の精度は画像によって異なる」という観点から処理を削減した効率的認識法を提案する。具体的には、近似最近傍探索に基づく識別器を多段階に縦列接続することにより、認識に用いる近似の程度を画像に応じて変更し、大幅な効率化を実現する。1 万枚の平面物体のデータベースを用いた実験の結果、処理時間を、多段階化を行わなかった場合の約 1/10、ANN や LSH を近似最近傍探索の手法として用いた場合の約 1/40 に削減でき、例えば、認識率 98%、処理時間 1 ms/query を達成できることが分かった。また、8.6%のリジェクトを行うことによって、誤認識率を 0%に抑えることもできた。更に、10 万画像を用いた実験から、高いスケーラビリティがあることも分かった。

キーワード 物体認識, PCA-SIFT, 近似最近傍探索, 多段階化, リジェクト

### 1. ま え が き

デジタルカメラやカメラ付き携帯電話の普及に伴って、カメラを情報入力機器としても利用したいという要望が高まっている。典型的な利用法としては、ポスターなどのインスタンス（特定物体）を撮影して関連情報にアクセスすることが考えられる。このためには、画像中の特定物体を認識する必要がある。特に本論文では、ポスターなどの平面物体を対象とした特定物体認識（以後、単に物体認識と呼ぶ）に焦点を当てる。

物体認識の一手法として、SIFT (Scale-Invariant Feature Transform) [1] などによって得られる局所特徴量の最近傍探索によるものがある。この手法には、特徴ベクトルが局所的に得られるために、隠れや変動に対して比較的ロバストであるという利点がある。一方、得られる特徴ベクトルの次元が数十から数百、そ

の数が画像当り数百から数千と膨大なため、最近傍探索に必要な時間が膨大になるという問題点もある。

この問題は、最近傍探索の効率を向上させることによって解決可能である。一つのアプローチとして、最近傍探索に近似を導入し、効率化を図るものがある [2], [3]。これを物体認識に導入すると、効率を大幅に向上させることができる [4]。

近似最近傍探索を用いた物体認識では、近似の程度が認識率と効率をバランスするための重要なパラメータとなる。大幅な近似をすればするほど処理時間を削減できるが、近似をしすぎると多くの特徴ベクトルに対して最近傍が求まらなくなり、結果として誤認識を引き起こしてしまう。ここでの問題は、誤認識を引き起こす近似の程度が画像によって異なる点である。大幅な近似を行っても認識できる「簡単な」画像がある反面、それでは誤認識となる「難しい」画像もある。固定的な近似によって一定の認識率を確保するには、近似の程度を認識の難しい画像に合わせる必要があり、効率向上の妨げとなっている。

そこで本論文では、近似の程度を画像に対して適応的に調節する効率的物体認識法を提案する。本手法では、近似の程度が異なる識別器を複数用意し、それら

<sup>†</sup> 大阪府立大学大学院工学研究科, 堺市  
Graduate School of Engineering, Osaka Prefecture University, 1-1 Gakuencho, Naka-ku, Sakai-shi, 599-8531 Japan  
a) E-mail: kise@cs.osakafu-u.ac.jp  
b) E-mail: masa@cs.osakafu-u.ac.jp

を大幅な近似のものからあまり近似のないものへと多段階に縦列接続する．これによって、簡単に認識できる画像は、前段の部分で大幅な近似の識別器によって高速に認識し、それでは認識できない画像に対してのみ、後段の部分で近似の少ない識別器によって時間をかけて精密に認識する．本手法の特徴は、多段階化する識別器の構成方法にある．後段の識別器では、近似の違いによる差分のみ、すなわち、それより前段の識別器で対象とならなかった特徴ベクトルのみを距離計算の対象とすることによって、最後段まで処理が進んでも、最後段の識別器を単独で用いる場合とほぼ同等の計算量しかかからないという利点が得られる．

提案手法の性能を評価するため、1万画像のデータベースを用いて実験を行い、認識率98%を得るためにかかる処理時間が、ANN, LSHといった従来の近似最近傍探索手法を用いた場合と比較し、大幅に削減できることを示す．更に、10万画像を用いた実験から、高いスケーラビリティがあることも示す．

## 2. 最近傍探索に基づく物体認識

画像データベースに多数の画像が収められており、各々の画像は一つの物体を表すものとする．認識対象の画像（以下、検索質問と呼ぶ）が与えられたとき、物体認識のタスクを、検索質問と同じ物体が写った画像をデータベースから検索することと定義する．本論文では、このようなタスクを最近傍探索によって達成する方法について考える．

最近傍探索を用いて物体認識を行う手法は、一つの画像を一つの特徴ベクトル（ここでは、大域特徴量と呼ぶ）で表現する方法と、一つの画像を複数の特徴ベクトル（同様に、局所特徴量と呼ぶ）で表現する方法に分類できる．以下では、検索質問の画像を  $Q$ 、データベース中の  $l$  番目の画像を  $I_l$  と表す．また、 $Q$  から得た特徴ベクトルを  $q_1, \dots, q_m$ 、 $I_l$  から得た特徴ベクトルを  $p_{l1}, \dots, p_{ln_l}$  と表す．

まず、大域特徴量 ( $m = 1, \forall l, n_l = 1$ ) を用いて認識する方法について述べる．この方法では、 $q_1$  に対して、データベースから最近傍を求め、 $p_{l1}$  が得られたとすると、画像  $I_l$  を回答とする．このような場合の最近傍探索法として、荒井らはデータの分布の偏りを利用して空間を分割する主成分木を提案している [5]．主成分木では、高次元のベクトルに対して、距離計算対象の絞込みと距離計算の打ち切りを効率良く行えるように空間を分割することによって処理を削減している．

大域特徴量の場合は、最近傍探索の精度が物体の認識率に直結するため、荒井らは近似最近傍探索ではなく、最近傍探索を行っている．

一方、局所特徴量 ( $1 < i \leq m, 1 < j \leq n_l$ ) を用いて認識する方法では、検索質問の特徴ベクトルが複数あるため、各々の最近傍探索の結果を統合しなければならない．最も単純な手法として、次の投票方式がある． $q_i$  に対してデータベース中から最近傍を求めて  $p_{ij}$  が得られたとすると、画像  $I_l$  に1票を投じる．このような投票を  $Q$  から得られた  $m$  個の特徴ベクトルに対して実行し、最終的に得票数が最大となった画像を認識結果として回答する．本論文では、 $m, n_l$  は数百から数千の場合を考えるため、処理が膨大なものになる．

