

(12) 特許協力条約に基づいて公開された国際出願

(19) 世界知的所有権機関  
国際事務局



(43) 国際公開日  
2010年12月16日(16.12.2010)

PCT

(10) 国際公開番号  
WO 2010/143573 A1

- (51) 国際特許分類:  
G06T 7/00 (2006.01) G06T 1/00 (2006.01)  
G06F 17/30 (2006.01)
- (21) 国際出願番号: PCT/JP2010/059352
- (22) 国際出願日: 2010年6月2日(02.06.2010)
- (25) 国際出願の言語: 日本語
- (26) 国際公開の言語: 日本語
- (30) 優先権データ:  
特願 2009-139148 2009年6月10日(10.06.2009) JP
- (71) 出願人 (米国を除く全ての指定国について): 公立大学法人大阪府立大学(OSAKA PREFECTURE UNIVERSITY PUBLIC CORPORATION) [JP/JP]; 〒5998231 大阪府堺市中区学園町1-1 Osaka (JP).
- (72) 発明者; および
- (75) 発明者/出願人 (米国についてのみ): 井上 勝文 (INOUE, Katsufumi) [JP/JP]; 〒5998231 大阪府堺市中区学園町1-1 公立大学法人大阪府立大学内 Osaka (JP). 黄瀬 浩一 (KISE, Kouichi) [JP/JP]; 〒5998231 大阪府堺市中区学園町1-1 公立大学法人大阪府立大学内 Osaka (JP).

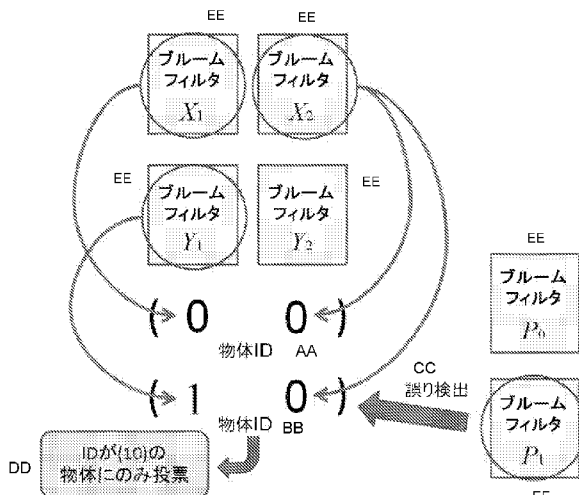
- (74) 代理人: 野河 信太郎 (NOGAWA, Shintaro); 〒5300047 大阪府大阪市北区西天満5丁目16-3 西天満ファイブビル 野河特許事務所 Osaka (JP).
- (81) 指定国 (表示のない限り、全ての種類の国内保護が可能): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.
- (84) 指定国 (表示のない限り、全ての種類の広域保護が可能): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), ユーラシア (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), ヨーロッパ (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

[続葉有]

(54) Title: OBJECT IDENTIFICATION IMAGE DATABASE CREATING METHOD, CREATING APPARATUS AND CREATING PROCESS PROGRAM

(54) 発明の名称: 物体認識用画像データベースの作成方法、作成装置および作成処理プログラム

[図24]



EE X<sub>1</sub>, X<sub>2</sub>, Y<sub>1</sub>, Y<sub>2</sub>, P<sub>0</sub>, P<sub>1</sub> BLOOM FILTERS  
 AA, BB OBJECT IDS  
 CC ERROR DETECTION  
 DD VOTE ONLY FOR OBJECTS THE IDS OF WHICH ARE (10)

(57) Abstract: Provided are methods for creating and searching, by use of a stochastic association data structure, an image database having an excellent spatial efficiency. There are included a feature extracting step and a registering step. In the feature extracting step, one or more feature vectors, which are representative of the respective ones of the local features of portions of an image that is to be registered with an image database and is representative of an object, are extracted, as local feature amounts, from the image. The registering step utilizes an association data structure wherein values, which have been allocated to the respective ones of a plurality of keys serving as elements to be registered, are caused to be associated with based on those keys and wherein a compressed data structure, in which false positive is tolerated to thereby compress the data amount, is included as a constituent element. That is, the registering step uses the association data structure to register the feature vectors as the keys in the association data structure.

(57) 要約:

[続葉有]

WO 2010/143573 A1



添付公開書類:

— 国際調査報告 (条約第 21 条(3))

---

確率的な連想データ構造を用いた、空間的効率の良い画像データベースの作成手法および検索手法を提供する。画像データベースに登録されるべき画像であって物体を表す画像から、その画像の各所の局所的な特徴をそれぞれ表現する1以上の特徴ベクトルを局所特徴量としてそれぞれ抽出する特徴抽出工程と、登録されるべき要素としての複数のキーにそれぞれ割り当てられた値をそのキーに基づいて連想させる連想データ構造であって、偽陽性を許容することによりデータの容量を圧縮する圧縮データ構造を構成要素として含む連想データ構造を用い、各特徴ベクトルをキーとして前記連想データ構造に登録する登録工程とを備える。

## 明 細 書

### 発明の名称：

# 物体認識用画像データベースの作成方法、作成装置および作成処理プログラム

### 技術分野

[0001] この発明は、物体認識用画像データベースの作成方法、作成装置および作成処理プログラムに関する。より詳細には、データを要素とする集合があるとき各データ（キー）に割り当てられた値をそのキーに基づいて連想させるデータ構造であって、偽陽性(False Positive)を許容することによりデータ容量を圧縮する連想データ構造を用いてなる、特徴ベクトルの圧縮表現とその特定物体認識への応用に関する。

### 背景技術

[0002] 特定物体認識とは、物体のインスタンスを認識するタスクである。この明細書では、SIFT（例えば、非特許文献1参照）などの局所特徴量を用いた特定物体認識について述べる。

[0003] 我々の周囲には物体のインスタンスが無数にあるため、特定物体認識を真に実用的なものとするには、識別できる物体数の大規模化が必須である。局所特徴量は数十から数百の次元を持つ実数値ベクトルであり、高い識別性を持つため、多数の物体を識別する目的に適している。認識処理の基本は、未知の物体から得た局所特徴量と既知の物体から得た局所特徴量を照合する処理である。特徴ベクトルの照合は最近傍探索によって容易に実現できるものの、その大規模化には、まだいくつかの障害を乗り越える必要がある。最も重要な障害は、処理速度とメモリ量に関するものである。

[0004] 幸い、処理速度については、これまでに有効な手法が提案されている。その中には、k-d木などの木構造に基づくもの（例えば、前記非特許文献1参照）、ハッシュに基づくもの（例えば、特許文献1、非特許文献2参照）などがある。これらの手法では、近似最近傍探索を用いることによって、処理速

度を大幅に改善している。

[0005] 一方、メモリ量の問題については十分な解決策は得られていない。局所特徴量は画像あたり数百から数千個にもなるため、物体数が大規模になるとその記録には膨大なメモリが必要となる。

これまでに、特徴ベクトルをVisual Word（例えば、非特許文献3参照）と呼ばれる代表ベクトルに置換することで必要なメモリ量を削減する手法が提案されている。この手法は、画像に写っている物体のクラス（例えば、一般的な「車」というカテゴリ）を認識する「一般物体認識」には有効な手法といえる。しかし、画像中の物体のインスタンス（例えば、前者の「車」というカテゴリの中の特定のモデル名）まで認識する「特定物体認識」への適合性は高いとはいえない。大規模な特定物体認識を行うためには、Visual Wordの数を多くしなければならない。高い認識率を得るためには、一つのVisual Wordに2~3個の特徴ベクトルを対応付けるのが限界でありそれより多くの特徴ベクトルを一つのVisual Wordで代表させるようにすると認識率が低下してしまう（例えば、非特許文献4参照）。特定物体認識におけるメモリ量削減の決め手にはなり得ない。

[0006] この他のメモリ量削減手法として、スカラ量子化（例えば、非特許文献5参照）や特徴ベクトルの取捨選択（例えば、非特許文献6、前記非特許文献4参照）などがある。しかし、上記の近似最近傍探索を用いた照合では、特徴ベクトル間の距離を計算する必要があるため、いずれにせよ個々の特徴ベクトルを記録しなければならない。従って、記録する特徴ベクトル数に応じたメモリ量が必要となり、その削減には限界が生じる。

[0007] また、この問題を解決する一つの考え方は、照合に際して距離計算を放棄することである。これにより特徴ベクトルの記録も不要となるため、メモリ量の大幅な削減が可能となる。実際、特徴ベクトルをハッシュ表に登録しておき、それを検索することによって認識する手法が提案されている（例えば、特許文献1、前記非特許文献2参照）。この手法では、特徴ベクトルの照合を、距離あるいは類似度という量的な概念に基づく類似検索ではなく、ハ

ハッシュ関数の値（ハッシュ値）が同じかどうかという一致検索によって実現している。ハッシュ表には特徴ベクトルの有無が記録されるだけなので、メモリ量の大幅な削減が可能となる。

## 先行技術文献

## 特許文献

[0008] 特許文献1：国際公開第2008/026414号パンフレット

## 非特許文献

[0009] 非特許文献1：D. Lowe: "Distinctive Image Features from Scale-Invariant Keypoints", International Journal of Computer Vision, Vol. 60, No. 2, pp. 91-110 (2004).

非特許文献2：野口和人，黄瀬浩一，岩村雅一：“近似最近傍探索の多段階化による物体の高速認識”，画像の認識・理解シンポジウム(MIRU2007)論文集、0S-B2-02, pp. 111-118 (2007).

非特許文献3：K. Kise, K. Noguchi and M. Iwamura: "Memory Efficient Recognition of Specific Objects with Local Features", Proc. of the 19th International Conference of Pattern Recognition (ICPR2008) WeAT3.1 (2008).

非特許文献4：井上勝文，三宅弘志，黄瀬浩一：“局所記述子に基づく3次元物体認識のためのメモリ削減—局所記述子の取捨選択によるアプローチ—”，画像の認識・理解シンポジウム(MIRU2008)論文集、0S15-3, pp. 363-370 (2008).

非特許文献5：本道貴行，黄瀬浩一：“特定物体認識のためのデータベース容量削減法の検討～局所特徴量の量子化と取捨選択～”，電子情報通信学会技術研究報告、Vol. 108, No. 484, PRMU2008-265, pp. 171-176 (2009).

非特許文献6：B. Chazelle, J. Kilian, R. Rubinfeld and A. Tal: "The Bloomier Filter: An Efficient Data Structure for Static Support Lookup Table", Proc. 15th Annual ACM-SIAM SODA, pp. 30-39 (2004).

## 発明の概要

### 発明が解決しようとする課題

[0010] ところが、この手法にもまだメモリ量削減の余地がある。正しい認識のためには、類似した特徴ベクトルが同じハッシュ値を持つようにしなければならない。ところが、これによって、ハッシュ表に記録される特徴ベクトルに偏りが生じる。その結果、ハッシュ表の大部分には何も記録されず、空間効率、即ち、メモリ空間のうちで有効利用されるメモリ領域の割合が悪化する。

[0011] この発明は、以上のような事情を考慮してなされたものであって、前述の課題を解決するため、データ集合の要素としての「キー」にそれぞれ割り当てられた値をそのキーに基づいて提供する連想データ構造を用いた、空間的効率の良い画像データベースの作成手法および検索手法を提供する。前記連想データ構造は、データ容量の圧縮、即ち、データの格納に必要なメモリ量を削減し高い空間効率を得る代償として偽陽性を許容するデータ構造を組み合わせてなる。前記データ構造は、多数のキーが登録されている状態で、あるキーが登録されているか否かを調べる（検索する）ために用いられる。前述の偽陽性は、検索を行うとそのキーに割り当てられていない値が返される場合があったり、そのキーが登録されていないにもかかわらず値が返されたりすることが起こることがあり得ることである。その意味で、前記データ構造は、確率的なデータ構造といえる。つまり、正解が得られる保証がなく誤った結果が返されることがあっても、誤った結果を返す確率が実用上支障のない程度に低ければよいとするのである。データベースの作成に要するメモリ量を抑制するために、偽陽性を許容することも一つの選択肢である。

[0012] 上記のハッシュ表を用いる代わりに前記データ構造を用いて、特徴ベクトルが記録されているかどうかを検索できる。大きな違いは、記録されていない特徴ベクトルを誤って検出するという偽陽性が生じるかどうかにある。ハッシュ表を用いる手法では偽陽性は生じ得ない。一方、前記データ構造を用いる手法では、一定の偽陽性を容認することの引き替えに、高い空間効率を

得ることが可能となる。

### 課題を解決するための手段

[0013] この発明は、物体認識用画像データベースに登録されるべき物体が撮影された画像から、その画像の局所的特徴を表す複数の特徴ベクトルを抽出する特徴抽出工程と、データを登録できかつあるデータが登録済か否かを判定し得るデータ構造に対し、ある物体から抽出された各特徴ベクトルを、その物体に予め割り当てられた識別子が前記特徴ベクトルから連想されるように登録する登録工程とを備え、各工程をコンピュータが実行し、前記登録に係る物体は、 $n$ ビット（ $n$ は自然数）の識別子が割り当てられ、前記登録工程は、前記識別子の各ビットのゼロ値登録用と1値登録用として用意された $2 \times n$ 個の前記データ構造に対し、各特徴ベクトルが抽出された物体の識別子に応じて各ビットのゼロ値登録用と1値登録用の何れか一方のデータ構造に前記特徴ベクトルをそれぞれ登録し、前記画像データベースは、検索質問としてある物体の画像が与えられたとき同じ物体が登録されているか否かの認識処理を行うべく、前記コンピュータまたは他のコンピュータによってアクセスされ、前記認識処理は、前記検索質問から複数の特徴ベクトルを検索質問ベクトルとして抽出し、各検索質問ベクトルと同値の特徴ベクトルが各データ構造に登録されているか否かを調べ、すべてのビットについてゼロ値または1値のいずれかに同値の特徴ベクトルが登録されているときは登録された各ビットの値に応じた $n$ ビット値を識別子の候補として得、各検索質問ベクトルにつき得られた候補を統計処理して検索質問に適合する識別子を見出すことを特徴とする物体認識用画像データベースの作成方法を提供する。

[0014] また、異なる観点から、この発明は、物体認識用画像データベースに登録されるべき物体が撮影された画像から、その画像の局所的特徴を表す複数の特徴ベクトルを抽出する特徴抽出部と、データを登録できかつあるデータが登録済か否かを判定し得るデータ構造体の集合からなるデータ構造部と、ある物体から抽出された各特徴ベクトルからその物体に予め割り当てられた識別子が前記特徴ベクトルから連想されるように前記特徴ベクトルを前記デー

タ構造部に登録する登録部とを備え、前記登録に係る物体は、 $n$ ビット（ $n$ は自然数）の識別子が割り当てられ、前記データ構造部は、前記識別子の各ビットのゼロ値登録用および1値登録用として $2 \times n$ 個のデータ構造体を少なくとも有し、前記登録部は、各特徴ベクトルが抽出された物体の識別子に応じて各ビットのゼロ値登録用と1値登録用の何れか一方のデータ構造に前記特徴ベクトルをそれぞれ登録し、前記画像データベースは、検索質問としてある物体の画像が与えられたとき同じ物体が登録されているか否かの認識処理を行う認識装置によってアクセスされ、前記認識装置は、前記検索質問から複数の特徴ベクトルを検索質問ベクトルとして抽出し、各検索質問ベクトルと同値の特徴ベクトルが各データ構造に登録されているか否かを調べ、すべてのビットについてゼロ値または1値のいずれかに同値の特徴ベクトルが登録されているときは登録された各ビットの値に応じた $n$ ビット値を識別子の候補として得、各検索質問ベクトルにつき得られた候補を統計処理して検索質問に適合する識別子を見出すことを特徴とする物体認識用画像データベースの作成装置を提供する。

