

特定物体認識手法による大量画像を用いた一般物体認識

秋山 瑞樹[†] 柳井 啓司[†]

[†] 電気通信大学大学院 情報理工学研究科 総合情報学専攻 〒182-8585 東京都調布市調布ヶ丘 1-5-1

あらまし 本研究の目的は、同一インスタンス検索のための特定物体認識手法を Web から収集した大量の画像に適用することで、カテゴリー分類である一般物体認識が実現可能かどうか検証することである。本研究では、収集した画像の局所特徴をデータベース化し、未知画像の局所特徴を 4000 万点以上ものデータベース化した局所特徴に対して近似最近傍探索を行うことで、類似特徴をもつカテゴリーに投票を行い、投票数による画像のカテゴリー分類を行った。独自に構築した 5 カテゴリー約 14 万枚のデータセットを用いて、bag-of-features による一般的な手法の分類率 66.9 % に匹敵する、60.1 % の分類率が得られた

キーワード 一般物体認識, 特定物体認識, SIFT, approximate nearest neighbor

Generic Object Recognition by the Methods for Specific Object Recognition Using a Large Number of Images

Mizuki AKIYAMA[†] and Keiji YANAI[†]

[†] Department of Informatics, Graduate School of Informatics and Engineering, The University of Electro-Communications Chōfugaoka 1-5-1, Chōfu-shi, Tokyo, 182-8585 Japan

Abstract Generic object recognition is category-level object recognition, while specific object recognition is instance-level object recognition. Although their objectives are different, instance-level recognition possibly becomes category-level recognition, if we can prepare a large number of instances that belong to certain categories. Then, in this paper, we examine if generic object recognition is possible by using specific object recognition methods with a large number of sample images. In the experiments, we used two kinds of the common methods for specific object recognition. One is matching SIFT features, and the other is matching visual words. For a five-category dataset we built by ourselves which consists one hundred and forty thousand images, we obtained 60.1% for classification rate with SIFT feature matching. This result is almost equivalent to the result by the generic object recognition method which employs standard bag-of-features and SVM.

Key words general object recognition, specific object recognition, SIFT, approximate nearest neighbor

1. はじめに

1.1 背景

実世界シーンの一般的な画像に含まれる物体に対して、椅子、自動車など一般的な物体カテゴリーを言い当てる処理を計算機にさせる研究である「一般物体認識」[1] が近年盛んに行われている。特定の制限下ではない一般的な画像を認識させる場合、「一般物体認識」の実現はまだ難しいのが現状である。

一方で画像に写っている物体とまったく同一のものが写った画像を、膨大なデータベースから発見する処理は、異なる画像でも物体特有の特徴が取れているならば計算機にとって容易である。まったく同じ物体かどうかを言い当てる処理を「特定物体認識」と言われている。

もし一般物体認識として認識したい画像と類似した画

像がデータベースとして登録されているならば、類似画像を高速に探し出し、登録された情報を元に認識が可能になるのではないかと考え、特定物体認識の手法を一般物体認識に適応を試みた。

1.2 目的

本研究では、一般に撮影された画像に対して写っている対象の物体のカテゴリーを、大量の画像データと特定物体認識手法を用いて、認識することがどの程度可能かどうかを検証することを目的とする。

認識手法として、大量の画像を Web 上から収集、それらの画像から特徴を抽出しデータベース化する。実際に認識をしたいクエリ画像に対して最近傍探索の高速マッチングを用いた特定物体認識の手法 [4] から一般物体認識の実現を目標とする。以上のようにデータ量で認識の

問題を解決するアプローチの研究を目指す。Web上の画像データ量は指数関数的に増大しており、膨大なデータに対する単純な最近傍探索で認識問題を解決するアプローチは、画像認識の実用化を目指す上で今後有効な方法の一つとなると考えられる。

2. 関連研究

今回の実験では一般物体認識の研究を行ったが、研究としては特定物体認識の手法を一般物体認識に応用した場合どうなるかという試みである。ここでは特定物体認識の手法と最近の研究についての紹介を行う。

実験手法の参考となった研究として、特定物体認識の手法として黄瀬らの研究 [2] [3] がある。[2] では局所特徴を用いた特定物体認識手法に関する基本的な技術や研究成果など局所特徴に用いた単純な照合、投票でも高い認識精度が得られることが述べられている。[3] では SIFT, PCA-SIFT, SURF など局所特徴の違いをポスターや雑誌の画像を対象として認識精度の実験が行われている。

特定物体認識の先駆けとしては、Sivic らの Video Google [5] がある。指定した物体をビデオ中で探索する研究であり、visual words を用いた高速オブジェクト探索が研究されている。また [5] を改良した、Philbin らの研究 [7] がある。この研究では visual words の作成方法を改良し画像内のランドマークの位置検出を行っている。最近の研究としてランドマークに絞った研究としては Zheng らの研究 [9] がある、2000 万枚のジオタグ画像からランドマーク画像データベースを作成し、類似画像検索を用いて未知のランドマークを特定する研究がある。Gammeter らの研究 [6] では旅行画像などに写るランドマークを特定することで、画像に写っているランドマークに関する位置情報や関係するコンテンツをインターネットから自動で収集しアノテーションを行うといった研究がある。