投票方式では、正解の物体と他の物体の得票数が逆転しなければ問題は生じない．そのため、大域特徴量の場合と比べ、正確に最近傍を求める必要はない．このような観点から距離計算の対象の数を削減する方法として、vocabulary tree という手法が提案されている [6]．この手法では、特徴ベクトルをそのまま扱うのではなく、vocabulary tree と呼ばれる木構造を用いて量子化して扱う．これにより、大量の visual word を効率良く扱うことができ、処理が大幅に軽減される．しかし、量子化の影響により、認識率は最近傍探索を行う方法に比べ劣ることが多い．そのため本研究では、近似最近傍探索を用いて、認識率をあまり落とすことなく高速に認識することを考える．

## 3. 近似最近傍探索の従来法

提案手法の詳細について述べる前に、近似最近傍探索の従来法についてまとめておく．ここで取り上げる従来法は、比較実験に用いるものである．

最近傍探索で最も時間がかかるのは距離計算である．最近傍探索の高速化には、個々の距離計算自体を高速に行う方法と、距離計算の対象を効率良く絞り込む方法の2通りがある．近似最近傍探索では、主に後者の絞込みを大胆に行うことで、処理時間を削減する．ところがその代償として、最近傍が距離計算の対象から外れてしまい、求まらない可能性がある．どの程度の近似が適切であるのかは、対象とするタスクに依存するため、近似最近傍探索の手法では、近似の程度を調節するためのパラメータを設けている．以下では、近似最近傍探索の代表的な手法として ANN と LSH について述べる．

ANN (Approximate Nearest Neighbor) [2] は、二分木を用いて近似最近傍探索を高速に行う手法である。木のノードは、特徴空間を分割した hyperrectangle に対応しており、葉ノードには単一の特徴ベクトルが対応づけられている。ANN では、木構造の探索によって距離計算の対象となる特徴ベクトルを収集し、その中で距離が最短のものを近似最近傍探索の結果として出力する。ANN には近似の程度を表すパラメータとして許容誤差  $\epsilon$  がある。 $\epsilon$  が大きければ、より大幅な近似を行って対象となる特徴ベクトルを絞り込むため、処理時間が短縮できる。

LSH (Locality-Sensitive Hashing) は、ハッシュ表を用いた近似最近傍探索の手法である [3]。LSH では、一つの特徴ベクトルを  $L$  通りの  $k$  次元ベクトルに変換し、各々に対応する  $L$  個のハッシュ表に登録する。検索時には、検索質問の特徴ベクトル  $q$  を  $L$  通りの  $k$  次元ベクトルに変換し、すべてのハッシュ表を検索する。そして得られた特徴ベクトルの集合  $P$  の中から  $q$  との距離が最小のものを結果とする。LSH では、 $k$  が一定ならば、 $L$  が小さいほど処理時間を短縮できるものの、真の最近傍も得られにくくなる。

## 4. 多段階化

### 4.1 構成と要件

局所特徴量を用いて画像を表現し、特徴ベクトルの近似最近傍探索と投票によって認識する場合、性能の限界は近似を行わない場合の認識率である。近似を行えばそれだけ高速化が実現できるが、一般に認識率は低下する。先に述べたように、このような近似の程度と認識率の関係は、認識対象の画像によって異なるため、認識率を保ちつつ処理時間を短縮するには、近似の程度を適応的に調整する必要がある。

問題は、正しい認識が可能となる近似の程度を、認識の前に推定することが容易ではない点である。この問題に対処する一手法は、近似の程度が異なる多数の識別器を用意して、それらの識別器の出力を見ながら、適切なものを選択することであろう。

処理効率を保ちつつ複数の識別器を利用する方法としては、図 1 に示す多段階化が考えられる。ここで、1 から  $N$  の数字が付けられた方形は識別器を表し、数字が若いほど大幅な近似を行うものとする。検索質問の画像から得た特徴ベクトルの集合は、まず 1 段目の識別器によって認識に用いられる。具体的には検索質問画像から得た  $m$  個の局所特徴量すべてを用いて投

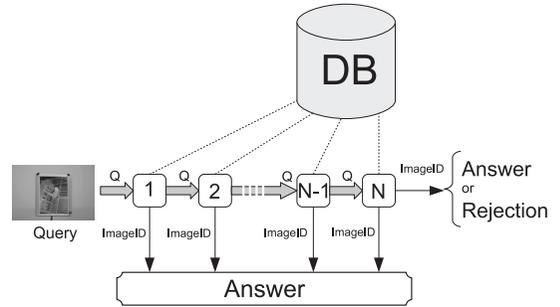


図 1 多段階化による認識

Fig. 1 Recognition by cascaded recognizers.

票を行う。この段階で十分な証拠が得られれば、認識処理を打ち切って結果を回答する。一方、証拠が不十分ならば、特徴ベクトルの集合を、次段の、より小幅な近似の識別器にかけて再度認識する。すなわち、同じ  $m$  個の局所特徴量に対して、近似の程度が異なる識別器を用いて認識処理を施す。このような認識処理を繰り返して最後の  $N$  段まで到達しても十分な証拠が得られない場合には、最大得票数のものを回答するか、その画像についてはリジェクトするかのいずれかをとる。以上の処理によって、早い段階で処理が打ち切られる画像については大幅な効率化が期待できるとともに、必要に応じて時間をかけた認識が可能となる。

このような構成を採用する場合、要件となる事項は、

(1) 認識処理打ち切りの判定方法

(2) 「難しい」画像に対しても処理効率を保つ方法の 2 点である。(1) については、少ない計算量で、なるべく正確に判定することが望まれる。(2) は、後段まで認識処理を繰り返す画像についても、処理効率を低下させないための方策である。理想的には、多段階化した識別器で  $s$  段まで処理が進んだ場合の計算量が、 $s$  段目と同じ近似の識別器を単独で使った場合と同等であればよい。以下、各々について述べる。

### 4.2 認識処理打ち切りの判定方法

認識誤りを引き起こす画像には、そもそも得票数が少ない、得票数がある程度得られる場合でも第 2 位の候補と得票数に開きがほとんどない、という性質がある。これらの点に着目すると、得票数を用いた次のような簡便な判定方法が考えられる。1 位の得票数を  $V_1$ 、2 位の得票数を  $V_2$  とすると、 $V_1 > t$ 、 $rV_1 > V_2$  を同時に満たすならば、処理を打ち切って 1 位得票の画像を回答とする。ここで、 $t$  は得票数のしきい値、 $r$  は 1 位と 2 位の得票数の比のしきい値である。