- [0015] さらに、この発明は、物体認識用画像データベースに登録されるべき物体が撮影された画像から、その画像の局所的特徴を表す複数の特徴ベクトルを抽出する特徴抽出処理と、データを登録できかつあるデータが登録済か否かを判定し得るデータ構造に対し、ある物体から抽出された各特徴ベクトルを、その物体に予め割り当てられた識別子が前記特徴ベクトルから連想されるように登録する登録処理とをコンピュータに実行させ、前記登録に係る物体は、 $n$ ビット（ $n$ は自然数）の識別子が割り当てられ、前記登録処理は、前記識別子の各ビットのゼロ値登録用と1値登録用として用意された $2 \times n$ 個の前記データ構造に対し、各特徴ベクトルが抽出された物体の識別子に応じて各ビットのゼロ値登録用と1値登録用の何れか一方のデータ構造に前記特徴ベクトルをそれぞれ登録し、前記画像データベースは、検索質問としてある物体の画像が与えられたとき同じ物体が登録されているか否かの認識処理を行うべく、前記コンピュータまたは他のコンピュータによってアクセスさ

れ、前記認識処理は、前記検索質問から複数の特徴ベクトルを検索質問ベクトルとして抽出し、各検索質問ベクトルと同値の特徴ベクトルが各データ構造に登録されているか否かを調べ、すべてのビットについてゼロ値または1値のいずれかに同値の特徴ベクトルが登録されているときは登録された各ビットの値に応じたnビット値を識別子の候補として得、各検索質問ベクトルにつき得られた候補を統計処理して検索質問に適合する識別子を見出すことを特徴とする物体認識用画像データベースの作成処理プログラムを提供する。

### 発明の効果

[0016] この発明の物体認識用画像データベースの作成方法において、前記登録工程は、前記識別子の各ビットのゼロ値登録用と1値登録用として用意された $2 \times n$ 個の前記データ構造に対し、各特徴ベクトルが抽出された物体の識別子に応じて各ビットのゼロ値登録用と1値登録用の何れか一方のデータ構造に前記特徴ベクトルをそれぞれ登録ので、特徴ベクトルが各ビットのゼロ値用または1値用のデータ構造に登録されているか否か、いずれのデータ構造に登録されているかに基づいてその特徴ベクトルから識別子の値を連想することができる。前記データ構造は、偽陽性を許容することによりデータ容量を圧縮するものであるから、従来と同じメモリ量でより大規模な物体認識用画像データベース実現することができる。後述する図9、図10、図15、図16などに示すように、本願発明は、ハッシュ表を用いて近似最近傍探索を行う手法と比較した場合に、より一層メモリ量の削減が可能であることが実験的に検証されている。

[0017] この発明において、前記データ構造として適用可能なものの具体的な一例は、ブルーム・フィルタ (Bloom Filter) であり、前記連想データ構造として適用可能なものの具体的な一例は、ブルーミア・フィルタ (Bloomier Filter、例えば、前記非特許文献6参照) である。ブルーミア・フィルタは、登録されるべき要素としての複数のキーにそれぞれ割り当てられた値をそのキーに基づいて連想させるための手法として知られたデータ構造である。ここ

で、連想とは、あるキーが与えられたときに、所定のデータ構造によりそのキーに対応付けられた値を短時間で得ることのできる仕組みを意味し、連想するとは、その仕組みによりキーに対応付けられた値を得ることといえる。ブルーム・フィルタは、一般に複数の共通なハッシュ関数を有する複数のブルーム・フィルタ（ $m$ ビットのビット配列）で構成され、あるキーが与えられたとき各ハッシュ関数から得られるハッシュ値を用いて各ブルーム・フィルタを参照し、参照結果に基づいて前記キーに対応付けられた値を得ることができる。その詳細は後述する。

[0018] この発明の物体認識用画像データベースに登録されるものは、物体を表す画像であり、検索質問（クエリー）も画像として与えられる。検索質問の画像が表す物体と一致する物体を前記画像データベースから検索できれば、その物体が認識されたといえる。

このような物体認識では、短時間のうちに物体認識が完了すること、および、少ないメモリ量で画像データベースを実現することが重要である。両者は、しばしばトレードオフの関係にあるが、この発明は、どちらかといえば、それらのうちメモリ量の削減に課題の重点がある。即ち、多量の画像データが登録される画像データベースを構築するときに有効な手法である。

以上の作用効果は、この発明による物体認識用画像データベースの作成装置および作成処理プログラムについても同様である。

### 図面の簡単な説明

[0019] [図1]従来の物体認識において認識時間を短縮するため識別器を直列に多段階化した構成を示す説明図である。

[図2]この発明に係るブルーム・フィルタへの登録処理の流れを示す説明図である。

[図3]この発明の実験例で用いた3次元物体の例を示す説明図である。

[図4]この発明の実験例で用いた2次元物体の例を示す説明図である。

[図5]この発明の実験例で検索質問として用いた画像であって、前記2次元物体の画像をみる視点の角度を変えて撮影した画像の様子を示す説明図である

。

[図6]この発明の手法と比較例である野口らの手法の各種パラメータを変えたときの物体の認識率を示す第1のグラフである（実験1、55個の3次元物体の認識実験）。

[図7]この発明の手法と比較例である野口らの手法の各種パラメータを変えたときの物体の認識率を示す第2のグラフである（実験1、55個の3次元物体の認識実験）。

[図8]比較例である野口らの手法の各種パラメータを変えたときの物体の認識率を示す第3のグラフである（実験1、55個の3次元物体の認識実験）。

[図9]この発明の手法と比較例としての野口らの手法とで、パラメータ $c$ を変化させたとき、認識処理に必要なメモリ量を比較した結果を示す第1のグラフである（実験1、55個の3次元物体の認識実験）。

[図10]この発明の手法と比較例としての野口らの手法とで、パラメータ $c$ を変化させたとき、認識処理に必要なメモリ量を比較した結果を示す第2のグラフである（実験1、55個の3次元物体の認識実験）。

[図11]この発明の手法で、パラメータ $t$ を変化させたとき、認識率を示すグラフである（実験1、55個の3次元物体の認識実験）。

[図12]この発明の手法と比較例である野口らの手法の各種パラメータを変えたときの物体の認識率を示す第1のグラフである（実験2、5000個の2次元物体の認識実験）。

[図13]この発明の手法と比較例である野口らの手法の各種パラメータを変えたときの物体の認識率を示す第2のグラフである（実験2、5000個の2次元物体の認識実験）。

[図14]比較例である野口らの手法の各種パラメータを変えたときの物体の認識率を示す第3のグラフである（実験2、5000個の2次元物体の認識実験）。

[図15]この発明の手法と比較例としての野口らの手法とで、パラメータ $c$ を変化させたとき、認識処理に必要なメモリ量を比較した結果を示す第1のグラフである（実験2、5000個の2次元物体の認識実験）。

[図16]この発明の手法と比較例としての野口らの手法とで、パラメータ $c$ を変化させたとき、認識処理に必要なメモリ量を比較した結果を示す第2のグラフである（実験2、5000個の2次元物体の認識実験）。

[図17]同程度のメモリが使える状況で、この発明の手法、比較例である野口らの手法の $d=28$ 、 $d=24$ についてパラメータ $b, c$ を変化させたときの認識率と処理時間を示すグラフである（実験2、5000個の2次元物体の認識実験）。

[図18]この発明の手法において、 $t$ を変化させたときの物体の認識率を示すグラフである（実験2、5000個の2次元物体の認識実験）。

[図19]この発明の手法において、誤り検出ブルーミア・フィルタの有効性を示す第1のグラフである（55個の3次元物体の認識実験）。

[図20]この発明の手法において、誤り検出ブルーミア・フィルタの有効性を示す第2のグラフである（55個の3次元物体の認識実験）。

[図21]この発明の手法において、誤り検出ブルーミア・フィルタの有効性を示す第3のグラフである（5000個の2次元物体の認識実験）。

[図22]この発明の手法において、誤り検出ブルーミア・フィルタの有効性を示す第4のグラフである（5000個の2次元物体の認識実験）。

[図23]この発明に係るデータベース作成処理の具体例を説明する第1の説明図である。

[図24]この発明に係るデータベース作成処理の具体例を説明する第2の説明図である。

### 発明を実施するための形態

[0020] 以下、この発明の好ましい態様について説明する。

この発明の物体認識用画像データベースの作成方法において、各データ構造は、偽陽性を許容することによりデータを格納するメモリの容量を削減するものであってもよい。

前記登録工程は、所定の規則を適用して前記識別子に係る誤り検出用の値を得、得られた値を前記誤り検出用のデータ構造に登録し、前記認識処理は、得られた識別子の候補と前記誤り検出用データ構造に登録された値とを照

合し、両者が一致する場合は前記検索質問ベクトルについての判定結果を統計処理の対象とし、一致しない場合は統計処理の対象としなくてもよい。このようにすれば、前記登録工程は、前記識別子の値に所定の規則を適用して誤り検出用の値を得、得られた値を前記誤り検出用データ構造にさらに登録するので、検索質問ベクトルがあるビット桁のゼロ値にも1値にも登録されていると判断された場合にも、誤り検出用データ構造に登録された値との整合性を判定し、誤り検出用データ構造を用いない場合よりも高い精度で検索質問ベクトルに関連する画像を検索することができる。

また、前記誤り検出用の値が少なくとも1ビットからなり、前記誤り検出用データ構造は、ビットのゼロ値登録用と1値登録用としてデータ構造が用意されてなるものでもよい。このようにすれば、 $2 \times n$ 個の前記データ構造を用いて識別子を連想する連想データ構造が構成される場所、さらに前記データ構造を追加することにより検出用データ構造を構成することができる。

[0021] また、前記認識処理は、偽陽性のために、ゼロ値および1値登録用のいずれのデータ構造にも同一の検索質問ベクトルが登録されていると判断されるビットの数が予め定められた数を超える場合、その検索質問ベクトルについての判定結果を統計処理の対象から除外してもよい。このようにすれば、いわゆる偽陽性によりある検索質問ベクトルに関連する画像の識別子として所定の数を超える値が得られた場合、誤った値を多数含むその判断結果を特定の基礎から除外することにより、前記除外を行わない場合よりも高い精度で検索質問ベクトルに関連する画像を検索することができる。

[0022] さらにまた、前記認識処理は、前記検索質問ベクトルに関連があると判定される識別子の値に対して所定の得点を与え、前記検索質問ベクトルに関連があると判定される識別子が存しない場合はいずれの識別子にも得点を与えず、前記検索質問から抽出される検索質問ベクトルについての統計処理に基づき最多得点を得た識別子を決定してもよい。このようにすれば、複数の検索質問ベクトルの判断結果を多数決処理する、いわゆる投票処理により、個

々の判断結果が正しい確率が100%でなくても、多数決処理により妥当な結果を得ることができる。

[0023] 前記データ構造は、ブルーム・フィルタであってもよい。

また、前述の作成方法および検索方法において、各データ構造は、それを構成するビットの数 $m$  ( $m$ は自然数)が予め定められた登録ベクトルの最大数よりも大きくてもよい。より好ましくは、 $m$ は、各登録ベクトルが登録された後の各データ構造が、適度にゼロ値のビットを含む疎な状態になるように決定される。このようにすれば、データ構造の誤検出の確率、即ち、偽陽性の発生確率を低い値にすることができる。ただし、あまり $m$ を大きくしすぎると空間効率が低下する。設計者は、両者の兼ね合いから実験的にあるいは経験的に妥当な値を決定すればよい。

ここで示した種々の好ましい態様は、それら複数を組み合わせることもできる。

[0024] 以下、図面を用いてこの発明をさらに詳述する。なお、以下の説明は、すべての点で例示であって、この発明を限定するものと解されるべきではない。

この発明の詳細な説明に先立ち、その技術的意義を把握しやすくするために関連研究について、少し詳しく説明しておく。

#### 1. 関連研究について

[0025] 局所特徴量を用いた特定物体認識では、メモリ量の削減が重要な課題である。削減の従来手法には、特徴ベクトルのベクトル量子化を行うもの（前記非特許文献7参照）や、スカラ量子化を行うもの（前記非特許文献3参照）がある。

[0026] 前者の、ベクトル量子化をする手法では、複数の特徴ベクトルをVisual Wordと呼ばれる代表ベクトルに置き換えることでメモリ量を削減している。しかし大規模な認識を行うためには、一つのVisual Wordに2~3個の特徴ベクトルを対応付けるのが限度であり、それ以上の数を一つのVisual Wordに対応付けると認識率が低下することが報告されている（前記非特許文献5参照）。

- [0027] 一方、後者のスカラ量子化する手法では、特徴ベクトルの各次元をスカラ量子化することでメモリ量を削減している。この手法では、ある程度大規模な物体認識でも有効であるが、特徴ベクトルの数を1 bitで表現したとしても、特徴ベクトルの数に比例してメモリ量が増加するため、更なる大規模な物体認識を行うには限界がある。このため、特徴ベクトルそのものの容量を削減する手法には限界がある。
- [0028] 特徴ベクトル自体の容量を削減する以外のアプローチには、主記憶ではなく補助記憶を利用する手法がある（例えば、F. Fraundorfer, H. Stewenius and D. Nister: "A Binning Scheme for Fast Hard Drive Based Image Search", Proc. of CVPR2007, pp. 1-6 (2007)参照。また、姫井教考, 和田俊和: "B+木を用いた補助記憶装置上での近似最近傍探索", 電子情報通信学会技術研究報告, Vol. 108, No. 484, PRMU2008-273, pp. 223-228 (2009)参照)。その他、データベースに登録する特徴ベクトルをサンプリングする手法（例えば、前記非特許文献5、4参照）がある。
- [0029] 補助記憶を利用する手法では、記録している場所へのポインタのみを主記憶に保存することでメモリ量を削減している。この手法では、かなり大規模な物体認識を行えるが、認識処理に時間がかかるという問題点が生じる。
- [0030] 特徴ベクトルをサンプリングする手法では、認識に有効な特徴ベクトルを取捨選択することによりメモリ量を削減している。この手法でも、大規模な物体認識を行うためには、削減できる特徴ベクトルの数に限界があり、それ以上の数を削減すると認識率が低下してしまう。
- [0031] 上に述べたような手法では、特徴ベクトルの距離を計算するため、どの手法でも個々の特徴ベクトル自体を記録する必要があり、認識に必要なメモリ量を削減するには限界がある。
- [0032] この問題を解決するための一つのアプローチとして、特徴ベクトルの距離計算を放棄することが考えられる。このような考えに基づいた手法として、ハッシュ表を用いた手法（前記特許文献1、非特許文献2参照）が提案されている。しかし、この手法にも問題点があり、ハッシュ表に登録される特徴

ベクトルに偏りがあり、ハッシュ表の大部分が何も記録されず、空間効率が悪い。そこでこの発明では、特徴ベクトルの距離計算を行わないという考えに基づき、ハッシュ表と比べてはるかに空間効率の良いブルーミア・フィルタを用い、メモリ量を削減する手法を提案する。

## 2. 野口らの手法

[0033] 本節では、この発明の基礎となる野口らの手法（前記特許文献1、非特許文献2参照）について説明する。野口らの手法は、PCA-SIFT（例えば、Y. Ke and R. Sukthankar: "PCA-SIFT: A more distinctive representation for local image descriptors", Proc. of CVPR2004, Vol. 2, pp. 506-513 (2004)参照）で求めた36次元の特徴ベクトルを用いる特定物体認識手法である。野口らの手法は、ハッシュ表に特徴ベクトルを登録するデータベース作成処理と、このデータベースを用い、投票処理に基づいて物体を認識する認識処理の二つに分かれる。以下に、具体的な処理について説明する。

### 2-1. データベース作成

[0034] 本節では、野口らの手法のデータベース作成処理について説明する。野口らの手法ではまず、特徴ベクトル $p$ の第1次元から第 $d$ 次元( $d \leq 36$ )までをとり、 $p' = (p_1, p_2, \dots, p_d)$ を作成する。次にこのベクトル $p'$ を用い、

[0035] [数1]

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

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

[0036] [数2]

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

[0037] というハッシュ関数より、ハッシュ表に登録するためのインデックスを求める。ここで、 $H_{size}$ は、ハッシュ表のサイズである。求まったインデックスに、特徴ベクトル $p$ に対する物体IDを登録する。ここで、物体IDは、この発明の「識別子」に対応する。

[0038] 野口らの手法では、特徴ベクトルの登録時に衝突が生じた場合、リストとして特徴ベクトルを登録していく。但し、計算コストの削減のために、リスト長に閾値 $c$ を設け、リスト長が $c$ よりも長くなる場合、リスト全体をハッシュ表から削除し、以後の登録を禁止する。これは、同じハッシュ値を持つ特徴ベクトルは互いに類似しているため、物体認識にあまり寄与しないという考えに基づく。以上の処理を全ての特徴ベクトルに対して行い、データベースを作成する。