このようにランドマークやポスターなど決まった形状で常にある程度同様な局所特徴が取れることが期待される特定物体に対する認識は高精度での認識が可能となっている。

一方、大量のデータを用いた一般物体認識の研究としては、A. Torralba らによる 8,000 万枚の Web 画像による類似画像検索を用いた研究がある [12]。A. Torralba らは、8,000 万枚もの大量の Web 画像を収集し、 32×32 の画像に縮小して、単純な k-最近傍分類で画像分類を行った。その結果、Web から収集しただけのノイズが含まれた画像データを学習データとしてもその量が十分に多ければ、単純な手法であっても bag-of-features などの最新の手法に匹敵する一般物体認識が実現できることを示した [12]。本研究でも、同様に Web から大量の画像を収集するためにノイズ除去を行なうことが困難であるので、ノイズが含まれているデータをそのまま利用する。しかしながら [12] が画像全体からの特徴量を用いているの

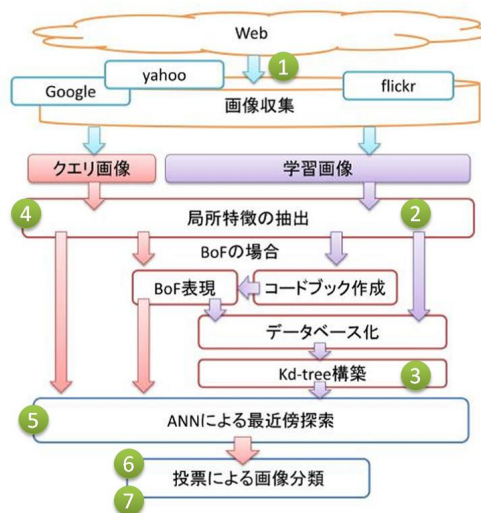


図 1 フローチャート

に対して、本研究では特徴点の対応をカウントする特定物体認識手法を利用する点が異なっている。

3. 提案手法概要

本研究では、大量の画像の特徴点をデータベースに登録し、最近傍探索によって類似特徴点探索を行い、類似特徴点のカテゴリの投票によって、特定物体認識手法による一般物体認識の実験を行う。局所特徴量の表現としては、SIFT, PCA-SIFT, SIFT 特徴の bag-of-features (BoF) 表現の 3 種類を用いる。

本研究での大まかな流れとしては以下ようになる。図 1 はシステムのフローチャートである。フローチャート中の番号はシステムの流れの番号と対応する。

- (1) Web 上からデータベース用の学習画像収集をする。
- (2) すべての画像から局所特徴を抽出しテキストファイルに書き出しデータベース化する。またそれらの特徴がどの画像から抽出された特徴であるかについてもデータベース化する。BoF による実験の場合はすべての特徴からコードブック作成後、学習画像の BoF ベクトルをデータベース化する。
- (3) データベースを読み込み、最近傍探索のためのデータ構造として kd-tree を構築する。
- (4) 認識対象の未知画像から特徴を抽出する。
- (5) 得られた未知画像の特徴 1 つ 1 つに対して近似最近傍探索、ANN (Approximate Nearest Neighbor) によって kd-tree から最近傍特徴を高速マッチングする。
- (6) 選ばれた特徴を持っているデータベースの画像に投票を行う。
- (7) 未知画像のすべての特徴に対してマッチングを行い、最終的に投票が多かったデータベースの画像のクラスを未知画像の認識結果とする。

4. 提案手法詳細

4.1 画像収集

画像はクエリ語によって Web 画像検索エンジンを利用して収集した。使用した検索エンジンは Google, Yahoo!, Flickr の 3 つを利用した。

Google, Yahoo 検索エンジンではクエリ語に対して表示される検索結果に約 1000 枚までの限界が設けられているので、クエリ語を変えながらランク順に画像を取得し、Flickr では API を使い relevance(関連度)のランク順に画像を取得した。

変化させるクエリ後は収集したい画像を下位クラスとしたとき、「下位クラス名」と「上位クラス名 + 下位クラス名」をクエリ語として使用した。さらにそれぞれについて日本語、英語について検索を行い計 4 種類のクエリ語で検索を行った。例えばバラの画像を集める場合、「バラ」、「花 バラ」、「rose」、「flower rose」といった具合で画像検索を行う。

一つの下位クラスの画像収集に対して、3 エンジン*4 種類の計 12 種類での画像検索を行った。

重複が無いように 12 種類の検索結果を各種類上位から順に画像として保存していくことで、検索エンジンのランキングに基づいて画像を取得した。また、今回の実験では手動で 25 種類の下位クラスを決め実験を行った。