4.3 「難しい」画像に対しても処理効率を保つ方法  
 近似の程度が異なる  $N$  個の近似最近傍探索器  
 $1, \dots, N$  (以後, 単に探索器と呼ぶ) を考える. 探索器  
 $(s-1)$  の方が探索器  $s$  よりも大幅な近似を行うとす  
 る. 探索器  $s$  を用いて, 特徴ベクトル  $q_i$  に対して近  
 似最近傍探索を行う際に, 距離計算の対象として得ら  
 れる特徴ベクトルの集合を  $P_i^{(s)}$  とする. 近似最近傍  
 探索では, 通常, 大幅な近似であるほど, 距離計算の  
 対象となる特徴ベクトル数が少ないという性質がある.  
 すなわち, すべての  $i$  と  $s$  に対して  $|P_i^{(s)}| \geq |P_i^{(s-1)}|$   
 が成り立つ.

今, これらの探索器に対して, 次の二つの性質を考  
 える.

[定義 1] (単調性) すべての  $i$  と  $s$  について,

$$P_i^{(s)} \supseteq P_i^{(s-1)} \quad (1)$$

が成り立つとき, 近似最近傍探索器には単調性がある  
 という.

[定義 2] (差分検索性) 近似最近傍探索器が差集合

$$P_i^{(s)} - P_i^{(s-1)} \quad (2)$$

を効率的に求められるとき, 差分検索性があるという.

単調性をもつ探索器を用いて, 図 1 の多段階識別器  
 を構成する場合,  $s$  段目では,  $P_i^{(s)}$  ではなく, 前段と  
 の差分  $P_i^{(s)} - P_i^{(s-1)}$  を距離計算の対象とすることが  
 考えられる. このように構成すると, 1 段目から  $s$   
 段目までで距離計算の対象となった特徴ベクトルの和集  
 合は, 探索器  $s$  を単独で用いた際の集合  $P_i^{(s)}$  と等し  
 くなるため, 距離計算の回数は同一となる. 更に, 探  
 索器が差分探索性をもつ場合には, 多段階化を行って  
 も計算量の増加を低く抑えることができる.

認識のプロセスを図 1 に沿って具体的に述べる. 処  
 理が  $(s-1)$  段目まで進んでいるときには, 各特徴ベ  
 クトル  $q_i$  に対する暫定最近傍  $\hat{p}_*$  が見つかっている.  
 したがって,  $s$  段目では,  $p \in (P_i^{(s)} - P_i^{(s-1)})$  に対  
 してのみ距離計算を行い,  $\hat{p}_*$  よりも距離の近いベク  
 トルが見つければ, それを新たに暫定最近傍  $\hat{p}_*$  として,  
 投票をやり直せばよい.

## 5. 提案手法

以上の概念に基づく手法を提案する. 提案手法で  
 は, 局所特徴量として PCA-SIFT [7] を用いる. PCA-  
 SIFT を用いる場合, ANN や LSH を用いた認識より

も高速な手法として, 野口らの手法 [4] がある. この  
 手法は, 後述するように単調性並びに差分検索性をも  
 つため, 多段階化にも好都合である<sup>(注1)</sup>. そこで提案  
 手法では, この既存手法を多段階化の識別器として用  
 いて, 物体認識法を新規に構成する.

野口らの手法には, 近似最近傍探索の最終段階で距  
 離計算を行う手法 (距離計算ありの手法) に加えて,  
 距離計算を全く行わずに済ませる手法 (距離計算なし  
 の手法) がある. 距離計算なしの手法は, 距離計算あ  
 りの手法と比較し認識率は劣るものの, 特徴ベクトル  
 をメモリ上に保持しておく必要がないため, 省メモリ  
 であるという利点がある. 以下では, まず, 距離計算  
 ありの手法, 距離計算なしの手法に共通のデータ登録  
 について述べたあと, 各々の手法, 多段階化の方法に  
 ついて述べる.

### 5.1 データの登録

LSH では, 用いるハッシュ表の数が増えると大量  
 のメモリを消費する. そこで野口らの手法では, メモ  
 リ量を削減するため, ハッシュ表を一つだけ使う. ハッ  
 シュ表に特徴ベクトルを登録する方法は次のとおりで  
 ある. PCA-SIFT によって得られた  $d'$  次元のベクト  
 ル  $p = (p_1, \dots, p_{d'})$  の第 1 次元から第  $d$  次元 ( $d \leq d'$ )  
 までを用い,

$$u_j = \begin{cases} 1 & \text{if } p_j \geq 0, \\ 0 & \text{otherwise,} \end{cases}$$

によって各次元を 2 値化し, ビットベクトル  $u =$   
 $(u_1, \dots, u_d)$  を作成する. そして,

$$H_{\text{index}} = \left( \sum_{i=1}^d u_i 2^{(i-1)} \right) \bmod H_{\text{size}} \quad (3)$$

によってハッシュのインデックスを求め, 画像 ID を  
 ハッシュ表に登録する. ここで  $H_{\text{size}}$  は, ハッシュ表の  
 サイズである. 距離を用いる場合には, 画像 ID のほ  
 か,  $p$  そのものを登録し, 検索時の距離計算に用いる.

登録時に衝突が生じた場合は, チェイン法により複  
 数の画像 ID をリストとして登録する. このとき, リ  
 ストが長くなりすぎると, 距離計算のコストがかかり  
 すぎるといった問題が生じる. 野口らの手法では, リス  
 ト長  $n$  に対するしきい値  $c$  を設け,  $n > c$  を満たすと

(注1): 単調性は ANN や LSH でも満たされる. ANN では  $\epsilon$  の値を  
 段階的に変更し, LSH では探索するハッシュ表の数  $L$  を段階的に増や  
 せばよい.