## 2-2. 認識処理

[0039] 野口らの手法では、質問画像からPCA-SIFTで求めた特徴ベクトルを用い、データベース作成処理と同様に、ハッシュ表のインデックスを求める。そして、求めたハッシュインデックスに登録されている物体ID全てに投票を行う。この処理を、質問画像から得られる特徴ベクトル全てに対して行い、最終的に最も得票数が多い物体を認識結果とする。以下に具体的な処理について説明する。

[0040] まず、質問画像から得られる特徴ベクトルから、ハッシュ表のインデックスを求める。このとき、野口らの手法で用いるハッシュ関数の性質より、次のような問題が発生する。検索質問画像とデータベースの画像が全く同じになるということはほとんどないため、質問画像から得られる特徴ベクトルは、各次元の値がデータベース作成に用いる特徴ベクトルと異なったものとなる。これが原因で異なったビットベクトルに変換されてしまうと、真に対応する物体IDをハッシュ表より検索することができなくなる。そこで、この問題に対処するために、各次元の値に、許容変動幅 $e$ を設ける。具体的に説明すると、検索質問となる特徴ベクトル  $q=(q_1, q_2, \dots, q_d)$  において、

[0041]

[数3]

$$|q_j| \leq e$$

を満たす次元jに対して、 $u_j$ だけではなく

[0042] [数4]

$$u'_j = (u_j + 1) \bmod 2 \quad (0 \text{ ならば } 1, 1 \text{ ならば } 0)$$

[0043] も用いて特徴ベクトルを検索する。例を示すと、ビットベクトル  $u=(1, 0, 0, 1)$  の第3次元目がこの処理の対象であった場合、 $u'=(1, 0, 1, 1)$  も用いて検索する。しかし、この処理を全ての次元に対して行くと、処理時間が膨大になってしまうため、処理の対象とする次元数に閾値bを設ける。これにより、最も多い場合で、検索に用いるビットベクトル $u'$ の数を $2^b$ 個に抑えている。

[0044] [数5]

$$|q_j| \leq e$$

となる次元数がb個より多くなる場合、次元のインデックスの大きいものからb個採用する。

以上の処理を行うことで、ハッシュ表に登録されている物体IDに投票を行い、最も得票数の多い物体を認識結果とする。

[0045] 野口らの手法では、処理を効率化するために図1に示すような複数の識別器を直列に並べた構成とする。この多段階化の構成は、本願発明にも適用可能である。第s段では、 $b = s-1$ とした識別器を用いる。質問画像をこの識別器にかけた場合の処理について具体的に説明する。まず第1段では、 $b=0$ としたときのビットベクトルを用いて認識する。このとき、得票数が一位の物体と他の物体との得票数の差が十分に大きければ、この時点で得票数一位の物体を認識結果として出力する。十分な得票数差が得られなければ、第2段の識別器に移り、 $b=1$ としたときのビットベクトルを用いて認識する。この処理を繰り返すことにより質問画像を認識する。この識別器の多段階化による認識

手法の特徴は、識別器の段が移ったとき、前の段で検索に用いられたビットベクトル以外のビットベクトルを用いて認識を行うという、差分探索性があることである。このため、最終段の識別器まで移ったとしても、最初から $2^b$ 個のビットベクトルを用いて認識するのとほぼ同じ処理時間で認識できる。

### 2-3. 野口らの手法の課題

[0046] 野口らの手法では、認識率を確保するために、ビットベクトルの次元 $d$ の値を大きくする必要がある。しかし、 $d$ の値を大きくすればハッシュ表の大きさも指数的に増加する。

また $H_{\text{size}} = 2^d$ とした予備実験により、約一千万個のPCA-SIFT特徴ベクトルを野口らの手法でハッシュ表にマッピングしてみたところ、 $d=24$ のとき65%以上、 $d=28$ のとき96%以上のハッシュインデックスが、一つの特徴ベクトルにも対応しない、またはリストが削除された状態となっていることが分かった。以上から、野口らの手法には、ハッシュ表の空間効率の観点から、なお改善の余地があるといえる。

この課題に対処すべく、この発明ではハッシュ表と比べると、はるかに空間効率の良いデータ構造として、偽陽性を許容することによりデータの容量を圧縮するデータ構造を用いることにした。前記データ構造の具体的な構成の一例は、ブルーム・フィルタであり、前記連想データ構造の具体的な構成の一例は、ブルーミア・フィルタである。

### 3. ブルーム・フィルタとブルーミア・フィルタ

[0047] 本節では、この発明に係る連想データ構造として適用可能なブルーミア・フィルタ（前記非特許文献6参照）と、その基礎となるブルーム・フィルタ（詳細は、例えば、B. H. Bloom: "Space/Time Trade-offs in Hash Coding with Allowable Errors", Commun. ACM, Vol. 13, No. 7, pp. 422-426 (1970)参照）について説明する。

#### 3-1. ブルーム・フィルタ

[0048] ブルーム・フィルタは、平衡2分探索木やハッシュ表と比べてはるかに空間効率の良いデータ構造である。このブルーム・フィルタは、あるデータ集合

と要素が与えられたとき、この要素がデータ集合のメンバであるか否かを調べるために用いられる。この手法には、ある要素がデータ集合のメンバではないにもかかわらず、集合のメンバであると判断される偽陽性(False Positive)の可能性が高くなるという問題点や、データ集合から元の要素を取り出すことができないという問題点がある。一方この手法には、メモリ量を増やすことなくデータ集合にいくらかでも要素を追加することができるという利点がある。もちろん要素を追加すればするほど偽陽性の可能性は高くなるが、同じメモリ量で保存できる要素数は、平衡2分探索木やハッシュ表と比べると圧倒的に多い。

[0049] この発明では、この空間効率の良さを利用し、ブルーム・フィルタを特徴ベクトルの圧縮表現に用いる。この明細書では、以後データ集合の要素を特徴ベクトルとして話を進める。以下に、ブルーム・フィルタへの特徴ベクトル登録方法と特徴ベクトルの認識方法の具体的な処理について説明する。

[0050] 図2に特徴ベクトルの登録処理の流れを示す。まず、空のブルーム・フィルタとして $m$  bitの配列を用意し、全てのbitを0で初期化する。以下では、 $m$ を“TableSize”と呼ぶこととする。次に、 $k$ 個のハッシュ関数を用意し、それぞれのハッシュ関数に対して、特徴ベクトルをキーとしてハッシュ値を計算する。ここで、得られるハッシュ値は $1\sim m$ の整数値とする。そして、得られた $k$ 個のハッシュ値  $x_1, x_2, \dots, x_k$ を基に、ブルーム・フィルタの $x_i$  ( $i=1, 2, \dots, k$ )番目のbitを1にすることで特徴ベクトルを登録する。特徴ベクトルの認識処理も探索処理と同様に、特徴ベクトルをキーとしてそれぞれのハッシュ関数からハッシュ値を得る。そして、得られたハッシュ値に対応するbitが全て1になっていれば、データ集合にその特徴ベクトルが含まれているとする。

### 3-2. ブルーミア・フィルタ

[0051] この発明で、ブルーミア・フィルタ (Bloomier Filter) は、複数のブルーム・フィルタを用いることで、登録した特徴ベクトルと関連する値を連想させるために用いられるデータ構造である。この手法にも、偽陽性の可能性が

あるという問題点や、元の特徴ベクトルをデータ集合から取り出すことができないという問題点があるものの、ハッシュ表などと比べると、圧倒的に空間効率が良いという利点がある。この発明では、この性質を利用し、ブルーミア・フィルタを特定物体認識に用いる。以下に、ブルーミア・フィルタの動作について具体的に説明する。

[0052] ブルーミア・フィルタで連想させる値が0と1の二種類のみについて説明する。まず2つのブルーム・フィルタ XとYを用意する。次に、特徴ベクトルに連想させたい値が0である場合、ブルーム・フィルタ Xに特徴ベクトルを登録しておき、連想させたい値が1である場合、ブルーム・フィルタ Yに特徴ベクトルを登録しておく。これによりある特徴ベクトルを認識させたとき、ブルーム・フィルタ Xに存在すれば、その特徴ベクトルから連想される値は0である可能性が高く、また、ブルーム・フィルタ Yに存在すれば、連想される値は1である可能性が高いとすることができる。

[0053] この発明では、上記のブルーミア・フィルタの動作を利用し、特定物体を認識するために用いる物体IDを連想させることを考える。物体IDをn bitで表現するとしたとき、bitごとに二つのブルーム・フィルタを用意することで、これを実現させる。これにより、N個の物体を識別するために、物体ごとにブルーム・フィルタを用意するとN個のブルーム・フィルタが必要となるが、ブルーミア・フィルタを用いることで、 $\log_2 N$ 個のブルーム・フィルタで物体を識別することができる。

#### 4. この発明の手法

[0054] 本節では、ブルーミア・フィルタを用いた特定物体認識手法について述べる。この発明でも、野口らの手法と同様に、画像からPCA-SIFTによって求めた特徴ベクトルを用いる。まず、ブルーミア・フィルタへ特徴ベクトルを登録するデータベース作成処理について説明する。次に、ブルーミア・フィルタを用いた物体認識手法について具体的に説明する。

##### 4-1. データベース作成

[0055] 本節では、ブルーミア・フィルタへ特徴ベクトルを登録するデータベース

作成処理について説明する。まず、物体を識別するために用いる物体IDをn bitで表現するとしたとき、2n個のブルーム・フィルタを用意する。このとき0を連想させるブルーム・フィルタを $X_1, X_2, \dots, X_n$ 、1を連想させるブルーム・フィルタを $Y_1, Y_2, \dots, Y_n$ とする。それぞれのブルーム・フィルタのTableSizeは、

[0056] [数6]

$$a \times M_f^g [\text{bit}] \quad (3)$$

とする。ここで、

[0057] [数7]

$$M_f^g (f \in \{0, 1, \dots, n\}, g \in \{0, 1\})$$

は物体IDのf番目のbitがgである物体から得られる特徴ベクトルの総数、aは一特徴ベクトルを何bitで保存するかを表す値である。

[0058] 具体例を示すと、ブルーム・フィルタ  $X_i$  の場合、物体IDをbit表現したときにi番目のbitが0である全ての物体から得られる特徴ベクトル数がMの値となり、

[0059] [数8]

$$a \times M_i^0 \quad (4)$$

が $X_i$ のTableSizeとなる。

[0060] 次に、物体IDを連想させるために、n個のブルーム・フィルタに対して特徴ベクトルを登録していく。例えば、2bitで物体IDを表現するとき、物体IDの3を“10”で表現したとすると、bit列の1 bit目が“1”、2 bit目が“0”であるため、 $Y_1$ と $X_2$ のブルーム・フィルタに物体ID3の物体から得られる特徴ベクトルを保存する。以下に具体的な処理を説明する。

[0061] この発明では野口らの手法と同様に、特徴ベクトルのd次元を用いて、ビットベクトルuを作成する。データベースに登録する特徴ベクトル全てをビットベクトルで表現したとき、同じビットベクトルが閾値c個以上あれば、そのビットベクトルに変換される特徴ベクトルは、認識にはあまり有効でないと考え、データベースに登録しない。それ以外の場合は、求めたビットベクトルuをキーとし、k個のハッシュ関数を用いて、ブルーム・フィルタのどのビットを1にするかを定める。

[0062] この実施形態では、k=8とし、ハッシュ関数にインターネット上のGeneral Purpose Hash Function Algorithms, [online]、平成21年6月3日検索、<URL : <http://www.partow.net/programming/hashfunctions/index.html#RSHashFunction>. > で紹介されている8個のハッシュ関数を用いる。ただし、これは一例にすぎず、他のハッシュ関数を用いてもよい。

[0063] 以上の処理を全ての特徴ベクトルに対して行い、データベースを作成する。この発明では、データベースから元の特徴ベクトルを取り出すことができないが、ブルーム・フィルタの性質上、元の特徴ベクトルをそのまま保存するよりも、かなりメモリ量を圧縮して特徴ベクトルを表現することができる。

#### 4-2. 物体認識

[0064] 本節では、ブルーミア・フィルタを用いた物体認識手法について説明する。まず、提案する物体認識手法の処理の流れを説明する。この手法では、検索質問画像から得られる特徴ベクトルqを用い、物体IDのi (i=1, 2, ..., n) 番目のbitが0か1かを求めるために、 $X_i$ と $Y_i$ のブルーム・フィルタにqに対応する特徴ベクトルが登録されているかを調べる。このとき、 $X_i$ に登録されていれば、物体IDのi番目のbitを0とし、 $Y_i$ に登録されていれば、物体IDのi番目のbitを1とする。そして、全てのbitに対して値を求め、最終的に求まった物体IDの物体に投票を行う。この処理を検索質問画像から得られる特徴ベクトル全てに対して行い、最も得票数の多かった物体を認識結果とする。以下に具体的な処理について説明する。

- [0065] 質問画像から得られる特徴ベクトルは、一般に各次元の値がデータベース作成に用いる特徴ベクトルのものと異なったものとなる。このためこの実施形態でも、野口らの手法と同様に、各次元の値の許容変動幅 $e$ を設け変動に対処する。また、この処理を行う次元数を閾値 $b$ 個以下に制限し、次元数が $b$ を上回るときには、次元のインデックスの大きいものから $b$ 個を採用する。そしてこれらのビットベクトルを用いて、物体IDの $i$  ( $i=1, 2, \dots, n$ ) 番目のbitが0か1かを求める。
- [0066] このとき、ブルーム・フィルタ  $X_i$ に存在していれば、物体IDの $i$ 番目のbitを0とし、ブルーム・フィルタ  $Y_i$ に存在していれば、物体IDの $i$ 番目のbitを1とする。 $X_i$ と $Y_i$ の両方に含まれていない場合は、このビットベクトルをキーとした特徴ベクトルはデータベースに登録されていないとし、 $i$ 番目以降の処理を打ち切る。
- [0067] この処理では、 $X_i$ と $Y_i$ の両方に含まれていた場合に問題が起こる。これは、どちらかのブルーム・フィルタが偽陽性を引き起こし、実際には登録されていないにもかかわらず、登録されていると判断されるために起こる。この発明では、このような場合、両方の可能性を試すことにより対処する。具体的には、物体IDの $i$ 番目のbitが0となる物体IDと1となる物体IDの両方を投票の対象とする。但し、全てのbitに対して上記の処理を行うと誤投票が増加してしまうため、この発明では、偽陽性に対処する物体IDの次元数に閾値 $t$ を設け、投票する物体ID数を $2^t$ 個に制限する。対処する次元数が $t$ 個を超える場合、次元の小さいものから $t$ 個採用する。
- [0068] 上記の処理を繰り返し、求めた物体IDの物体に投票する。この処理を検索質問画像から得られる全ての特徴ベクトルに対して行い、最終的に最も得票数の多かった物体を認識結果とする。
- [0069] またこの実施形態では、認識処理の効率化を図るために、野口らの手法と同様に、第 $s$ 段が $b = s-1$ とした識別器を用いて認識を行う。これによって、十分に他の物体との得票数差が得られれば、途中の段で処理を打ち切ることができるため、認識処理を効率的に行うことができる。

#### 4-3. 誤り検出

[0070] この発明では、上記の処理で作成したデータベースに加え、投票時に投票誤りを検出するための処理も提案する。まず、データベース作成の際には、上記の通常のブルーミア・フィルタに加え、誤り検出用のブルーミア・フィルタも作成する。これは、ある検索質問画像から得られた特徴ベクトルを用いて認識処理で求めたIDに、本当に投票を行ってよいかを決めるための簡易的な誤り検出器である。

[0071] この誤り検出ブルーミア・フィルタを以下のように作成する。まず、二つのブルーム・フィルタ  $P_0$ ,  $P_1$ を用意する。そして、物体IDを2bit表現したとき、1の数が偶数個の物体IDから得られる特徴ベクトルを $P_0$ に登録し、1の数が奇数個の物体IDから得られる特徴ベクトルを $P_1$ に登録する。それぞれのブルーム・フィルタのTableSizeは、式(4)と同様に与えられる。

[0072] 認識処理では、求めた物体IDの物体に投票を行うかどうかを、誤り検出ブルーミア・フィルタを用いて決める。このとき、誤り検出ブルーミア・フィルタに登録されていなかった場合や、両方に登録されていた場合、求めた物体IDは誤ったIDと判断し、投票を行わない。

#### [0073] 5. データベース作成処理、認識処理の具体例

前節で述べたデータベース作成処理、認識処理についての具体例を本節に示す。なお、説明を理解しやすくするため、本節では非常に小さな物体IDのbit数、特徴ベクトル数を例示しているが、実際に本願発明が適用される場面では、後述する実験例、あるいはそれよりさらに大規模なデータベースが処理の対象となる。

#### [0074] 5-1. データベース作成処理

いま、物体IDが「3」の物体から得られる特徴ベクトルをデータベースに登録する処理について具体的に述べる。まず特徴ベクトルに登録するブルーム・フィルタを決める。ここで、物体ID3を2bitで表現するとしたとき、物体ID3に10というbit列を割り当てたとする。例では、bit列の1番目の数字が1、2番目の数字が0であるので、bit列の1番目の数字に1を連想させるブルーム・

フィルタ  $Y_1$ と、bit列の2番目の数字に0を連想させるブルーム・フィルタ  $X_2$ に特徴ベクトルを登録する。

また、物体IDをbit列で表現したとき、1が奇数個現れるので、誤り検出ブルーム・フィルタ  $P_1$ にも特徴ベクトルを登録する。次に、物体ID3の物体より得られる特徴ベクトルを、図2に示すような手順で登録する。

[0075] まず、物体ID3の物体から得られる特徴ベクトルの1次元目からd次元を用い、ビットベクトルを作成する。特徴ベクトルの各次元の値が非負であるとき1とし、負であるとき0に変換する。具体的に説明すると、今物体ID3の物体から得られる特徴ベクトルとして、

$$P_1 = (24, -500, 249, 32),$$

$$P_2 = (239, 123, -11, -57),$$

...