4.2 局所特徴

局所特徴とは、特徴点オペレータにより画像中の濃淡変化が大きい特徴点を検出し、その特徴点周りの領域を画素値や微分値等により特徴ベクトルにしたものである。この特徴は同一対象であれば、視点変化や回転、スケールの異なる画像であっても、同じ特徴点が検出されやすい。本実験で用いる局所特徴として、SIFT と PCA-SIFT を使用し実験を行った。さらに画像の表現方法として局所特徴の頻出頻度表現で表現する BoF (Bag-of-Features) 表現を用いた場合の計 3 方法で実験を行った。

また巨大な画像の場合画像の長辺が 640pixel になるように比率を保ち縮小させてから、局所特徴の検出を行った。

SIFT 特徴

SIFT (Scale Invariant Feature Transform) [8] は D.Lowe によって考案され、特徴点周りの局所画像パターンを 128 次元特徴ベクトルで表現し、回転・スケール変化・照明変化に対して耐性のある特徴である。特徴点検出には Difference of Gaussian (DoG) を使用している。

PCA-SIFT 特徴

PCA-SIFT [10] は Ke らによって考案され SIFT の拡張手法である。

SIFT で検出された特徴点周辺の 41x41 の正方形領域に対して勾配情報を求め、3042 次元の特徴ベクトルを得

る。その特徴ベクトルに対して PCA により得られた射影行列を用いて部分空間に投影し主成分分析を行うことで 36 次元へと圧縮した特徴ベクトルである。

次元数の減少によりマッチング処理、メモリー効率の点で利点がある。

Bag-of-Features

Bag-of-Features とは画像を局所特徴の集合と捉えた画像の表現方法である。Bag-of-Features の基本的な考え方は bag-of-words というテキスト検索のモデルであり、bag-of-words は文章中に出現する単語のコードブックをもとに、語順に関係なく文章を単語の出現頻度で表現する方法である。Bag-of-Features では単語の代わりにベクトル量子化された局所特徴を用いることで画像を局所特徴の集合として考える。

Bag-of-Features の作成手順としては以下のような流れとなる。

まず学習画像から局所特徴を抽出する。次に、すべての局所特徴を k 個にクラスタリングすることで visual words と呼ばれる似たような特徴が集まった特徴ベクトルが k 個作成し、それらの特徴ベクトルをまとめたものをコードブックと呼ぶ。画像を対象とし、画像一枚づつに対して得られた局所特徴をコードブックの k 個の特徴ベクトルのうち最も近い特徴ベクトルに投票を行うことで、出現回数のヒストグラムで画像を表現する。その画像の局所特徴の総数でヒストグラムの各 bin を割ることによって作成された、正規化されたヒストグラムがその画像に対しての Bag-of-Features 表現となる。

4.3 データベース

SIFT, PCA-SIFT による実験の場合、特徴ベクトルが 1 行 1 特徴として書かれた特徴データベースと、特徴データベースに書かれた特徴 ID と対応する画像名がされている画像名データベースの 2 種類のデータベースを作成した。

Bag-of-Features の実験の場合、まず作成したコードブックを元に、各行が画像 1 枚を表す特徴データベースを作成した。こうして作成した特徴データベースは類似コードブック特徴を持つ画像に対して投票が困難である。そこでコードブック特徴ベクトルについての各学習画像の評価値を読み込めるような、転置ファイルも作成した。転置ファイルはコードブックサイズの行を持ち、対応する行のコードブック特徴ベクトルをもつ学習画像名を列として持つ。

SIFT, PCA-SIFT 実験では、特徴データベースから木構造を作成し、近似特徴探索で得られた特徴 ID をもとに、学習画像を画像名データベースから探索することで投票を行う。BoF 実験では、コードブックから木構造を作成し、転置ファイルをから学習画像に対して投票を行う。

4.4 特徴探索

クエリ画像の1つ1つの特徴に対して、データベース化された学習画像の特徴から最近傍特徴を探す際に上から順に探索していくという単純な最近傍探索では、すべての特徴に対して類似計算を行わなくてはならず、今回の実験のようなデータベースの特徴数が1000万個を超えるような実験では計算コストが肥大し実用可能な範囲での物体認識は困難である。

そこで ANN (Approximate Nearest Neighbor) [11] という近似最近傍探索を導入することで最近傍探索の時間を高速化させた。

Approximate Nearest Neighbor

ANN (Approximate Nearest Neighbor) は木探索を用いた近似最近傍探索の手法である。今回の実験ではデータ構造として kd-tree という木構造を使用し、最近傍探索を行った。ANN の流れは以下ようになる。

まずデータベースから kd-tree を構築する。再帰的な処理により特徴空間で分割していくことで、セルと呼ばれる同じ分割ルールによって分けられた特徴が集まる領域に分割していく。こうしてできたセルを葉ノード、分割ルールを内部ノードとすることで木構造 (kd-tree) を構築する。