リスト全体をハッシュ表から削除する。同じハッシュ値をもつ特徴ベクトルが多いということは、その特徴ベクトルが画像の識別にあまり寄与しないことを意味する。したがって、削除をしても影響は比較的少ないと考えられる。

以上の処理を、データベースに登録するすべての特徴ベクトルに対して施すことにより、データの登録は完了する。

## 5.2 検 索

### 5.2.1 距離計算ありの手法

次に距離計算ありの手法の検索について述べる。この手法では、検索質問  $Q$  から得た各特徴ベクトル  $q_i$  に対して、登録時と同様にインデックスを計算することで、ハッシュ表から特徴ベクトルを検索する。得られた特徴ベクトルの集合を  $P_i$  とすると、次に  $P_i$  の中から  $q_i$  の最近傍となる特徴ベクトル  $p_*$  を求める。そして、 $p_*$  に対応する画像 ID に投票する。

検索質問画像とデータベースの画像が同じではないため、特徴ベクトルの各次元の値は変動したものになる。このとき、異なったビットベクトルに変換されると、検索が不可能になるという問題が生じる。野口らの手法では、値の許容変動幅  $e$  をパラメータとして、変動への対処を施す。具体的には、 $q = (q_1, \dots, q_d)$  とするとき、 $|q_j| \leq e$  を満たす次元  $j$  に対しては、 $u_j$  だけではなく  $u'_j = 1 - u_j$  も用いて特徴ベクトルを検索する。例えば、ビットベクトル  $u = (0, 0, 0, 1)$  の第二次元がこの処理の対象となったとすると、 $u' = (0, 1, 0, 1)$  も用いて検索する。ただし、このような「両方試す」という処理を制限なく導入すると、膨大な計算時間が必要となってしまうため、 $u'_j$  を適用する次元数を一定値  $b$  以下に制限する。これによって、最も多い場合でも、用いるビットベクトルの数を  $2^b$  に抑える。 $|q_j| \leq e$  を満たす次元の数が  $b$  を上回るときには、次元のインデックスが大きいものから  $b$  個を採用する<sup>(注2)</sup>。

### 5.2.2 距離計算なしの手法

距離計算なしの手法では、検索質問の特徴ベクトル  $q$  に対して上記のような距離計算を施して近似最近傍を求めるのではなく、ハッシュ表から得た特徴ベクトルの集合  $P$  に属するすべての特徴ベクトルに対して投票処理を施す。処理のパラメータは、距離を用いない方法と同様、特徴量の許容変動幅  $e$ 、変動に対処する次元の数  $b$  の二つである。

### 5.2.3 $b$ による多段階化

野口らの手法のパラメータは次の四つである。二つ

はハッシュのパラメータであり、ハッシュ表のリスト長の上限  $c$  とハッシュ関数に使う次元数  $d$  である。残り二つは検索のパラメータであり、ビットベクトルで反転の操作を行う次元数  $b$  と値の変動幅  $e$  である。提案手法では、このうち  $b$  を変更することで近似の程度を調整する。具体的には、第  $s$  段では  $b = s - 1$  とした識別器を用いる。野口らの手法は、 $b$  の増加に伴ってハッシュ表のアクセスに用いるインデックスが増加するだけである。そのため、単調性だけではなく差分探索性も満たす。

なお、距離計算なしの手法では、各段の処理で、暫定最近傍  $\hat{p}_*$  を更新しつつ投票するのではなく、式 (2) の差集合に属する特徴ベクトルすべてに対して投票する。ここで、 $P_i^{(s)}$  は、検索質問の特徴ベクトル  $q_i$  に対して、 $s$  段まででハッシュ表から検索される特徴ベクトルの集合である。

## 6. 実 験

### 6.1 実験条件

実験では、近似最近傍探索の従来法である ANN, LSH<sup>(注3)</sup>に加え、野口らの手法を用いて識別器を構成し、提案手法(野口らの手法の多段階化版)と比較した。提案手法並びに野口らの手法で用いている処理のパラメータを表 1 に示す。

実験には以下に述べる画像データベース、検索質問画像を用いた。局所特徴量としては、PCA-SIFT のサイト<sup>(注4)</sup>で提供されるものを用いた。ハッシュ表のサイズは  $H_{\text{size}} = 2^d$  とした。以下に示す処理時間は、検索質問の画像 1 枚当りの認識に要した時間を表す。ただし、特徴ベクトルの抽出に必要な時間は含めない。使用計算機は、CPU が AMD Opteron 2.8 GHz、メモリ 32 GByte のものである。

#### 6.1.1 画像データベース

実験に用いた画像について説明する。まず、収集方法の異なる A, B, C の 3 種類のデータセットを準備した。A は、Google のイメージ検索を用いて収集した 3,100 枚の画像である。検索キーワードとしては、

(注2): 次元のインデックスが大きいものから選ぶ理由は、特徴ベクトルが主成分分析により得られるためである。次元のインデックスの大きいものは固有値の小さい成分に対応する。また、 $|q_j|$  が小さいものから選ぶという方策も考えられるが、ソートにかかるコストのため、現在のデータベースでは効果が十分発揮できず、結果的に性能が低下する。

(注3): ANN としては <http://www.cs.umd.edu/~mount/ANN/>, LSH としては <http://www.mit.edu/~andoni/> で提供されているソースコードを用いた。

(注4): <http://www.cs.cmu.edu/~yke/pcasift/>

表 1 パラメータと意味．上段は提案手法と野口らの手法に共通のもの．下段は提案手法独自のもの．

Table 1 Parameters and their meanings. Upper part: parameters shared by the proposed method and the method by Noguchi et al.; lower part: parameters only for the proposed method.

| パラメータ | 意味  |
|-------|---|
| $b$   | ビット反転を適用して変動に対処する次元数の上限．提案手法では，多段階の段数 $N$ と $N = b + 1$ という関係にある． |
| $c$   | ハッシュ表におけるリスト長の上限．   |
| $d$   | ハッシュインデックスを求めるためのビットベクトルの次元数．実験では，ハッシュ表のサイズ $2^d$ を決めるパラメータでもある．  |
| $e$   | ビット反転を施す値の範囲．   |
| $r$   | 認識処理打ち切り判定のための，第 1 位に対する第 2 位の得票数比のしきい値．                          |
| $t$   | 認識処理打ち切り判定のための，第 1 位得票数のしきい値．                                     |

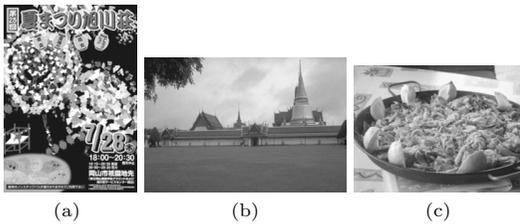


図 2 登録画像の例

Fig. 2 Examples of images in a database.