があるとき、 $d=3$ とすると、 $P_1, P_2, \dots$ はそれぞれ低次元側（左側）から3ビットの各要素を0または1に符号化してなるビットベクトル、

$$u_1 = (1, 0, 1),$$

$$u_2 = (1, 1, 0),$$

...

に変換される。

[0076] このとき、同じビットベクトルに変換される特徴ベクトルがc個以上あった場合、このビットベクトルに変換される特徴ベクトルは、類似しているため認識にはあまり有効でないと考え、以後の処理に用いない。

[0077] 次に、ビットベクトルをキーとしてk個のハッシュ関数からそれぞれハッシュ値を得る。そして、得られたハッシュ値に対応するブルーム・フィルタのbitを1にする。ここで、対象のブルーム・フィルタは、前述のごとく $Y_1, X_2, P_1$ であり、何も登録されていない初期後の状態で、各ブルーム・フィルタは、全ての要素が0（ゼロ）である。

ここで、kを $k=3$ とし、ブルーム・フィルタのTableSizeを $m=8$ としたとき、ビットベクトル $u_1, u_2$ を登録する例を図23に示す。

[0078] 図23で、まず、ビットベクトル $u_1$ をキーとして、前述の $k$ 個のハッシュ関数から $k=3$ 個のハッシュ値を求める。いま、求まったハッシュ値が1, 2, 4であったとき、(00000000)に初期化されたブルーム・フィルタ  $Y_1$ ,  $X_2$ ,  $P_1$ の1, 2, 4番目のbitを1にする。 $u_1$ 登録後の $Y_1$ ,  $X_2$ ,  $P_1$ は、いずれも(11010000)になる(図23(a)参照)。

[0079] 続いて $u_2$ をキーとし、前述の $k$ 個のハッシュ関数から2, 5, 7というハッシュ値が求まったとすると、既に、 $u_1$ が登録された $Y_1$ ,  $X_2$ ,  $P_1$ のベクトル(11010000)のうち2, 5, 7番目のbitを1にする。 $u_2$ 登録後の $Y_1$ ,  $X_2$ ,  $P_1$ は、いずれも(11011010)になる(図23(b)参照)。

[0080] 以上の処理を、物体ID3の物体から得られる全ての特徴ベクトルについて繰り返し、ブルーム・フィルタ $Y_1$ と $X_2$ と $P_1$ に登録する。

さらに、他の物体IDについても同様の処理を行う。例えば、物体ID4について、11というbit列を割り当てたとする。bit列の1番目および2番目の数字がいずれも1であるので、bit列の1番目の数字に1を連想させるブルーム・フィルタ  $Y_1$ と、bit列の2番目の数字に1を連想させるブルーム・フィルタ  $Y_2$ と、偶数ビットの誤り検出ブルーム・フィルタ  $P_0$ とに、物体ID4の物体から得られる全ての特徴ベクトルを登録する。

以上のようにして、各物体の各特徴ベクトルについて登録を行い、データベースを作成する。

[0081] 5-2. 認識処理

続いて、認識処理の具体例について述べる。まず、検索質問画像から得られる特徴ベクトルを用いて、全てのブルーム・フィルタに対して検索を行う。

いま、検索質問画像から得られる特徴ベクトル(検索質問特徴ベクトル)として、

$$q_1 = (31, -480, 220, 49),$$

$$q_2 = (239, 20, 113, -82),$$

...

があるとき、データ登録処理時と同様に、特徴ベクトルからビットベクトル

$$u_1' = (1, 0, 1),$$

$$u_2' = (1, 1, 1),$$

...

を作成する。

[0082] このビットベクトルをキーとして、前述の $k=3$ 個のハッシュ関数からそれぞれハッシュ値を求める。物体IDのビット列に対応する各ブルーム・フィルタ  $X_1, X_2, Y_1, Y_2$  について、対応するbitが全て1になっているかを調べる。もし、全てのbitが1となっていれば、キーの特徴ベクトルがそのブルーム・フィルタに登録されているとする。

[0083] 具体的に述べると、ブルーム・フィルタ  $Y_1$  が (11011010) となっているとする。

検索質問の特徴ベクトル $q_1$ から得られたビットベクトル $u_1'$ についてハッシュ値を求めると、 $u_1'$ は登録時の $u_1$ と同じ値であるから、登録処理と同様のハッシュ値1, 2, 4が得られる。ブルーム・フィルタ  $Y_1$ の1, 2, 4番目のbitをみると全てが1になっている。この場合、 $q_1$ はブルーム・フィルタ  $Y_1$ に登録されていると判断する。

[0084] 但し、一般に、ブルーム・フィルタでは、登録されていないにも関わらず、登録されていると判断されることがある。今 $q_2$ が実際にはブルーム・フィルタに登録されていない特徴ベクトルであると仮定する。 $q_2$ より求めたビットベクトル $u_2'$ からハッシュ値を求めると2, 4, 5であったとする。このハッシュ値よりブルーム・フィルタ  $Y_1 = (11011010)$ の対応する第2, 4, 5 bitを調べると、全て1になっている。このように、実際には登録されていない特徴ベクトルでも、ブルーム・フィルタが偽陽性を起こし、登録されていると判断されることがある。

[0085] 説明を続ける。このように、検索質問の各特徴ベクトルにつき、前述した手順で検索処理を行うことで、物体IDの*i*番目のbitが何であるかを判断する

。ある検索質問の特徴ベクトルがブルーム・フィルタ  $X_i$  に登録されていたとする。この場合、物体IDの  $i$  番目のbitを0とする。一方、その特徴ベクトルが  $Y_i$  に登録されていたとすると、物体IDの  $i$  番目のbitを1とする。仮に、 $Y_1$  と  $X_2$  に特徴ベクトルが登録されていたとすると、物体IDのbit列は「10」となるため、ID3の物体に投票を行う。

[0086] このとき、ブルーム・フィルタが偽陽性を起こし、 $X_i$  と  $Y_i$  の両方に特徴ベクトルが登録されていると判断されたとする。その場合、この発明では両方の可能性を試す。即ち、その検索質問特徴ベクトルについては、図23に示すように、物体IDの  $i$  番目のbitが0の物体と1となる物体の両方に投票を行うこととする。例えば、検索処理の結果、ある検索質問特徴ベクトルが  $X_1$ ,  $Y_1$ ,  $X_2$  に登録されていたと判断されると、物体IDのbit列00と10の2つのID候補を用意する。そして、誤り検出ブルーム・フィルタを参照し、投票してもよい物体IDを決める。即ち、誤り検出ブルーム・フィルタ  $P_0$  に特徴ベクトルが登録されていたとき、前記ID候補のうち物体IDに1が偶数個のID「00」のみ投票を行い、逆に、 $P_1$  に特徴ベクトルが登録されていたとき、物体IDに1が奇数個のID「10」のみ投票を行う（図24参照）。なお、ここでは、ブルーム・フィルタを用いた誤り検出の一例を示したが、誤り検出に代えて、例えば、複数のブルーム・フィルタにより公知の誤り符号訂正の手法を適用できるように構成してもよい。このようにすれば、偽陽性に起因する誤った判断をより低減することができる。

[0087] 上記の検索処理および投票処理を検索質問画像から得られる全ての特徴ベクトルに対して行う。その結果、最終的に最も得票数が多かった物体を認識結果とする。

なお、投票に際しては、票に重みを付けてもよい。例えば、物体IDが  $i$  の物体から抽出されデータベースに登録された特徴ベクトル数が  $N_i$  個のとき、その物体IDへの投票に  $1/\sqrt{N_i}$  の重みを付けて投票するようにしてもよい。

[0088] 5. 実験例

本節では、この発明の有効性を確かめるために、55個の3次元物体を用いた

データセットと、5000個の2次元物体を用いたデータセットに対して行った実験の結果について述べる。

#### 5-1. 実験条件

まず、55個の3次元物体のデータセットについて説明する。図3に本実験で用いた3次元物体の例を示す。本実験では、物体に対して、真正面、真正面から上へ15°、30°の角度から、ウェブカメラで、5°ごとに物体を回転させて撮影した画像を用いた。ウェブカメラは、株式会社ロジクール製、製品型名 Qcam（登録商標） Pro 9000、解像度：640480のものである。このうち、データベース作成用画像として、回転角度が0°、10°、…、350°の画像36枚×3方向の、1物体当たり108枚を用い、残りの画像を検索質問画像とした。データベース作成用画像から得られる特徴ベクトルの数は、約120万個である。

[0089] 次に、5000個の2次元物体のデータセットについて説明する。本実験では、Googleイメージ検索において、“雑誌”、“ポスター”、“表紙”等の検索キーワードで収集した1667枚の画像と、PCA-SIFTのサイトで公開されていた1667枚の画像と、写真共有サイトのFlickrにおいて、“animal”、“birthday”、“food”、“japan”等のタグにより収集した1666枚の画像をデータベース作成用画像とした。

[0090] 図4に例を示す。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640pixel以下になるように縮小した。また、得られる特徴ベクトルが100個以下になる画像も除外した。データベース作成用画像から得られる特徴ベクトルの数は、約1千万個である。検索質問画像としては、次のように用意した。まず、データベースに含まれる画像の内、Googleイメージ検索で収集した画像から100枚、PCA-SIFTのサイトから得た画像から200枚、Flickrから収集した画像から200枚の合計500枚を無作為に選択した。次に、これらの画像をA4の用紙にカラーレーザープリンタ（株式会社沖データ製、製品型名 C5200n）を用いて印刷し、カメラ（キャノン株式会社製、製品型名 EOS Kiss Digital、630万画素、付属レンズ型名：EF-S 18-55mm USM）で撮影した。

[0091] 撮影した画像の例を図5に示す。図5に示す通り、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。さらに、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。

[0092] 両実験において、この発明と野口らの手法を用いてそれぞれ物体の認識を行った場合の、物体の認識率と、質問画像一枚の認識にかかった平均の処理時間と、認識に必要なメモリ量を比較した。但し、処理時間に特徴ベクトルの抽出時間等は含めないものとする。

[0093] 本実験では、この発明と野口らの手法で用いられているパラメータの内、両方のビットベクトルを試す次元数 $b$ 、ハッシュのリスト長 $c$ 、ビットベクトルの次元数 $d$ を変化させて物体の認識率を調べた。ここで各パラメータの変動幅は、 $b=0, 1, \dots, 10$ 、 $c=1, 2, \dots, 10$ 、 $d=24, 28$ である。また両手法において、特徴ベクトルの許容変動幅 $e$ は $e=200$ とした。使用した計算機は、CPUがAMD Opteron8378 2.4GHz、メモリが128GBのものである。

#### 5-2. 実験1：55個の3次元物体を用いた実験

[0094] この発明の手法において、前節で述べたパラメータ以外に、特徴ベクトルの保存容量 $a$ 、偽陽性に対処する次元数 $t$ も変化させて物体の認識率を調べた。 $a$ と $t$ の変動幅は、 $a=8, 16, 24, 32$ 、 $t=1, 2, 3, 4, 5$ とする。野口らの手法のハッシュ表のハッシュサイズは、 $H_{\text{size}}=2^d$ とした。 $d=24$ のときの結果を図6、 $d=28$ のときの結果を図7に示す。横軸に認識率、縦軸に処理時間を示す。

[0095] 図6、7に示す実験結果より、この発明の手法では $b$ の値を大きくすれば物体の認識率が高くなるが、大きくしすぎると認識率が低下し、処理時間も増加することが分かった。これは、 $b$ によって特徴ベクトルの変動に対処することができるが、 $b$ を増加しすぎると誤投票が多くなるため認識率が低下し、さらに投票回数が増加することが、処理時間の増加に繋がっていると考えられる。

[0096] 図8に $b$ を変化させたときの認識率を比較した例を示す。ここで用いたパラ

メータは、 $a=8$ 、 $d=28$ で最も認識率の高かった $c=8$ 、 $t=2$ を用いた。また結果より、この発明の手法の方が野口らの手法と比べて、一枚の認識にかかる処理時間は多くかかるものの、物体の認識率では若干良い結果を得るものがあった。認識処理に時間がかかる原因として、野口らの手法では、一つの検索質問ベクトル $q$ に対してハッシュ表を一度検索するだけで投票を行えるが、この発明の手法では、 $2n$ 個のブルーム・フィルタを検索しなければ投票を行えないことが挙げられる。

[0097] 次に、この発明の手法のパラメータ $a$ を変化させたものと、野口らの手法とで、パラメータ $c$ を変化させたとき、認識処理に必要なメモリ量を比較した結果を $d$ ごとに図9と図10に示す。横軸にパラメータ $c$ 、縦軸にメモリ量を示す。

[0098] 図9と図10に示す実験結果より、この発明の手法の方が野口らの手法と比べて、かなり空間効率がよいことが分かる。これは、野口らの手法ではメモリ量の大部分がハッシュ表のエントリに用いられていることに起因している。

上記の結果より、この発明の手法は野口らの手法と比べ、認識にかかる処理時間は多くかかるものの、認識率とメモリ量の点で優れているといえる。

[0099] 最後に、この発明の手法において、 $t$ を変化させたときの物体の認識率を比較した。実験では、 $a=8$ 、 $d=28$ のときに最も認識率の高かったパラメータ $b=3$ 、 $c=8$ を用いた。図11に実験結果を示す。結果より、 $t$ の値を大きくすれば物体の認識率が高くなるが、大きくしすぎると認識率が低下することが分かった。これは、 $t$ を増加させることで誤投票が増加してしまうことが原因として挙げられる。

### 5-3. 実験2：5000個の2次元物体を用いた実験

[0100] 実験1と同様に、5.1節で述べたパラメータ以外に、パラメータ $a$ 、 $t$ も変化させて物体の認識率を調べた。パラメータの変動幅は、実験1と同じである。また野口らの手法に用いるハッシュ表のサイズも実験1と同じサイズとした。 $d=24$ のときの結果を図12、 $d=28$ のときの結果を図13に示す。横軸に認識

率、縦軸に一枚の認識にかかった平均の処理時間を示す。結果より、実験1と同様に、bを大きくすれば認識率が增加するが、さらに大きくすると認識率が低下し、処理時間も長くなることが分かった。

[0101] 図14にbを変化させたときの認識率を比較した例を示す。ここで用いたパラメータは、a=8, d=28で最も認識率の高かったc=2, t=3を用いた。また実験結果より、処理時間ではこの発明の手法は野口らの手法と比べて、特にd=24のとき大きく劣ることが分かったが、認識率ではd=28のとき、若干良い結果を得るものがあった。d=24で認識率、処理時間共に野口らの手法に比べて劣る原因として、検索処理時に同じビットベクトルに変換される割合が大きくなるため、投票処理時に他の物体との得票数差が開かないことが挙げられる。