次に作成した kd-tree からクエリ画像の特徴に対して最近傍探索を行う。クエリ画像の特徴ベクトル q がどのセルの内部にあるのかを、木構造探索によって求める。発見したセルに対応づけられている特徴ベクトルを p とするとき、真の最近傍は $r(p, q)$ を半径とする円の中にある。そしてその円と重なりを持つほかのセルを訪問し、そのセルに含まれている特徴ベクトルとの距離を計算し、最小のユークリッド距離を与える特徴ベクトルを最近傍特徴とする。

さらに探索の時間を短くするために、 r の距離をそのまま用いるのではなく $1/(1 + \epsilon)$ を乗じ半径を小さくすることで計算対象の特徴ベクトルが減り、高速化が可能になる。

4.5 認識

クエリ画像から得たすべての特徴それぞれに対してデータベース内の近似最近傍特徴を持つ画像に投票を行い、すべての投票が終わったら投票数順に画像名をソートすることで投票数ランキングが得られる。こうして得られたランキングから kNN (k-Nearest Neighbor) によってクエリ画像のクラスを認識する。

k-Nearest Neighbor

k-Nearest Neighbor とは k 個の最近傍のオブジェクトの中で最も一般的なクラスに分類する方法である。上位 k 位までについてクラスの多数決を行い、もっとも票が多くなったクラスをそのクエリ画像のクラスと認識する。

表1 クエリ語

動物	ネコ	イヌ	ゾウ	ライオン	トラ
車	インプレッサ	レクサス	オデッセイ	パジェロ	プリウス
花	コスモス	タンポポ	ラベンダー	ユリ	バラ
食べ物	ケーキ	ハンバーガー	ピザ	ラーメン	スシ
楽器	ドラム	フルート	ギター	ピアノ	バイオリン



図2 画像例

表2 データベース内訳

	クラス辺りの画像数	総画像数	クラス辺りの特徴数	総特徴数
SIFT	1,050	26,250	600,000	15,000,000
PCA-SIFT	2,900	72,500	2,140,000	53,500,000
BoF	5,800	145,000	-	-

5. 実験

5.1 画像収集

実験には上位クラス 5 種類に属する計 25 のクラスで画像収集を行った。25 種類のクエリ語は表 1 のとおりである。左が上位クラス名、右が実際に画像収集する下位クラスである。また収集した画像例を図 2 で表す。

5.2 実験手順

まず収集した画像を学習画像とクエリ画像に分割する。クエリ画像は収集した画像に対して検索エンジンの結果として下位となった正しい画像を各クラス 50 枚ずつ選出し、残りの画像を学習画像とした。

実験は SIFT, PCA-SIFT, BoF (Bag-of-Features) の 3 種類の手法に対して行った。

SIFT, PCA-SIFT の実験では、クラスによってクラス内総特徴数に大きなばらつきがあると総特徴数が多いクラスに対して有利に投票が行われてしまうので、それぞれの実験において最小の特徴数となったクラスと同数になるまで他のクラスについてもランダムに特徴数の削減を行った。

それぞれの実験において使用した学習画像数と、特徴点数は以下の表 2 となる。

こうして選出した特徴量をデータベースとして登録する。BoF での実験では選出した SIFT の特徴を用いてコードブックを作成し、学習画像の BoF 表現を作成しデータベースとした。またコードブックサイズを変化さ

せた場合の実験を行った。

作成した特徴データベースに対して ANN を用いて、クエリ画像の特徴点に対し最近傍特徴を持つ画像に投票を行った。実験では、ANN で得られた近傍を上位 n 位 ($n=1,5,10,25$) までを近傍として許容した場合において得られた投票数ランキングに対して、 k NN($k=1 \sim 10,000$)での認識を行った。また BoF での実験では使用した画像が多いので、 $k=100,000$ まで認識を行った。

5.3 評価方法

認識精度の尺度として適合率、再現率、分類率によって評価を行った。それぞれの式は以下のように定義される。今回の実験では主に分類率を評価指標として利用した。

$$\begin{aligned} \text{適合率} &= \frac{\text{正しく識別された画像数}}{\text{正しいと識別された画像数}} \\ \text{再現率} &= \frac{\text{正しく識別された画像数}}{\text{正しいクエリ画像総数}} \\ \text{分類率} &= \frac{\text{正しく識別された画像総数}}{\text{全クエリ画像数}} \end{aligned}$$

6. 実験結果

近傍数 n と k NN 投票数 k の変化における認識精度を上位クラス分類と下位クラス分類の場合について実験を行った。CPU は Intel(R)Xeon(R)CPU 5140 2.33GHz、メモリ 32GB の計算機を使用した。特徴から最初に kd-tree を作成するには、SIFT の場合約 1 時間、PCA-SIFT の場合約 2 時間ほど時間がかかったが、一度作成された kd-tree を書き出すことで、次の読込の際は 10 分ほどで行える。