“ポスター”、“雑誌”、“表紙”などを用いた．図 2(a)に例を示す．B は PCA-SIFT のサイトで公開されていた画像であり，画像数は 18,500 枚である．このデータは主に自然写真や人物の写真などで構成されている．図 2(b)に例を示す．C は，写真共有サイトの flickr において “animal”，“birthday”，“food”，“japan” などのタグにより収集した 78,400 枚の画像からなる．主に図 2(c)に示すような物体や自然の写真，人物の写真などを含む．なお，収集の際には，600×600 pixel 以下のサイズの画像は除外し，画像の長辺が 640 pixel 以下になるように縮小した．また，特徴ベクトルが 100 個以下の画像も除外した．画像の 1 辺の長さの平均は A, B, C それぞれ 498, 612, 554 pixel であった．

次に，A, B, C の画像を用いて，表 2 に示した画像数からなる 4 種類のデータベース DB1~DB4 を作成し実験に用いた．ここで，大きいデータベースは，小さいデータベースをその一部として含む．なお，DB2 (1 万枚) からは，1 画像当たり平均 2,069 個の特徴ベクトルが抽出された．

表 2 データベースに含まれる画像数

Table 2 The number of images in the databases.

| データベース | DB1   | DB2    | DB3    | DB4     |        |
|--------|-------|--------|--------|---------|--------|
| 登録画像数  | 5,000 | 10,000 | 50,000 | 100,000 |        |
| 内訳     | A     | 1,667  | 3,100  | 3,100   | 3,100  |
|        | B     | 1,667  | 3,450  | 18,500  | 18,500 |
|        | C     | 1,666  | 3,450  | 28,400  | 78,400 |

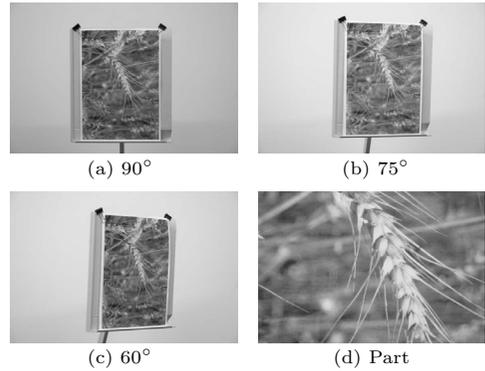


図 3 検索質問の例

Fig. 3 Examples of query images.

### 6.1.2 検索質問画像

検索質問としては，データベースに対応する画像のあるものとなないものの 2 種類を作成した．前者については，データベースに含まれる画像の中から 500 枚 (データセット A, B, C から各々 100, 200, 200 枚) を無作為に選択した．後者については，画像データベースには含まれない画像を 199 枚用意した (注5)．次に，これらを A4 の用紙に印刷し，カメラを用いて撮影した．得られた画像の例を図 3 に示す．図に示すとおり，紙面全体が写る配置で，紙面に対するカメラの光軸の角度  $\theta$  を  $90^\circ, 75^\circ, 60^\circ$  に変化させた．また，角度を  $90^\circ$  として紙面の一部を撮影した．その結果，1 枚の紙面に対して，合計 4 通りの画像を得た．更に，撮影した画像を  $512 \times 341$  pixel に縮小し，PCA-SIFT により特徴ベクトルを求めた．その結果，画像 1 枚当たり平均 612 個の特徴ベクトルが得られた．

### 6.2 リジェクトをしない場合

まず，対応する画像がデータベースにある検索質問のみを用いて，実験を行った．

#### 6.2.1 近似最近傍探索の精度と画像の認識率の関係

予備実験として，DB2 (1 万枚) を用いて，画像を

(注5)：当初は 200 枚用意したが，うち 1 枚が思いもかけずデータベースに含まれていることが判明したため 199 枚となった．

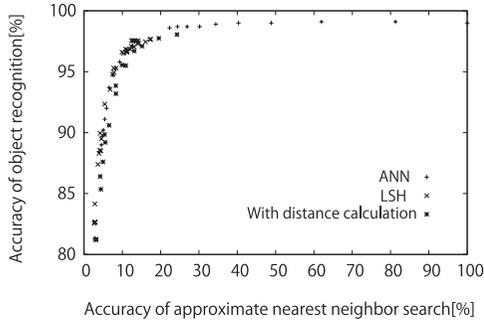


図 4 近似最近傍探索の精度と画像の認識率

Fig. 4 The relationship between the recognition rate and accuracy of approximate nearest neighbor search.

認識するために必要な最近傍探索の精度を調べた．具体的には，多段階化なしの手法（ANN，LSH，距離計算ありの手法）について，パラメータを様々に変え，近似最近傍探索の精度と画像の認識率の関係を計測した．近似最近傍探索の精度とは，近似最近傍探索によって真の最近傍が求まった割合である．

結果を図 4 に示す．この結果から，近似最近傍探索の精度が 100% から 20% あたりまでは，近似最近傍探索の精度が減少しても，認識率はほとんど減少しないことが分かる．これは，間違っただ他の画像に投票されても，正解の画像と他の画像の得票数が逆転するまでには至らないためと考えられる．また，近似最近傍探索の精度と認識率には，手法に依存しない関係があることもうかがえる．

### 6.2.2 認識処理打ち切りのパラメータと認識率，処理時間の関係

DB2 (1 万枚) を用いて，認識処理打ち切りに関するパラメータについて調べた．まず，距離計算ありの手法を用いて，4.2 で述べた認識処理打ち切りのパラメータ  $r, t$  と認識率と処理時間の関係について調べた．具体的には， $b = 10, c = 5, d = 28, e = 200$  とし， $t = 0, 2, 4$  について  $r$  を 0 から 1 まで 0.1 刻みに変化させ，処理時間と認識率を調べた．ここで  $r = 0$  の場合は，最終段まで認識処理打ち切りの条件が満たされず，最終段で最大得票数のものを出力することになる．結果を図 5 に示す．処理時間は  $r$  を大きくするにつれて減少している．認識率について見ると，最低得票数のしきい値  $t = 0, 2$  の場合には， $r$  を大きくすると認識率が低下する．一方， $t = 4$  とした場合には， $r$  を変化させても認識率は低下しない．このことから，得票

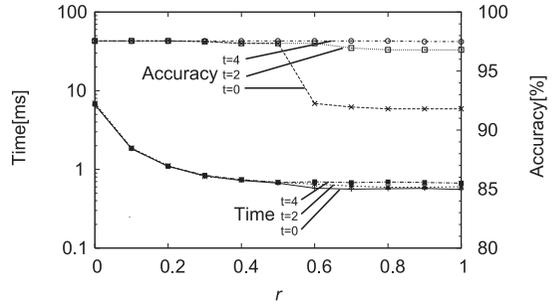


図 5 距離計算あり： $r, t$  と認識率，処理時間の関係

Fig. 5 With distance calculation: Relationship among  $r, t$ , recognition rate and processing time.