[0102] 次に、この発明の手法のパラメータaを変化させたものと、野口らの手法とで、パラメータcを変化させたとき、認識処理に必要なメモリ量を比較した結果をdごとに図15と図16に示す。横軸にパラメータc、縦軸にメモリ量を示す。

[0103] 図15と図16に示す実験結果より、d=24の結果において、a=24, 32のとき野口らの手法より認識に必要なメモリ量が多くなってしまったが、d=28のとき、野口らの手法に比べてこの発明の手法はかなり空間効率がよいことが分かった。d=24, a=24, 32のときにメモリ量が多くなった原因として、この発明の手法では、一つの特徴ベクトルに対して、n個のブルーム・フィルタに登録しなければならないことが原因として挙げられる。

[0104] さらに、同程度のメモリが使える状況での比較を行った。メモリ量がほぼ同じであるこの発明の手法のd=28, a=8, 16と、比較手法1として野口らの手法のd=24と、比較手法2として、野口らの手法においてd=28、 $H_{size} = 2^{24}-1$ としたときの手法の3手法に着目する。これらの3手法でb, cの値を変化させたときの認識率と処理時間の違いを調べた。但しこの発明の手法では、tの値もt=1, 2, 3, 4, 5と変化させて実験を行った。

[0105] 図17に、その実験結果を示す。横軸に認識率、縦軸に1枚の認識にかかっ

た平均の処理時間を示す。実験結果より、使用するメモリ量をほぼ同じにしたとき、この発明の手法は、比較手法と比べて処理時間は多くかかるものの、パラメータによって認識率をかなり向上させることができた。このことからこの発明の手法は、効率よく特徴ベクトル表現を圧縮できているといえる。

[0106] 最後に、この発明の手法において、 $t$ を変化させたときの物体の認識率を比較した。実験では、 $a=8$ 、 $d=28$ のときに最も認識率の高かったパラメータ $b=6$ 、 $c=2$ を用いた。図18にその実験結果を示す。実験1と同様に、 $t$ を大きくすれば認識率が增加するが、さらに大きくすると認識率が低下してしまうことが分かった。

#### 5-4. 誤り検出の効果

[0107] 最後に、前述の誤り検出ブルーミア・フィルタの有効性を調べるために、 $b$ および $t$ を変化させ、誤り検出ブルーミア・フィルタを用いる場合と用いない場合の認識率を比較した。 $b$ を変化させた場合の実験および $t$ を変化させた場合の実験で用いたパラメータは、それぞれ最も認識率の高いものを用いた。

まず、55物体を対象とした実験について述べる。用いたパラメータを表1に、 $b$ に関する実験結果を図19に、 $t$ に関する実験結果を図20にそれぞれ示す。

[0108] 図19と図20に示す実験結果より、 $b$ 、 $t$ 共に大きくすれば認識率が增加するが、大きくしすぎると認識率が低下してしまうことが分かった。これは、値を増加させると、誤投票も増加してしまうことが原因であると考えられる。また実験結果より、特に $t$ を変化させた場合、誤り検出ブルーミア・フィルタを用いた場合、誤った投票を検出することで誤投票が抑えられるため、用いない場合より高い認識率を得ることができると分かった。このため、誤り検出ブルーミア・フィルタを用いることは有効であるといえる。

[0109] 次に2次元物体の結果について述べる。ただし、この実験については、物体数を5000ではなく1万とした。用いたパラメータを表2に、 $b$ に関する実験結果を図21に、 $t$ に関する実験結果を図22にそれぞれ示す。55物体の結果と

同様に、共に大きくすれば認識率が增加するが、大きくしすぎると認識率が低下してしまうことが分かった。また、誤り検出ブルーミア・フィルタを用いた場合、用いない場合より高い認識率を得ることができると分かった。このため、誤り検出ブルーミア・フィルタを用いることは有効であるといえる。

[0110] 前述した実施の形態の他にも、この発明について種々の変形例があり得る。それらの変形例は、この発明の範囲に属しないと解されるべきものではない。この発明には、請求の範囲と均等の意味および前記範囲内のすべての変形とが含まれるべきである。

### 産業上の利用可能性

[0111] 特定物体を認識する手法に、局所特徴量を表す特徴ベクトルの最近傍探索を用いる手法がある。この手法では、特徴ベクトル間の距離を計算するために、特徴ベクトルそのものを記録する必要があるが、一般に物体認識に関与する特徴ベクトルの数は膨大であるため、莫大なメモリ量が必要となる。この問題を解決するため、特徴ベクトルの距離を計算しない枠組みが必要となる。

[0112] この発明は、特徴ベクトルの距離を計算しないメモリ量削減手法として、ハッシュ表などと比べてはるかに空間効率の良いブルーミア・フィルタを用いる手法を提案した。また、平面および3次元特定物体の認識実験を通して、ハッシュ表を用いる手法とこの発明を比較し、その有効性を検証した。

### 符号の説明

[0113] 1 画像データベース

## 請求の範囲

[請求項1] 物体認識用画像データベースに登録されるべき物体が撮影された画像から、その画像の局所的特徴を表す複数の特徴ベクトルを抽出する特徴抽出工程と、

データを登録できかつあるデータが登録済か否かを判定し得るデータ構造に対し、ある物体から抽出された各特徴ベクトルを、その物体に予め割り当てられた識別子が前記特徴ベクトルから連想されるように登録する登録工程とを備え、

各工程をコンピュータが実行し、

前記登録に係る物体は、 $n$ ビット（ $n$ は自然数）の識別子が割り当てられ、

前記登録工程は、前記識別子の各ビットのゼロ値登録用と1値登録用として用意された $2 \times n$ 個の前記データ構造に対し、各特徴ベクトルが抽出された物体の識別子に応じて各ビットのゼロ値登録用と1値登録用の何れか一方のデータ構造に前記特徴ベクトルをそれぞれ登録し、

前記画像データベースは、検索質問としてある物体の画像が与えられたとき同じ物体が登録されているか否かの認識処理を行うべく、前記コンピュータまたは他のコンピュータによってアクセスされ、

前記認識処理は、前記検索質問から複数の特徴ベクトルを検索質問ベクトルとして抽出し、各検索質問ベクトルと同値の特徴ベクトルが各データ構造に登録されているか否かを調べ、すべてのビットについてゼロ値または1値のいずれかに同値の特徴ベクトルが登録されているときは登録された各ビットの値に応じた $n$ ビット値を識別子の候補として得、各検索質問ベクトルにつき得られた候補を統計処理して検索質問に適合する識別子を見出すことを特徴とする物体認識用画像データベースの作成方法。

[請求項2] 各データ構造は、偽陽性を許容することによりデータを格納するメ

モリの容量を削減するものである請求項 1 に記載の作成方法。

[請求項3] 前記登録工程は、所定の規則を適用して前記識別子に係る誤り検出用の値を得、得られた値を前記誤り検出用のデータ構造に登録し、前記認識処理は、得られた識別子の候補と前記誤り検出用データ構造に登録された値とを照合し、両者が一致する場合は前記検索質問ベクトルについての判定結果を統計処理の対象とし、一致しない場合は統計処理の対象としない請求項 1 または 2 に記載の作成方法。

[請求項4] 前記誤り検出用の値が少なくとも 1 ビットからなり、前記誤り検出用データ構造は、ビットのゼロ値登録用と 1 値登録用としてデータ構造が用意されてなる請求項 3 に記載の作成方法。

[請求項5] 前記認識処理は、偽陽性のために、ゼロ値および 1 値登録用のいずれのデータ構造にも同一の検索質問ベクトルが登録されていると判断されるビットの数が予め定められた数を超える場合、その検索質問ベクトルについての判定結果を統計処理の対象から除外する請求項 2 ～ 4 のいずれか一つに記載の作成方法。

[請求項6] 前記認識処理は、前記検索質問ベクトルに関連があると判定される識別子の値に対して所定の得点を与え、前記検索質問ベクトルに関連があると判定される識別子が存しない場合はいずれの識別子にも得点を与えず、前記検索質問から抽出される検索質問ベクトルについての統計処理に基づき最多得点を得た識別子を決定する請求項 1 ～ 5 のいずれか一つに記載の作成方法。

[請求項7] 前記データ構造はブルーム・フィルタである請求項 1 ～ 6 のいずれか一つに記載の作成方法。

[請求項8] 物体認識用画像データベースに登録されるべき物体が撮影された画像から、その画像の局所的特徴を表す複数の特徴ベクトルを抽出する特徴抽出部と、データを登録できかつあるデータが登録済か否かを判定し得るデータ

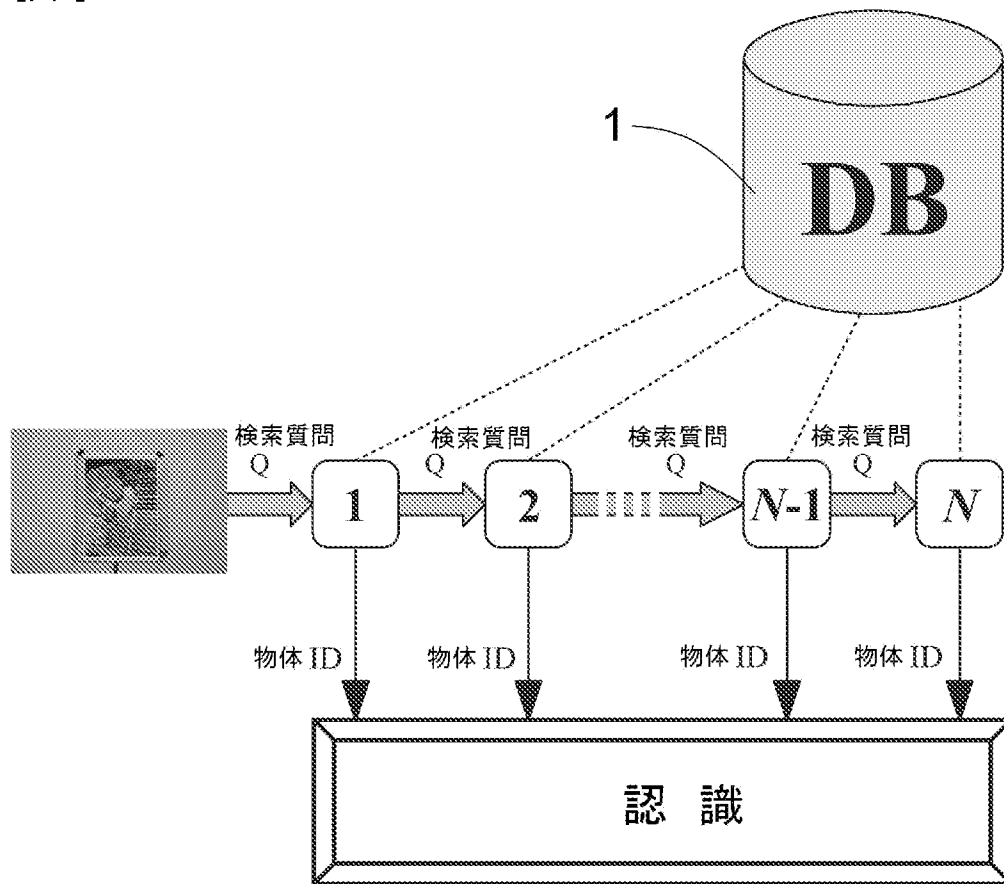
構造体の集合からなるデータ構造部と、  
ある物体から抽出された各特徴ベクトルからその物体に予め割り当てられた識別子が前記特徴ベクトルから連想されるように前記特徴ベクトルを前記データ構造部に登録する登録部とを備え、  
前記登録に係る物体は、 $n$ ビット（ $n$ は自然数）の識別子が割り当てられ、  
前記データ構造部は、前記識別子の各ビットのゼロ値登録用および1値登録用として $2 \times n$ 個のデータ構造体を少なくとも有し、  
前記登録部は、各特徴ベクトルが抽出された物体の識別子に応じて各ビットのゼロ値登録用と1値登録用の何れか一方のデータ構造に前記特徴ベクトルをそれぞれ登録し、  
前記画像データベースは、検索質問としてある物体の画像が与えられたとき同じ物体が登録されているか否かの認識処理を行う認識装置によってアクセスされ、  
前記認識装置は、前記検索質問から複数の特徴ベクトルを検索質問ベクトルとして抽出し、各検索質問ベクトルと同値の特徴ベクトルが各データ構造に登録されているか否かを調べ、すべてのビットについてゼロ値または1値のいずれかに同値の特徴ベクトルが登録されているときは登録された各ビットの値に応じた $n$ ビット値を識別子の候補として得、各検索質問ベクトルにつき得られた候補を統計処理して検索質問に適合する識別子を見出すことを特徴とする物体認識用画像データベースの作成装置。

[請求項9]

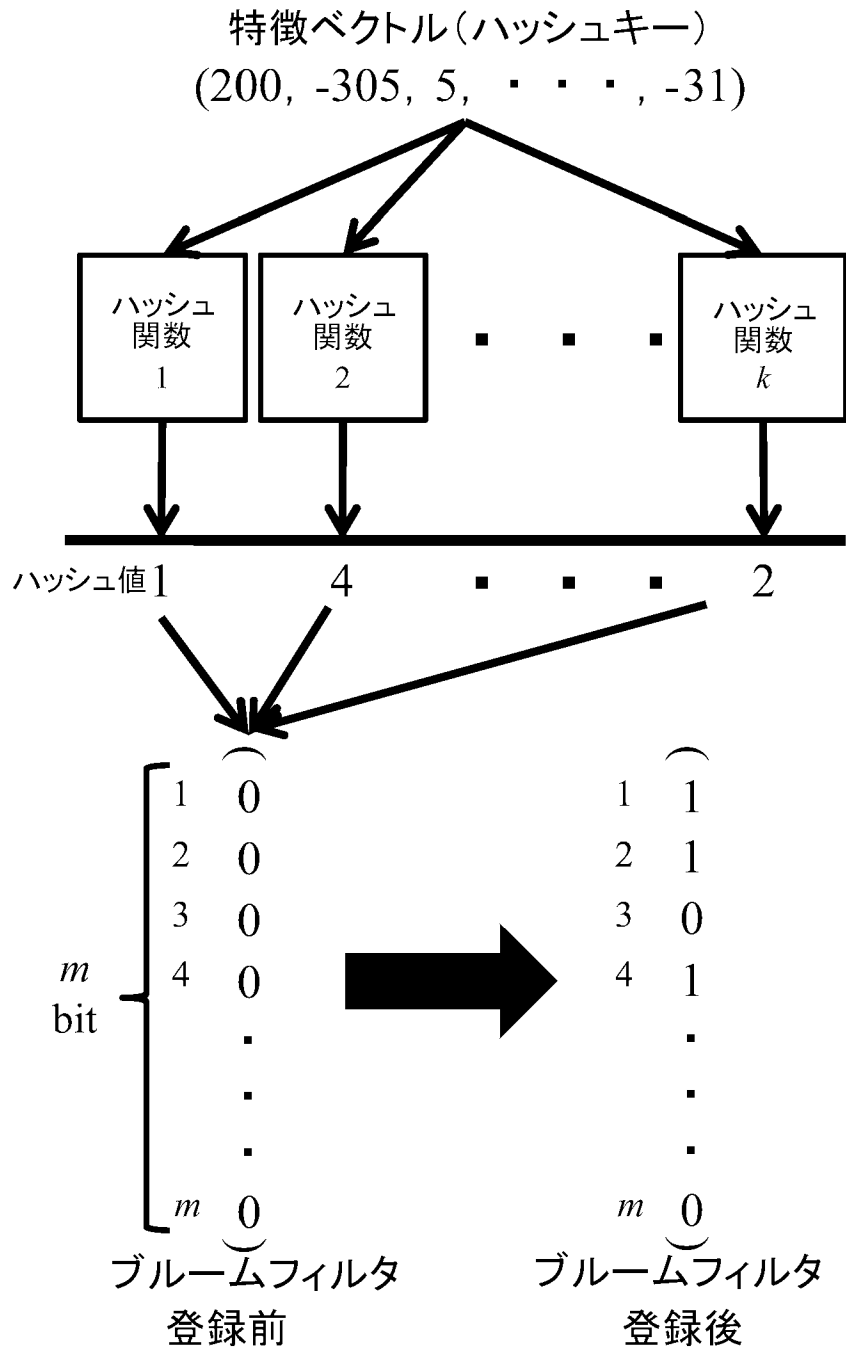
物体認識用画像データベースに登録されるべき物体が撮影された画像から、その画像の局所的特徴を表す複数の特徴ベクトルを抽出する特徴抽出処理と、  
データを登録できかつあるデータが登録済か否かを判定し得るデータ構造に対し、ある物体から抽出された各特徴ベクトルを、その物体に予め割り当てられた識別子が前記特徴ベクトルから連想されるように