またメモリ使用率は SIFT 実験の場合約 20GB、PCA-SIFT 実験の場合約 26GB、BoF 実験の場合約 5GB のメモリを必要とした。データ量は多いが探索は高速に行うことができる。クエリ画像から得られた特徴点数にも依存するが SIFT では約 2 秒、PCA-SIFT では 3 秒、BoF では 2 秒ほどで投票数ランキングを得られる。実行時間の多くの時間が特徴の抽出とソートによる実行時間である。 k NN は k 位までの投票を行うだけなので、時間はほぼかからない。

また各実験について未知画像のクラス分類が目的なので、認識分類率準として考え、下位クラスにおける分類率がもっとも高くなった n, k の組み合わせについて適合率 (%) と再現率 (%) を表として示した。

まずベースラインとして BoF(コードブックサイズ 1000)+サポートベクターマシンを使用した実験結果と今回の提案手法との比較結果は図 3 のようになった。ベースラインには少し及ばなかったが、トラやピアノなど特定のクラス分類においてはベースラインよりも高い精度を得ることができた(図 4)。

また提案手法のうち SIFT, PCA-SIFT を用いた場合の、ANN での近傍数 n と k NN 投票数 k の変化における精度

	上位クラス (5クラス) 分類率(%)	下位クラス (25クラス) 分類率(%)
SIFT($n=5, k=7000$) 提案手法	60.1	32.5
PCA($n=5, k=7000$) 提案手法	57.2	29.8
BoF(size=50000) 提案手法	43.9	25.4
BoF+SVM(線形カーネル) ベースライン	51.7	17.1
BoF+SVM(χ^2 カーネル) ベースライン	66.9	36.2

図 3 比較結果

	SIFT 再現率	BoF+SVM 再現率	SIFT 適合率	BoF+SVM 適合率
ピザ	40	64	33	48
ラーメン	18	60	33	35
ドラム	0	22	0	42
トラ	78	70	63	63
ピアノ	70	46	60	46
ギター	58	34	76	35

図 4 比較結果 2

の変移は図 5,6 のようになった。SIFT, PCA-SIFT 手法の場合、 k の値は 7000 ほどまで精度が上がったが、それ以降は横ばいか精度が下がる傾向がわかった。 n の値はどちらの場合でも $n=5$ の時もっとも良い分類率を得ることができた。

BoF のコードブックサイズによる精度の変移は図 7,8 のようになった。コードブックサイズは 20 万の精度が良く、 k の値は 20000 ほどまで精度が上がったが、それ以降は精度が下がる傾向がわかった。

6.1 SIFT 実験

分類率がもっとも良かった $n=5, k=7000$ について適合率と再現率を示す。上位クラス分類 (5 クラス分類) は表 3、下位クラス分類 (25 クラス分類) は表 4 となった。

6.2 PCA-SIFT 実験

分類率がもっとも良かった $n=5, k=7000$ について適合率と認分類率を示す。上位クラス分類 (5 クラス分類) は表 5、下位クラス分類 (25 クラス分類) は表 6 となった。

6.3 BoF 実験

分類率がもっとも良かったコードブックサイズ 20 万、 $k=20000$ について適合率と認分類率を示す。上位クラス分類 (5 クラス分類) は表 7、下位クラス分類 (25 クラス分類) は表 8 となった。

7. 考察

7.1 手法の違いに関して

SIFT, PCA-SIFT の結果の違いに関しては使用した画像データ数が異なるので結果から単純に PCA-SIFT の方が悪いとはいえないが、すべての n, k の値で SIFT の

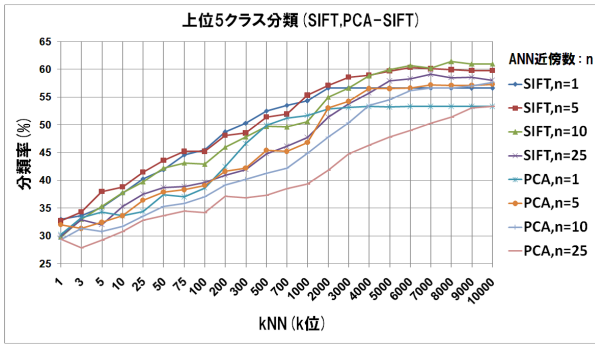


図 5 SIFT,PCA-SIFT 実験 (上位 5 クラス分類)

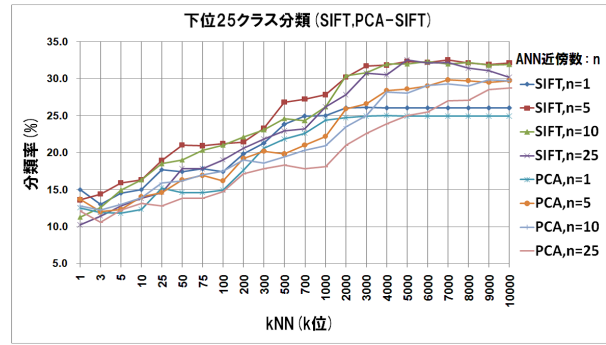


図 6 SIFT,PCA-SIFT 実験 (下位 25 クラス分類)