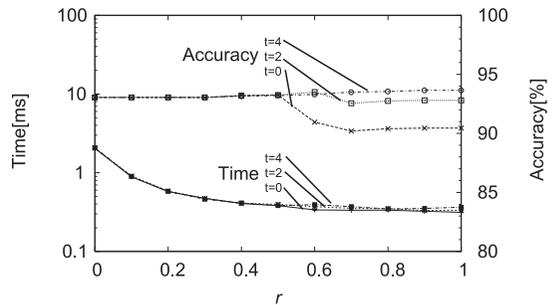


図 6 距離計算なし： $r, t$  と認識率，処理時間の関係

Fig. 6 Without distance calculation: Relationship among  $r, t$ , recognition rate and processing time.

数 1 位と 2 位の比は重要ではなく，1 位の得票数が一定数以上得られればよいことが分かる．そこで，これ以降の実験では  $r = 1$  (得票数 1 位と 2 位が同数の場合は次の段に処理を進める)， $t = 4$  とする．

同様に，距離計算なしの手法を用いて， $r, t$  と認識率と処理時間の関係について調べた． $b = 6, c = 3$  とし，他のパラメータについては，距離計算ありの手法と同一とした．結果を図 6 に示す．距離計算のありなしにかかわらず同じ傾向を示すことが分かる．

### 6.2.3 多段階化の段数と認識率，処理時間の関係

次に，DB2 (1 万枚) を用いて，多段階化の段数  $N$  を変更し，多段階化の効果を検証した．まず，距離計算ありの手法を用いて， $b = (N - 1)$  と認識率と処理時間の関係について調べた． $c = 5, d = 28, e = 200$  とし， $b$  を変化させた結果を図 7 に示す．多段階化ありの場合は，なしの場合と比較し，ほとんど認識率が低下せずに，処理時間を大幅に削減できることが分かった．また，多段階化の段数が多くなるにつれて，処理時間削減の効果が大きくなっていることが分かつ

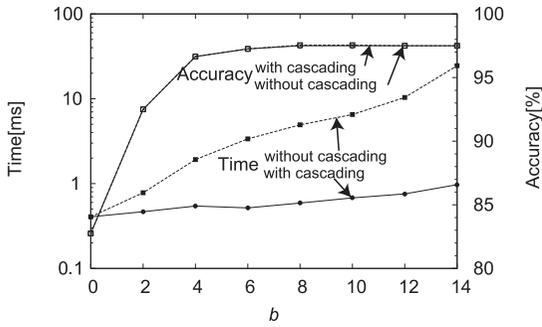


図 7 距離計算あり:  $b$  と認識率, 処理時間の関係  
 Fig. 7 With distance calculation: Relationship among  $b$ , recognition rate and processing time.

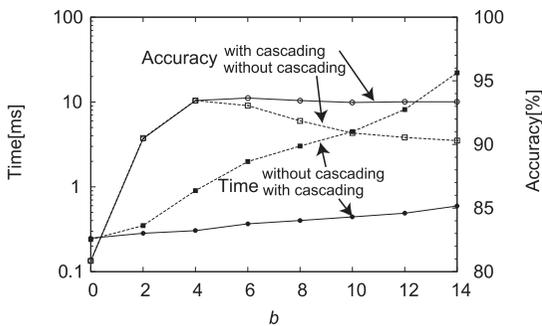


図 8 距離計算なし:  $b$  と認識率, 処理時間の関係  
 Fig. 8 Without distance calculation: Relationship among  $b$ , recognition rate and processing time.

た. 同様に, 距離計算なしの手法を用いて,  $b$  と認識率と処理時間の関係について調べた.  $c = 3, d = 28, e = 200$  とし,  $b$  を変化させた結果を図 8 に示す. 距離計算なしの手法でも, 処理時間を削減できることが分かった. また多段階化なしの手法では,  $b$  を大きくしすぎると認識率が低下する. この理由は,  $b$  の増加により誤った票が増え, 正解の票を上回るためと考えられる. 一方, 多段階化ありの手法ではあまり低下しないことが分かる. これは, ほとんどの検索質問は途中の段で検索処理を打ち切られており, 段数を増やしても  $b$  まで到達しないためと考えられる.

#### 6.2.4 各段の認識器で出力される割合

DB2 (1 万枚) を用いて, 各検索質問について認識打ち切りが行われている段数を調べた. パラメータは, 先ほどの実験の結果で多段階化による処理時間の削減効果が高かった,  $b = 10, c = 5, d = 28, e = 200$  とした. 距離計算ありの手法による結果を図 9 に示す. 図上部の枠内に, 多段階化の構成を示す. 各段で認識打ち切りが行われ, 回答が出力された検索質問の枚数を

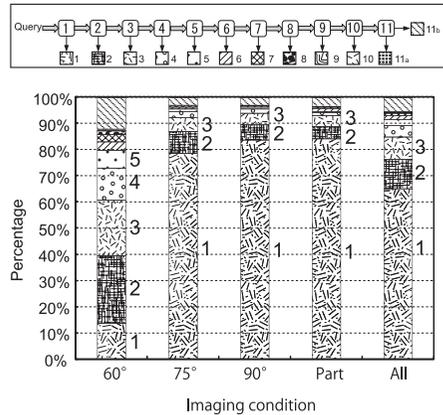


図 9 距離計算あり: 各段での出力の割合  
 Fig. 9 With distance calculation: The rate of output at each step.

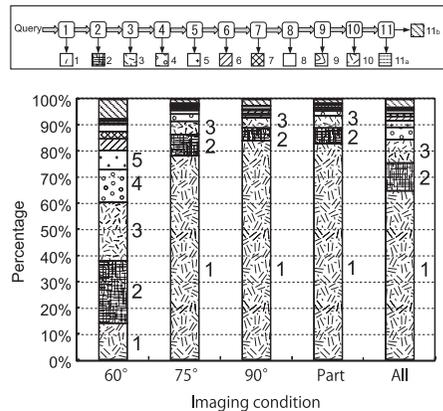
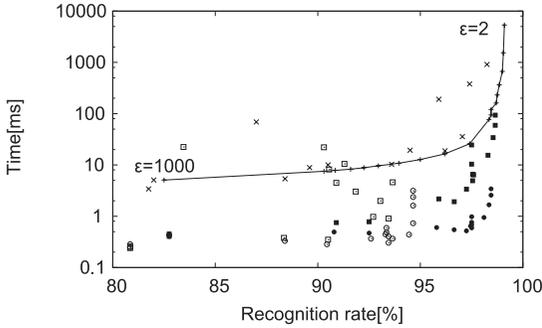


図 10 距離計算なし: 各段での出力の割合  
 Fig. 10 Without distance calculation: The rate of output at each step.