登録する登録処理とをコンピュータに実行させ、  
前記登録に係る物体は、 $n$ ビット（ $n$ は自然数）の識別子が割り当てられ、  
前記登録処理は、前記識別子の各ビットのゼロ値登録用と1値登録用として用意された $2 \times n$ 個の前記データ構造に対し、各特徴ベクトルが抽出された物体の識別子に応じて各ビットのゼロ値登録用と1値登録用の何れか一方のデータ構造に前記特徴ベクトルをそれぞれ登録し、  
前記画像データベースは、検索質問としてある物体の画像が与えられたとき同じ物体が登録されているか否かの認識処理を行うべく、前記コンピュータまたは他のコンピュータによってアクセスされ、  
前記認識処理は、前記検索質問から複数の特徴ベクトルを検索質問ベクトルとして抽出し、各検索質問ベクトルと同値の特徴ベクトルが各データ構造に登録されているか否かを調べ、すべてのビットについてゼロ値または1値のいずれかに同値の特徴ベクトルが登録されているときは登録された各ビットの値に応じた $n$ ビット値を識別子の候補として得、各検索質問ベクトルにつき得られた候補を統計処理して検索質問に適合する識別子を見出すことを特徴とする物体認識用画像データベースの作成処理プログラム。

[図1]



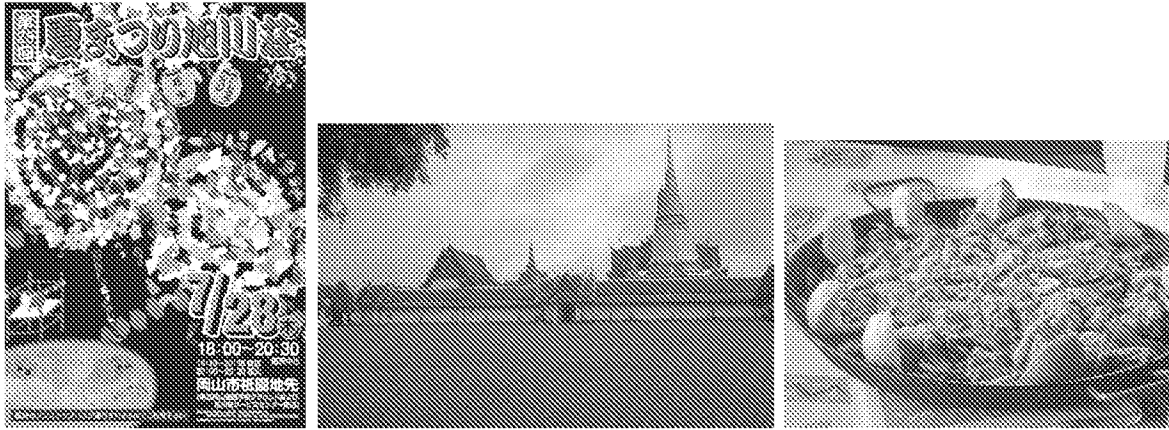
[図2]



[図3]



[図4]

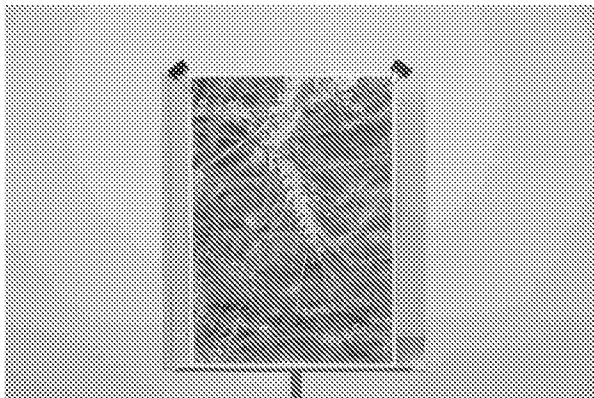


(a)

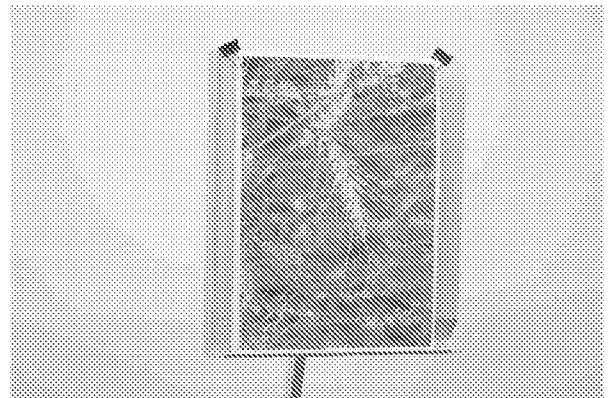
(b)

(c)

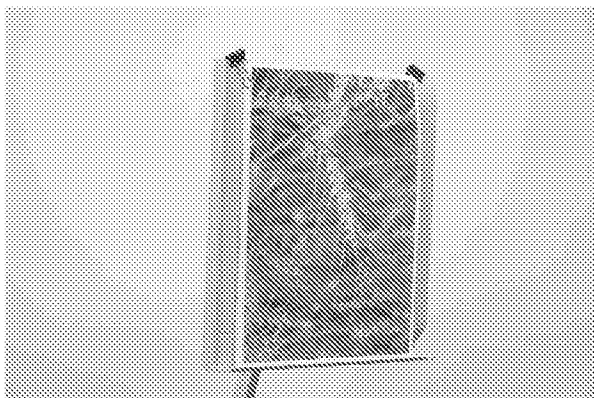
[図5]



(a) 撮影角度90°



(b) 撮影角度75°

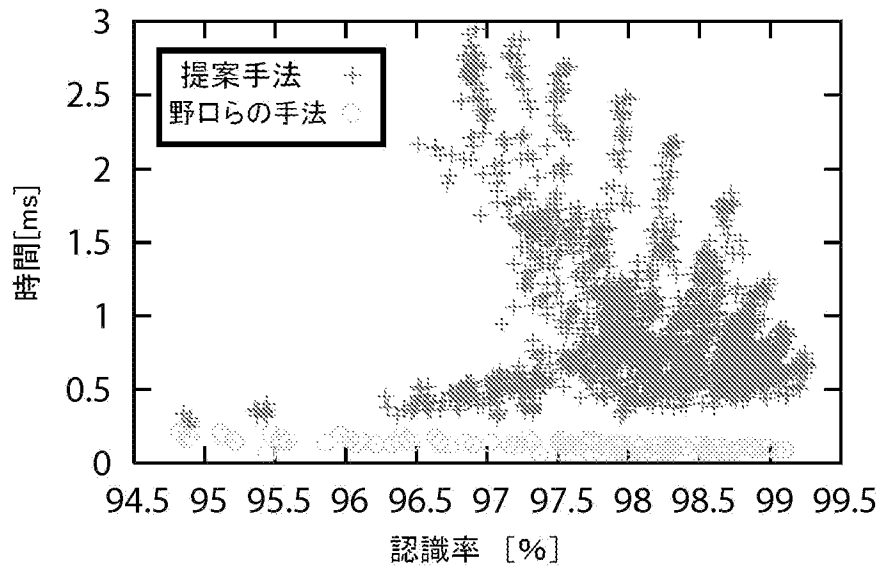


(c) 撮影角度60°

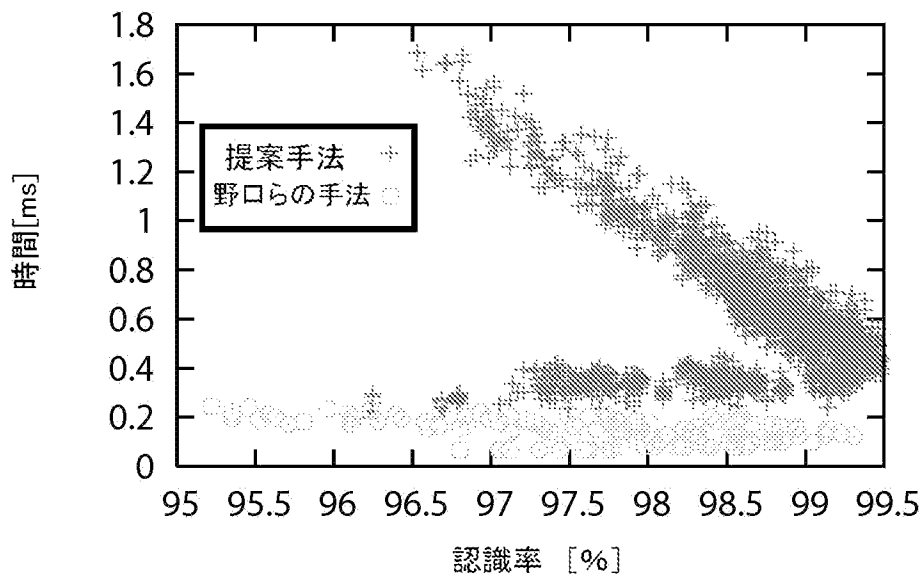


(d) 撮影範囲一部

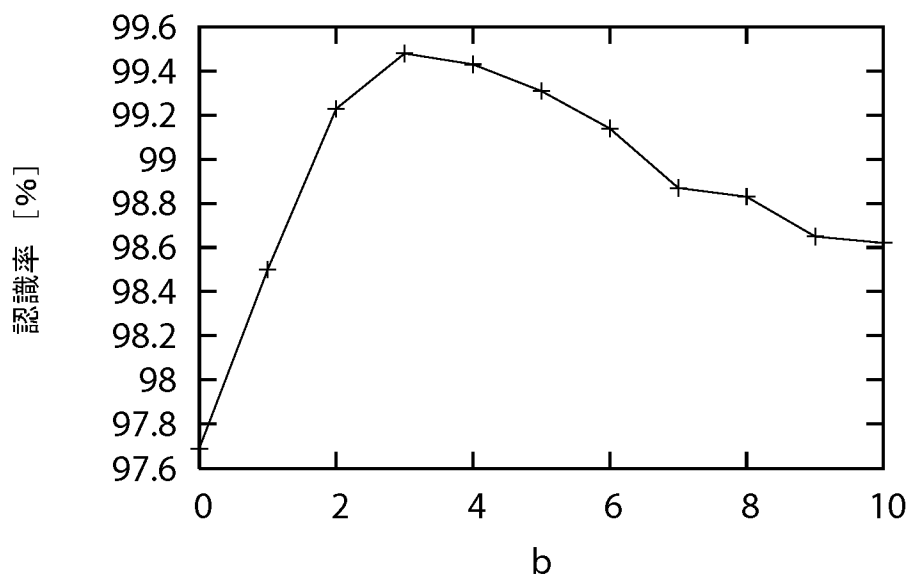
[図6]



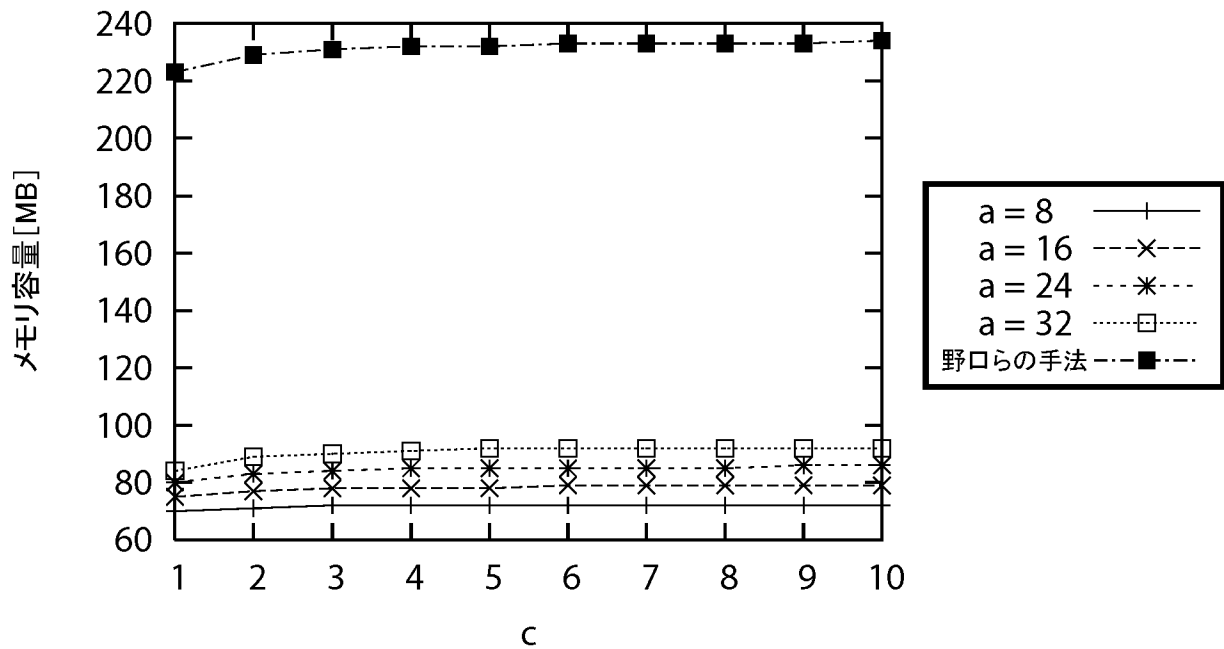
[図7]



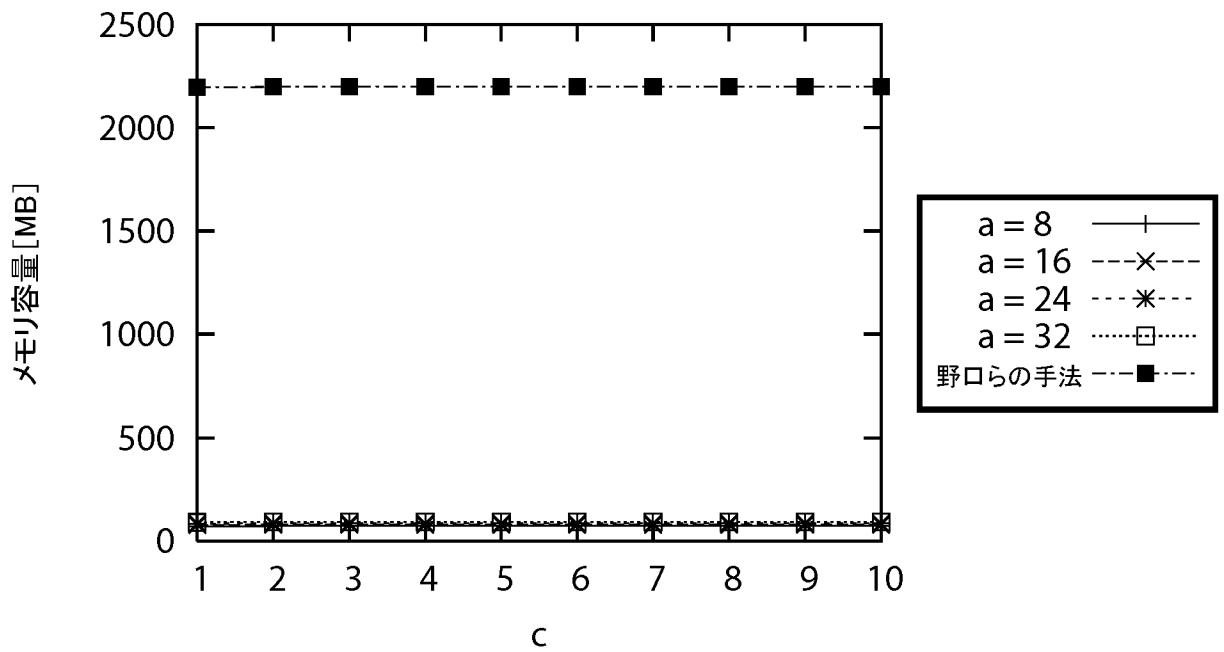
[図8]