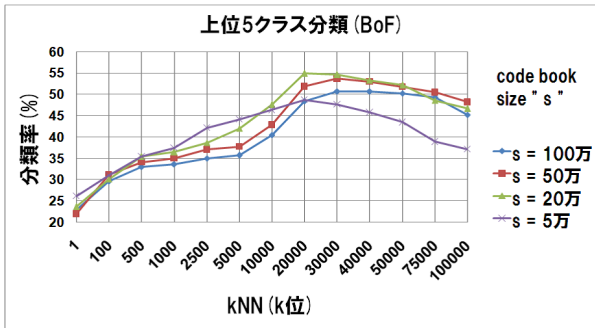


図 7 BoF 比較結果 (上位 5 クラス分類)

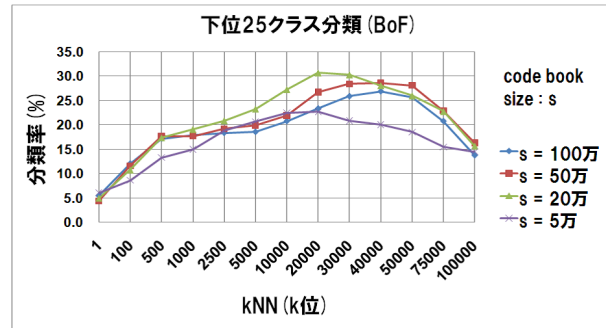


図 8 BoF 比較結果 (下位 25 クラス分類)

表 3 SIFT, 上位クラス分類 (適合率, 再現率)

	1	2	3	4	5	再現率 (%)
1: 動物	155	14	37	37	7	62
2: 車	10	228	3	4	5	91
3: 花	35	15	150	41	9	60
4: 食べ物	43	24	40	135	8	54
5: 楽器	12	128	10	14	86	34
適合率 (%)	61	56	63	58	75	60.3

表 5 PCA-SIFT, 上位クラス (適合率, 再現率)

	1	2	3	4	5	再現率 (%)
1: 動物	166	15	46	17	6	66
2: 車	14	216	6	4	10	86
3: 花	44	20	163	20	3	65
4: 食べ物	73	28	64	79	6	32
5: 楽器	27	107	19	6	91	36
適合率 (%)	51	56	55	63	78	57.2

表 4 SIFT, 下位クラス分類 (適合率, 再現率)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	再現率 (%)	
1: 動物	4	5	1	3	2	0	2	1	0	1	0	1	1	3	2	1	2	3	1	0	0	0	0	0	0	0	14
2: 車	6	2	0	3	2	1	0	0	1	5	0	3	7	4	3	4	3	0	0	2	0	1	1	0	2	4	
3: ソウ	2	3	9	7	3	2	0	1	0	0	4	2	13	0	0	0	1	1	0	0	0	1	1	0	0	18	
4: ライオン	2	1	1	2	6	0	0	0	0	2	3	8	0	1	1	1	3	0	0	0	0	0	0	0	0	42	
5: トラ	3	0	0	4	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	78	
6: インフレス	0	1	0	2	0	2	1	3	6	6	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	22	
7: アレックス	0	0	0	0	0	13	12	7	6	7	0	0	0	0	1	0	1	1	1	1	0	0	0	0	0	24	
8: オプンセイ	0	0	0	0	1	10	8	7	12	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0	14	
9: パジエロ	0	0	0	0	0	7	5	7	16	14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	32	
10: アリウス	1	1	2	0	0	5	6	9	5	15	1	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	1
11: コスモス	0	0	1	1	0	1	0	0	0	0	2	3	8	0	1	1	1	3	0	0	0	0	0	0	0	1	1
12: サンダー	1	0	0	4	0	0	0	1	0	0	9	22	0	0	0	0	1	0	0	0	0	0	0	0	0	0	44
13: ラベンダー	0	0	2	4	2	0	0	0	0	0	6	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	80
14: スリ	4	1	0	1	0	2	0	1	0	0	6	2	2	12	7	3	0	4	1	2	0	0	0	0	0	2	24
15: パウ	0	1	0	1	2	0	0	1	0	4	0	3	4	2	3	2	1	0	1	0	1	0	0	0	0	1	50
16: ケーキ	2	1	2	2	0	0	3	3	2	1	5	0	4	1	2	6	4	1	0	0	0	0	0	0	0	3	12
17: ハンバーガー	2	4	1	4	1	0	0	0	3	0	3	0	1	2	4	9	4	3	2	1	6	0	1	0	0	1	8
18: ビザ	3	3	0	6	0	0	0	0	0	3	0	4	0	3	3	2	2	2	0	2	0	0	0	0	0	0	40
19: ケン	1	0	3	1	1	0	2	2	0	1	1	2	2	3	8	3	0	5	2	0	0	0	0	0	0	1	10
20: スシ	0	4	0	1	1	1	1	1	0	1	3	0	1	3	4	13	3	2	1	6	0	1	0	0	0	3	12
21: ドラム	3	1	1	1	0	3	10	3	2	3	2	1	1	0	3	2	4	3	2	0	0	0	0	2	1	2	0
22: フルト	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23: キター	1	1	0	0	0	3	0	2	2	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
24: ヒアノ	0	0	0	0	0	4	3	3	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	35
25: バイオリン	0	0	1	1	0	5	4	5	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	52
適合率 (%)	18	7	32	30	63	25	17	11	27	19	35	59	32	33	40	10	11	33	33	32	0	38	76	60	55	32.5	