集計した. 最終段 (11 段目) では, 認識処理打ち切りの判定に合格しなかった検索質問についても, 強制的に回答が出力される. そのため, 認識処理打ち切りの判定に合格したものを  $11_a$ , 不合格のものを  $11_b$  として集計した.

結果から, 検索質問全体の 65% が 1 段目で回答されていることが分かる. 撮影条件を比較すると,  $90^\circ, 75^\circ, 60^\circ$  と撮影条件が厳しくなるほど, 1 段目で回答した割合が減少している. このことから, 近似の程度を画像に応じて変更し, 認識の簡単な画像に対しては, 時間をかけずに認識できていることが分かる.

同様に, 距離計算なしの手法を用いた結果を図 10 に示す. パラメータは, 先ほどの実験の結果より  $b = 10$ ,



| Conventional method          |   | Proposed method              |   |
|------------------------------|---|------------------------------|---|
| ANN                          | — | With distance calculation    | • |
| LSH                          | x | Without distance calculation | ○ |
| With distance calculation    | ■ |                              |   |
| Without distance calculation | □ |                              |   |

図 11 リジェクトをしない場合の各手法の比較  
Fig. 11 Comparison of methods without rejection.

$c = 3, d = 28, e = 200$  とした。各段で出力される割合は、距離計算ありの手法とあまり差がないことが分かる。このことから、距離計算なしの手法でも近似の程度を画像に応じて変更できていることが分かる。

### 6.2.5 各手法の比較

各手法の特徴を比較するため、パラメータを様々に変え、認識率と処理時間の関係を描いたグラフを図 11 に示す。ANN で許容誤差  $\epsilon$  を 2 から 1000 まで変化させたものを線で描き、評価の基準とした。右にプロットされているものほど認識率が高く、下にプロットされているものほど処理時間が短い。そのため、右下にプロットされているものほど優れているといえる。LSH は、ほぼ ANN の線を越えることはなかった。野口らの手法では、最大認識率は ANN に及ばないものの、認識率が 98% 以下の場合には、同じ認識率を、ANN に比べて 1/10 程度の処理時間で実現できている。提案手法では、多段階化を行うことで、処理時間が ANN の 1/40 程度にまで削減されている。

### 6.2.6 スケーラビリティ

次に、登録画像数と認識率と処理時間の関係について調べた。

まず、距離計算ありの手法を用いて、登録画像数を 5000 枚 (DB1) から 10 万枚 (DB4) まで変化させ、実験を行った。

パラメータは、DB4 (10 万枚) を用いた場合に、良い認識率が得られた  $c = 10, d = 28, e = 400$  を用いた。登録画像数を変化させた結果を図 12 に示す。多

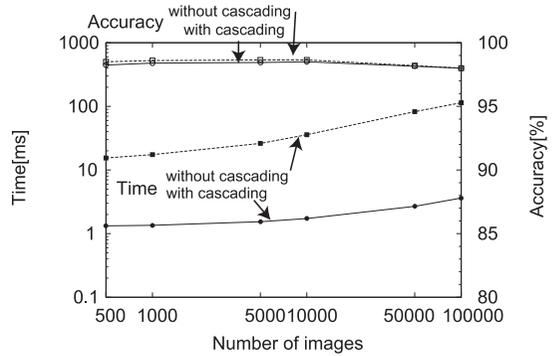


図 12 距離計算あり：登録画像数と認識率，処理時間の関係

Fig. 12 With distance calculation: Relationship among the number of images in the DBs, the recognition rate and processing time.

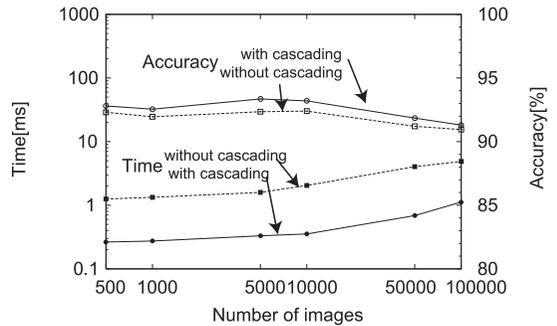


図 13 距離計算なし：登録画像数と認識率，処理時間の関係

Fig. 13 Without distance calculation: Relationship among the number of images in the DBs, the recognition rate and processing time.

段階化を行わない場合と比較すると、登録画像数が増えるにつれ、処理時間を削減できていることが分かる。

同様に、距離計算なしの手法を用いて、登録画像数と認識率と処理時間の関係について調べた。パラメータは、10 万枚の画像のデータベースを用いた場合に、良い認識率が得られた  $c = 3, d = 28, e = 200$  を用いた。登録画像数を変化させた結果を図 13 に示す。距離計算なしの手法では、多段階化することによって認識率が向上することが分かる。また、登録画像数によらず、ほぼ同じ割合で処理時間が削減できている。

なお、DB4 (10 万枚) を用いた場合のメモリ使用量は、距離計算ありの手法で約 16 GByte 距離計算なしの手法で約 5 GByte であった。

### 6.3 リジェクトをする場合

最後に、DB2 (1 万枚) を用いたリジェクトをする

表 3 リジェクトをする場合の結果  
Table 3 Result with rejection.

| パラメータの決定基準<br>(学習サンプルに対して)                              | 手法     | 対応する画像のある検索質問 |           |           |           | 対応する画像のない検索質問 |           |           |  |
|---|--------|---------------|-----------|-----------|-----------|---------------|-----------|-----------|--|
|   |        | $C_1$ [%]     | $E_1$ [%] | $R_1$ [%] | 処理時間 [ms] | $E_2$ [%]     | $R_2$ [%] | 処理時間 [ms] |  |
| 基準 A<br>$E_1 = 0, E_2 = 0$<br>処理時間 20 ms 以下, $\min R_1$ | 距離計算あり | 86.70         | 0.00      | 13.30     | 1.37      | 0.4           | 99.63     | 10.3      |  |
|   | 距離計算なし | 76.30         | 0.00      | 23.70     | 0.6       | 0.12          | 99.88     | 3.1       |  |
| 基準 B<br>$\min(E_1 + E_2 + R_1)$<br>処理時間 20 ms 以下        | 距離計算あり | 95.80         | 0.05      | 4.15      | 1.3       | 1.75          | 98.25     | 15.1      |  |
|   | 距離計算なし | 86.45         | 0.05      | 13.50     | 0.6       | 0.63          | 99.37     | 5.0       |  |