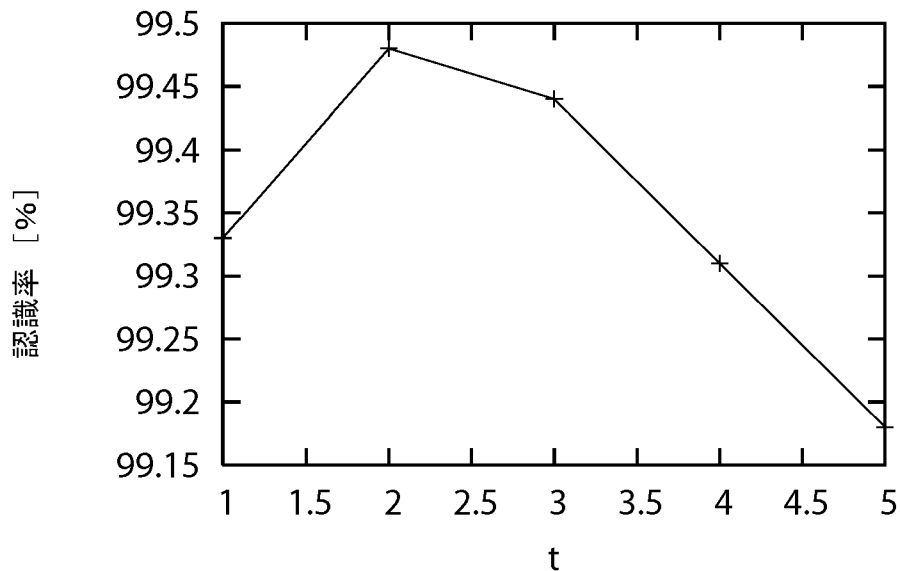
[図9]



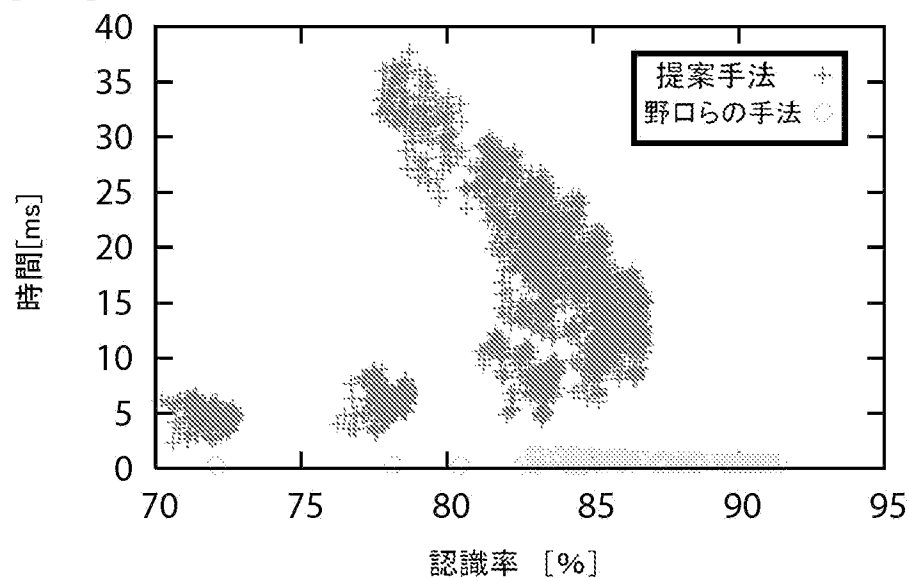
[図10]



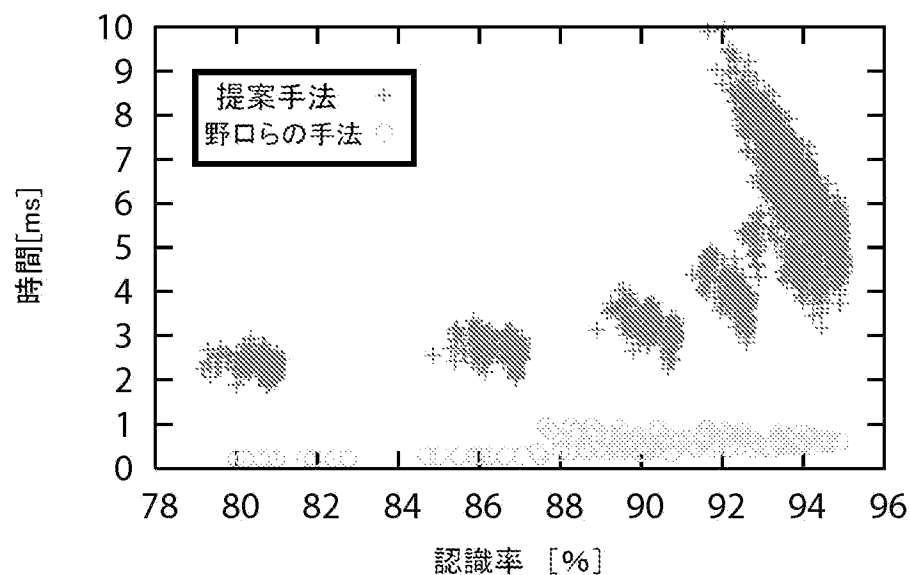
[図11]



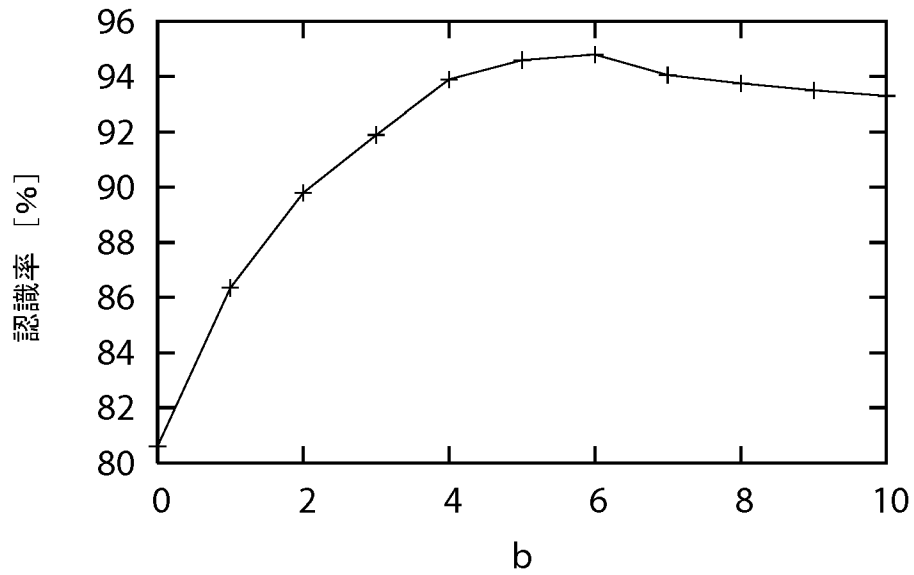
[図12]



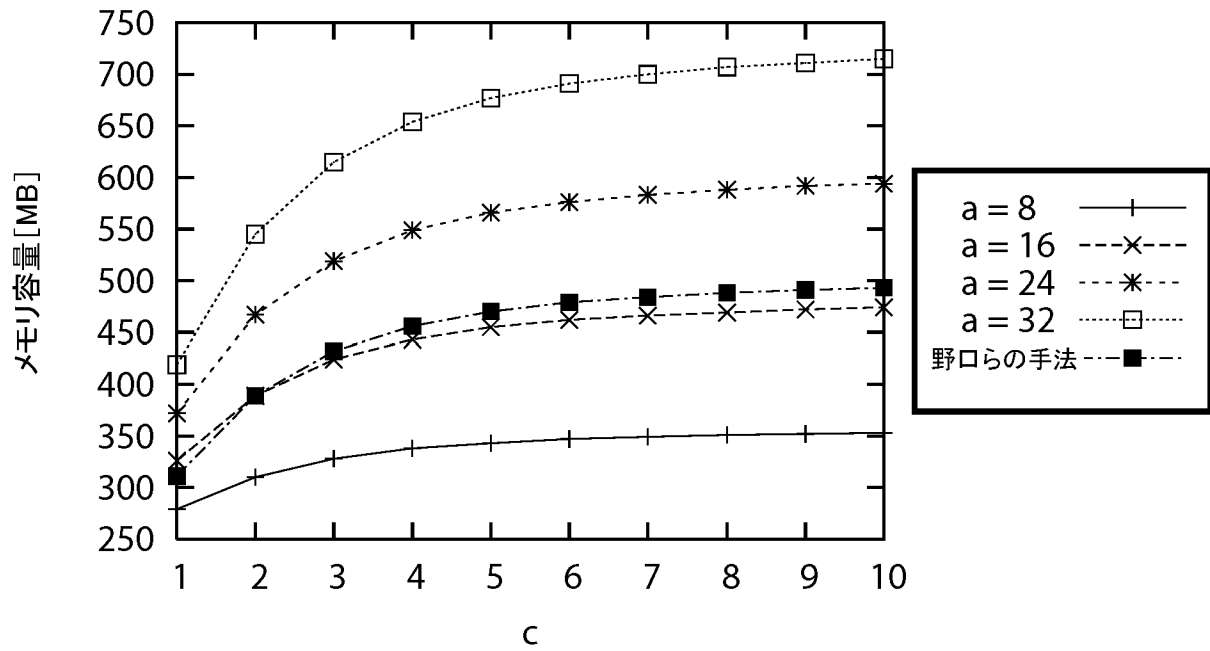
[図13]



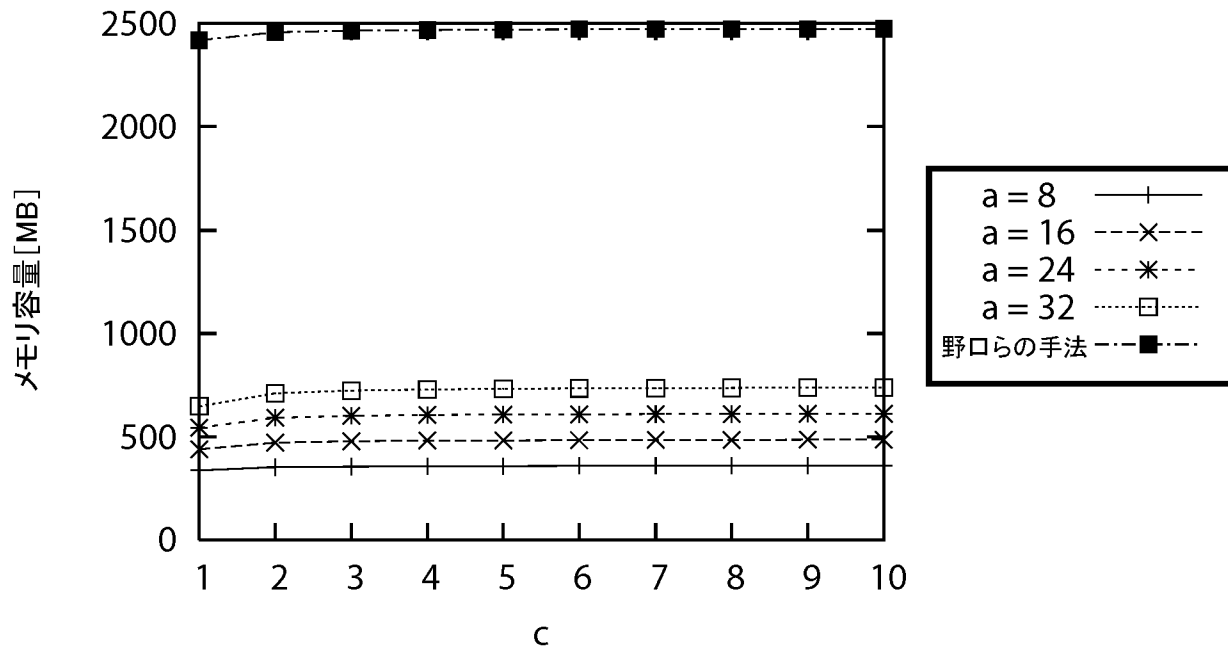
[図14]



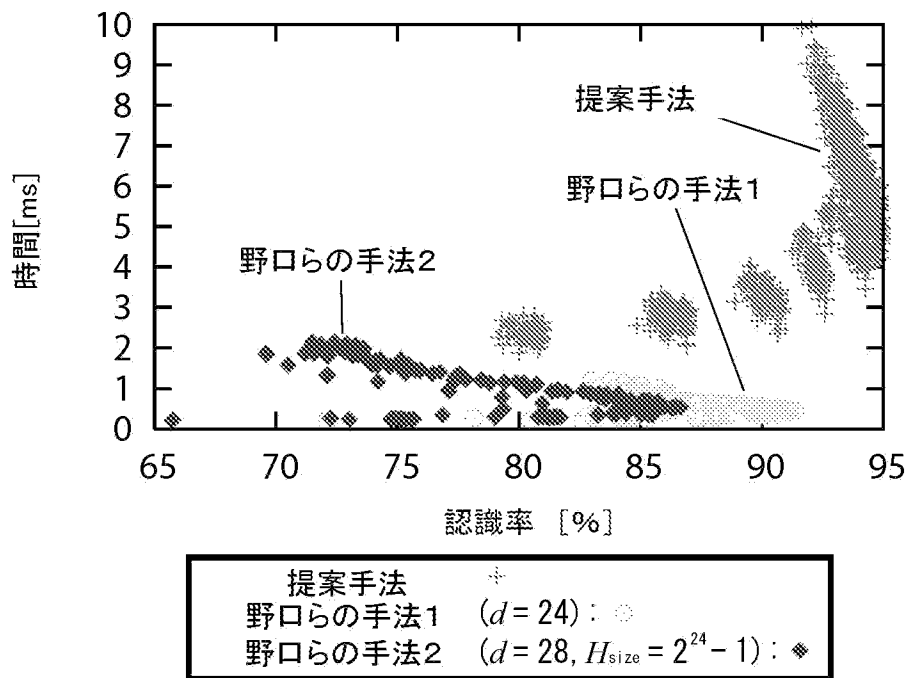
[図15]



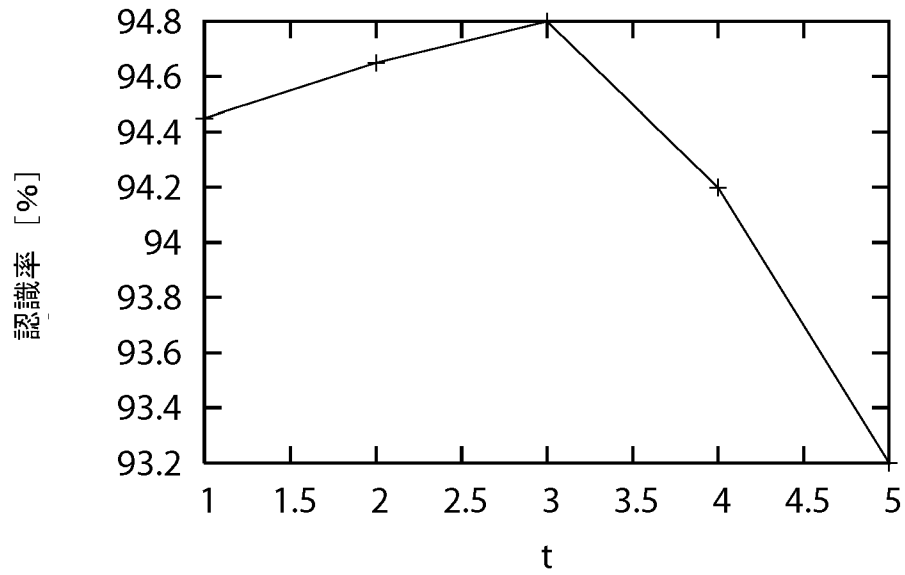
[図16]