表 6 PCA-SIFT, 下位クラス (適合率, 再現率)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	再現率 (%)	
1: 動物	10	3	4	6	4	1	3	1	1	0	0	1	5	0	2	4	0	3	1	0	0	0	0	0	0	0	20
2: 車	7	2	2	6	1	1	1	3	2	1	4	0	6	1	1	6	3	1	0	1	0	0	0	0	0	0	4
3: ソウ	2	0	12	4	5	0	1	0	2	0	4	1	16	0	1	1	0	0	0	0	0	0	0	0	0	2	24
4: ライオン	4	0	3	5	13	0	0	0	0	0	0	3	2	15	0	1	0	0	0	0	0	0	0	0	0	0	18
5: トラ	0	1	0	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	70
6: インフレス	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	26
7: アレックス	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	14
8: オプンセイ	0	0	4	1	0	11	6	7	5	5	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	18
9: パジエロ	2	0	0	0	0	3	5	7	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
10: アリウス	0	0	2	0	2	10	5	1	2	16	1	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	32
11: コスモス	2	3	1	2	2	1	2	0	0	0	22	2	8	2	2	0	0	0	0	0	0	0	0	0	0	0	44
12: サンダー	0	0	3	3	3	0	0	0	0	0	8	20	10	1	1	0	0	0	0	0	0	0	0	0	0	0	40
13: ラベンダー	0	0	2	1	4	1	0	0	0	0	4	1	8	0	0	0	0	0	0	0	0	0	0	0	0	0	2
14: スリ	1	1	2	3	3	0	4	1	3	0	4	4	5	12	3	3	0	1	0	0	0	0	0	0	0	0	24
15: パウ	2	1	1	3	0	1	0	0	2	1	8	1	5	7	10	1	1	0	2	3	0	0	0	0	0	0	20
16: ケーキ	2	0	3	4	2	1	0	0	2	0	2	2	4	3	2	7	2	3	5	2	0	0	0	0	0	0	1
17: ハンバーガー	4	2	5	2	4	1	0	0	2	0	4	2	5	0	1	4	6	2	4	1	0	0	0	0	0	0	12
18: ビザ	1	0	2	5	4	0	0	0	0	0	5	2	14	0	1	3	1	1	1	0	0	0	0	0	0	0	22
19: ケン	2	0	5	2	4	1	1	0	1	7	3	5	1	1	3	0	2	9	1	0	0	0	0	0	0	0	10
20: スシ	1	1	4	5	1	3	2	1	3	0	2	1	4	2	1	3	1	1	1	1	1	1	1	1	0	0	22
21: ドラム	0	1	1	2	2	7	4	5	3	2	1	1	1	0	1	0	1	0	2	0	0	0	0	0	0	0	3
22: フルト	2	0	2	0	0	6	1	3	2	1	0	2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	54
23: キター	0	0	0																								