場合の実験結果について述べる．結果の評価尺度を次のように定める．まず，対応する画像のある検索質問に対しては，認識率  $C_1$ ，誤認識率  $E_1$ ，リジェクト率  $R_1$  ( $C_1 + E_1 + R_1 = 1$ ) とする．対応する画像のない検索質問に対しては，誤認識率  $E_2$ ，リジェクト率  $R_2$  ( $E_2 + R_2 = 1$ ) とする．実験には，10-fold cross validation を用いた．学習サンプルに対して，以下の 2 通りの基準を用いてパラメータを求め，テストサンプルに適用した．

基準 A  $E_1 = 0, E_2 = 0$ ，処理時間 20 ms 以下という条件の下で  $R_1$  が最小

基準 B 処理時間 20 ms 以下という条件の下で  $E_1 + E_2 + R_1$  が最小

パラメータについては，距離計算ありの手法，距離計算なしの手法の両方で， $b = 5, 10, 15, c = 2, 5, 10, 100, d = 20, 24, 28, e = 200, 400, 600, r = 0.2, 0.4, 0.6, t = 4, 8, 12$  のすべての組合せについて試した．

結果を表 3 に示す．基準 A でパラメータを設定した場合，距離計算ありの提案手法では，リジェクト率  $R_1$  が 13.3% の場合に誤認識率  $E_1$  を 0% とすることができた．このとき，誤認識率  $E_2$  は 0.4% となった<sup>(注6)</sup>．基準 B の場合には，誤認識率  $E_2$  が 1.75% に増加したものの，認識率  $C_1$  を 95.8% に増加させることができた．一方，距離計算なしの提案手法では，距離計算ありの手法に及ばない結果となった．

処理時間については，対応する画像のない検索質問の方が，3 から 7 倍程度長くなった．これは，多段階化によって，対応する画像のない検索質問のほとんどは，最終段まで処理されてリジェクトされるのに対し，対応する画像のある検索質問は，最終段まで到達せずに回答が出力されているためである．

(注6):  $E_1 = E_2 = 0$  という条件は 10-fold CV でパラメータを設定するときのものであり，その結果得られるパラメータは，テストサンプルに対して  $E_2 = 0$  を保証するものではない．

以上のように，提案手法では，対応する画像のある検索質問の 4~13% 程度をリジェクトすることで，処理時間を犠牲にせず，誤認識率を極めて低く抑え得ることが分かった．

## 7. む す び

本論文では，近似最近傍探索による識別器を，近似の程度を変化させて多段階に接続することにより，高速に物体を認識する手法を提案した．1 万画像を対象とした実験の結果，ANN や LSH といった従来の近似最近傍探索手法を用いる場合と比較して，認識率 98% を 1/40 程度の処理時間で得ることができた．また，対応する画像がデータベースにある検索質問の 1 割程度をリジェクトすることで，誤認識を 0% に近づけることができた．更に，10 万画像を対象とした実験により提案手法の高いスケーラビリティを確認した．

今後の課題には，平面物体以外への適用などがある．謝辞 本研究の一部は日本学術振興会科学研究費補助金（基盤研究 (B)19300062）の補助による．

## 文 献

- [1] D. Lowe, "Distinctive image features from scale-invariant keypoints," Int. J. Comput. Vis., vol.60, no.2, pp.91–110, 2004.
- [2] S. Arya, D. Mount, R. Silverman, and A.Y. Wu, "An optimal algorithm for approximate nearest neighbor searching," J. ACM, vol.45, no.6, pp.891–923, 1998.
- [3] M. Datar, N. Immorlica, P. Indyk, and V. Mirrokni, "Locality-sensitive hashing scheme based on p-stable distributions," Proc. 20th Annual Symposium on Computational Geometry, pp.253–262, 2004.
- [4] 野口和人, 中居友弘, 黄瀬浩一, 岩村雅一, "特徴ベクトルの近傍探索と物体認識の効率に関する実験的検討," 信学技報, PRMU2006-68, 2006.
- [5] 荒井英剛, 武本浩二, 加藤丈和, 和田俊和, "階層的固有空間による高次元最近傍探索の高速化," 画像の認識・理解シンポジウム (MIRU2006), pp.291–297, 2006.
- [6] D. Nistér and H. Stewénius, "Scalable recognition with a vocabulary tree," Proc. CVPR2006, pp.775–781, 2006.

- [7] Y. Ke and R. Sukthankar, "PCA-SIFT: A more distinctive representation for local image descriptors," Proc. CVPR2004, vol.2, pp.506-513, 2004.

(平成 21 年 3 月 6 日受付, 7 月 13 日再受付)



野口 和人

平 19 阪府大・工・情報卒．平 21 同大学院博士前期課程了．現在，ソニー（株）勤務．在学中，画像認識に関する研究に従事．平 21 本会 PRMU 研究奨励賞受賞．



黄瀬 浩一（正員）

昭 61 阪大・工・通信卒．昭 63 同大学院博士前期課程了．同年同大学院博士後期課程入学．平 2 阪府大・工・電気助手．現在，同大学院工学研究科教授．博士（工学）．平 12～13 ドイツ人工知能研究センター客員教授．文書画像解析，情報検索，画像認識などの研究に従事．平 18 本学会論文賞，平 19 IAPR/ICDAR Best Paper Award 各受賞．現在，IAPR TC11 (Reading Systems) Vice Chair, IAPR ICPR Standing Committee, Conferences & Meetings Committee 委員．情報処理学会，人工知能学会，電気学会，IEEE，ACM 等各会員．



岩村 雅一（正員）

平 10 東北大・工・通信卒．平 15 同大学院博士課程了．同年同大学院工学研究科助手．平 16 阪府大大学院工学研究科助手．現在助教．博士（工学）．パターン認識，画像認識，情報検索などの研究に従事．平 18 本学会論文賞，平 19 IAPR/ICDAR The Best Paper Award 各受賞．IEEE，情報処理学会，Pattern Recognition Society 各会員．