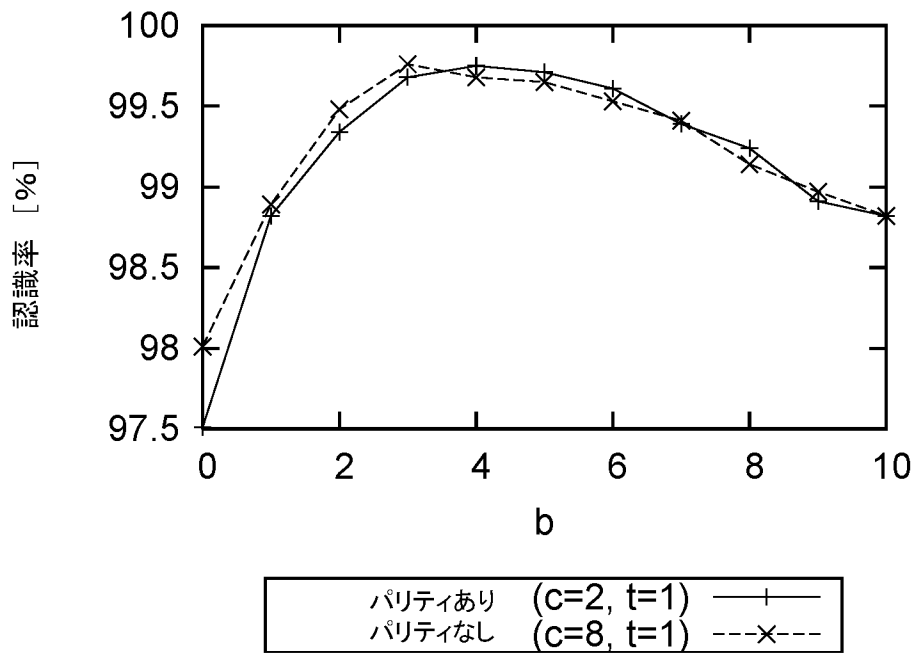
[図17]



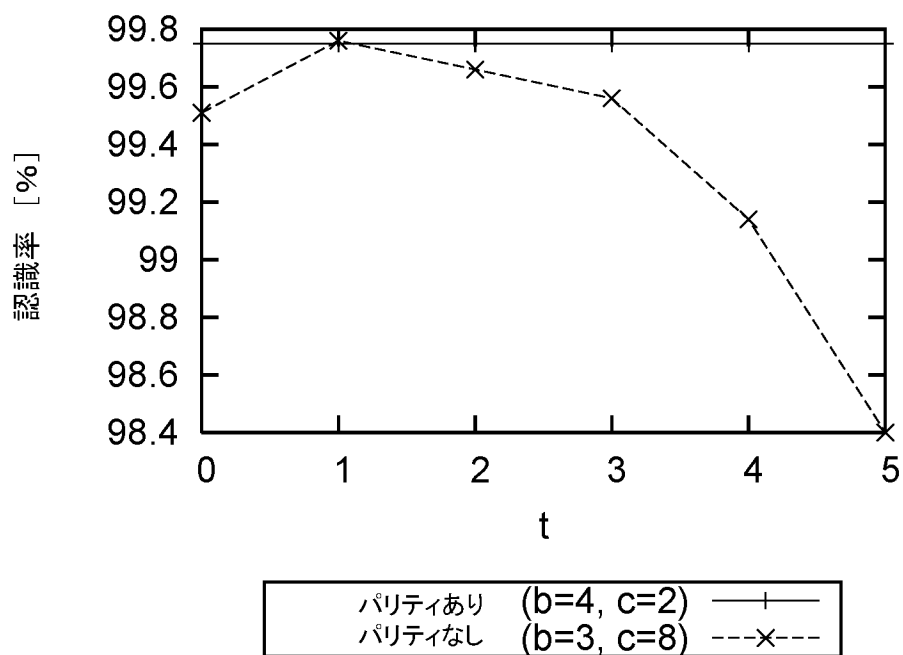
[図18]



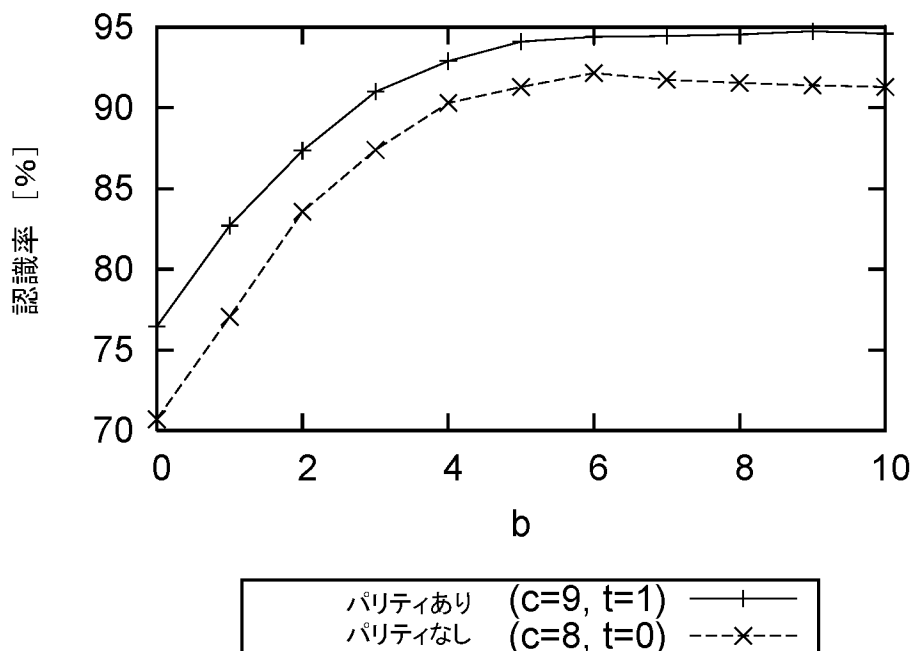
[図19]



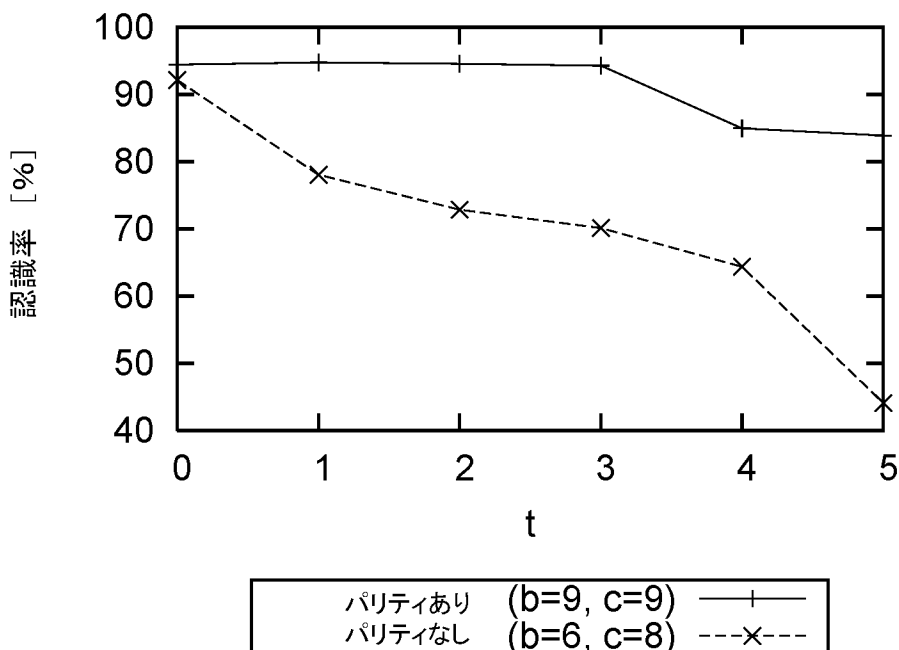
[図20]



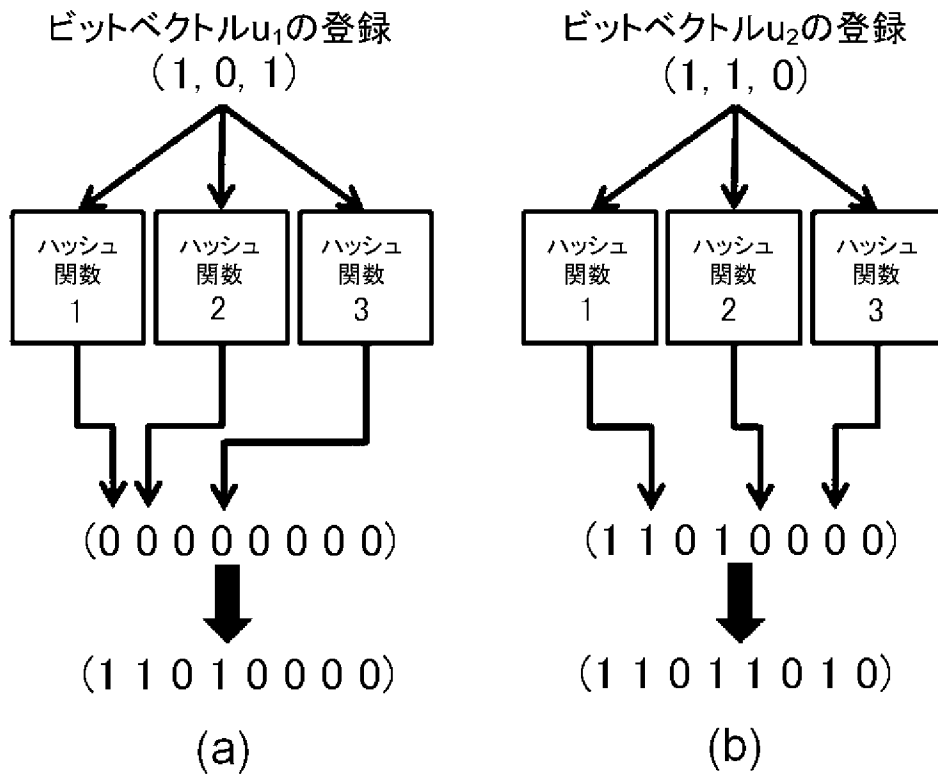
[図21]



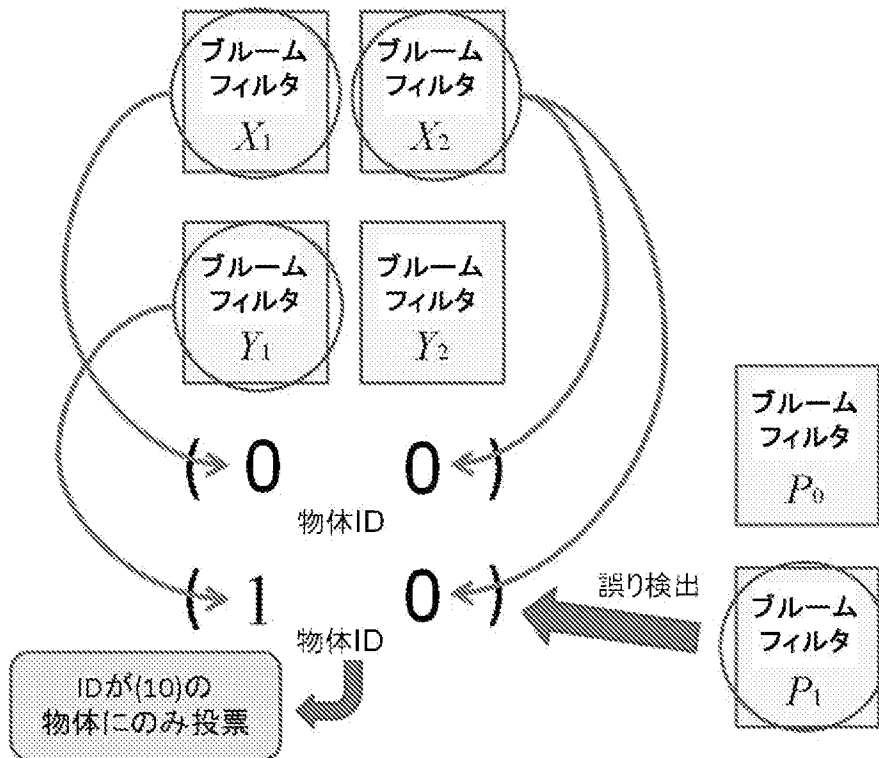
[図22]



[図23]



[図24]



## INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2010/059352

## A. CLASSIFICATION OF SUBJECT MATTER

G06T7/00(2006.01)i, G06F17/30(2006.01)i, G06T1/00(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06T7/00, G06F17/30, G06T1/00

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

|                           |           |                            |           |
|---------------------------|-----------|----------------------------|-----------|
| Jitsuyo Shinan Koho       | 1922-1996 | Jitsuyo Shinan Toroku Koho | 1996-2010 |
| Kokai Jitsuyo Shinan Koho | 1971-2010 | Toroku Jitsuyo Shinan Koho | 1994-2010 |

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

| Category* | Citation of document, with indication, where appropriate, of the relevant passages  | Relevant to claim No. |
|-----------|---|-----------------------|
| A         | Hiroya MINAMI, "WWW ni Okeru Bunsan Caching Gijutsu", NTT Gijutsu Journal, 01 May 2000 (01.05.2000), vol.12, no.5, pages 59 to 62   | 1-9                   |
| A         | BERNARD et al., The Bloomier filter: an efficient data structure for static support lookup tables, Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms, 2004, p.30-39   | 1-9                   |
| A         | Kazuto NOGUCHI, et al., "Efficient Recognition of Specific Objects by Cascading Approximate Nearest Neighbor Searchers", Meeting on Image Recognition and Understanding (MIRU2007) Ronbunshu, 30 July 2007 (30.07.2007), pages 111 to 118 | 1-9                   |

 Further documents are listed in the continuation of Box C. See patent family annex.

\* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&amp;" document member of the same patent family

Date of the actual completion of the international search  
28 June, 2010 (28.06.10)Date of mailing of the international search report  
13 July, 2010 (13.07.10)Name and mailing address of the ISA/  
Japanese Patent Office

Authorized officer

Facsimile No.

Telephone No.

A. 発明の属する分野の分類 (国際特許分類 (IPC))  
 Int.Cl. G06T7/00(2006.01)i, G06F17/30(2006.01)i, G06T1/00(2006.01)i

B. 調査を行った分野  
 調査を行った最小限資料 (国際特許分類 (IPC))  
 Int.Cl. G06T7/00, G06F17/30, G06T1/00

最小限資料以外の資料で調査を行った分野に含まれるもの  
 日本国実用新案公報 1922-1996年  
 日本国公開実用新案公報 1971-2010年  
 日本国実用新案登録公報 1996-2010年  
 日本国登録実用新案公報 1994-2010年

国際調査で使用した電子データベース (データベースの名称、調査に使用した用語)

C. 関連すると認められる文献

| 引用文献の<br>カテゴリー* | 引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示   | 関連する<br>請求項の番号 |
|-----------------|---|----------------|
| A               | 南裕也, WWWにおける分散キャッシング技術, NTT技術ジャーナル, 2000.05.01, 第12巻, 第5号, p.59-62  | 1-9            |
| A               | BERNARD et al., The Bloomier filter: an efficient data structure for static support lookup tables, Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms, 2004, p.30-39 | 1-9            |

C欄の続きにも文献が列挙されている。  パテントファミリーに関する別紙を参照。

|   |   |
|---|---|
| <p>* 引用文献のカテゴリー<br/>                 「A」特に関連のある文献ではなく、一般的技術水準を示すもの<br/>                 「E」国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの<br/>                 「L」優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献 (理由を付す)<br/>                 「O」口頭による開示、使用、展示等に言及する文献<br/>                 「P」国際出願日前で、かつ優先権の主張の基礎となる出願</p> | <p>の日の後に公表された文献<br/>                 「T」国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの<br/>                 「X」特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの<br/>                 「Y」特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの<br/>                 「&amp;」同一パテントファミリー文献</p> |
|---|---|

|  |  |
|--|--|
| 国際調査を完了した日<br>28.06.2010   | 国際調査報告の発送日<br>13.07.2010                               |
| 国際調査機関の名称及びあて先<br>日本国特許庁 (ISA/JP)<br>郵便番号100-8915<br>東京都千代田区霞が関三丁目4番3号 | 特許庁審査官 (権限のある職員)<br>松尾 俊介<br>電話番号 03-3581-1101 内線 3531 |

| C (続き) . 関連すると認められる文献 |   |                |
|-----------------------|---|----------------|
| 引用文献の<br>カテゴリー*       | 引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示   | 関連する<br>請求項の番号 |
| A                     | 野口和人, 外 2 名, 近似最近傍検索の多段階化による物体の高速認識, 画像の認識・理解シンポジウム (MIRU2007) 論文集, 2007.07.30, p.111-118 | 1-9            |