表 8 BoF, 下位クラス (適合率, 再現率)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	再現率 (%)	
動物	21	1	0	1	3	0	1	0	0	0	3	4	6	0	3	3	0	0	0	0	2	0	1	1	1	41.3	
2: イヌ	16	2	1	0	0	1	0	0	0	0	2	2	9	0	2	2	1	0	0	0	0	1	0	2	9	4	
3: ソウ	9	0	0	2	6	0	2	0	0	0	10	1	10	0	2	0	1	0	0	0	0	0	0	0	1	6	0
4: ライオン	5	0	0	12	4	0	0	0	0	0	5	7	13	0	0	0	3	0	0	0	0	0	0	0	1	24	
5: トラ	3	0	0	5	35	0	0	0	0	0	1	3	0	0	0	0	0	0	0	0	0	0	0	1	1	69.7	
6: インフレッサ	0	0	0	0	1	7	21	0	0	3	2	0	4	0	0	0	0	0	0	0	0	0	0	0	4	8	
7: レクサス	2	1	0	0	0	1	26	1	0	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	32	
8: オデッセイ	1	0	0	0	0	0	17	0	0	2	2	1	0	0	0	0	0	0	0	0	0	0	0	2	5	20	
9: パジコ	0	0	0	0	0	3	16	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	2	0	9	15	
10: プリウス	0	0	0	1	2	0	18	0	0	9	2	1	2	0	0	0	0	0	0	0	0	0	2	0	2	11	
11: コスモス	0	0	0	0	0	0	0	0	0	0	38	1	3	2	1	0	0	0	0	0	0	1	0	0	2	75.7	
12: タンポポ	0	0	0	0	2	0	0	0	0	0	15	23	6	0	0	0	0	0	0	0	0	0	1	0	0	45.3	
13: ラベンダー	0	0	0	1	1	1	0	0	0	0	10	3	31	0	0	1	0	1	0	0	0	0	0	0	0	61.4	
14: ユリ	2	0	0	0	0	1	2	0	0	10	4	3	12	3	4	0	0	0	0	0	0	0	0	0	2	23.7	
15: パラ	0	0	0	0	0	3	3	0	0	8	0	2	1	34	0	0	1	0	0	0	0	0	0	0	0	67.7	
16: ケーキ	6	1	0	1	0	0	3	0	0	7	0	5	6	7	1	0	0	0	0	0	1	0	0	5	7	13.8	
17: ハンバーガー	9	1	0	2	0	1	1	0	0	6	0	3	1	8	7	5	1	0	0	0	0	0	0	1	4	10	
18: ピザ	1	0	0	1	0	0	0	0	0	13	3	7	0	0	0	2	0	21	0	0	0	0	0	0	2	41.9	
19: ラーメン	4	0	0	1	0	2	9	0	0	1	8	5	1	0	2	6	0	1	2	0	0	0	0	0	1	7	
20: スシ	7	1	0	0	1	0	0	0	0	6	1	1	0	8	8	3	1	0	3	0	0	0	2	8	6		
21: ドラム	0	0	0	0	1	3	14	0	0	2	10	1	1	0	4	0	0	0	0	0	2	3	3	6	0		
22: フルート	2	0	0	0	1	11	13	0	0	2	0	0	0	0	0	0	0	0	0	0	3	0	0	3	11	6	
23: ギター	0	0	0	0	0	0	7	3	0	2	0	0	0	0	0	0	0	0	0	2	20	7	9	40			
24: ピアノ	0	0	0	0	0	0	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	38	5		
25: バイオリン	0	0	0	1	0	0	7	0	0	0	0	0	0	0	0	0	0	0	0	0	1	4	37	72.9			
適合率 (%)	23.9	28.6	0	42.9	61.4	22.6	15.6	0	0	32.1	23.6	39.7	28.2	75	45.9	16.7	45.5	70	100	100	0	15	83.3	40.4	19.1	30.7	

表 7 BoF, 上位クラス (適合率, 再現率)

	1	2	3	4	5	再現率 (%)
1: 動物	109	23	71	17	30	43.6
2: 車	4	203	13	0	30	81.2
3: 花	7	19	197	3	24	78.8
4: 食べ物	16	32	89	73	40	29.2
5: 楽器	5	121	16	4	104	41.6
適合率 (%)	77	51	51	75	46	54.9

る場合、認識精度が向上していった。n の範囲を広げる場合、その画像を表すのに有効な特徴も有効で無いノイズ特徴というのにも同等に n 個まで許容してしまうので、画像に対してノイズ特徴が有効な特徴よりも多く取れる時には、ノイズ特徴の総投票数が有効特徴の総投票数を大きく上回ることによって、認識精度が下がったのではないかと考えられる。一方 k の範囲を 7000 まであげると認識精度が上がったことに関しては、有効な特徴ではある程度正しく正解クラスの画像に投票が行われ、ノイズ特徴は他のクラス画像に散らばって投票が行われるためだと考える。有効特徴がノイズ特徴よりも少なかったとしても、ノイズ特徴は様々な画像に投票されるが、有効特徴は着実に正解画像に投票がされていくために、k=7000 の範囲まで許容した場合、投票数こそ余り多くないが確実に有効特徴の投票がされていた正解画像というのが多く拾えるようになったことで認識精度が上がったと考えられる。しかし k の範囲を広げるには総投票数自体も多くなければ投票された画像数を k が超えてしまうので、いかに有効特徴を大量に取れるかというのが重要になってくると思う。また k=7000 以上の値では投票数が足りなくなってしまうたり、投票される側の画像数が k に対して少なくなってしまうことから精度が伸び悩んだと考えられる。

7.3 認識精度に関して

全体として結果が最も良かった SIFT の n=5, k=7000 に関する結果を見て考察を行う。また後述するピアノ、ゾウのクエリ画像に対して投票数の図は左上から右に順位順に並べており、一番結果が良かった SIFT 実験での、n=5, k=7000 の結果である。

下位クラス認識では、ピアノの鍵盤、ギターの弦といった特徴的な特徴を持っている物体に対してはある程度認識ができていた。車クラスに関して車種名までの特定は難しいが、車クラス自体認識という点では認識がある程度できていた。こういった特徴は異なるピアノであっても鍵盤は必ず持っているし、花や食べ物画像にピアノやギターといったものが写っているのは少数であると考えられるため、ノイズ特徴にあまり影響を受けずに認識が

できたと考えられる。正解したピアノ画像については図 9 で表す。不正な画像も集まっているがすべての k に関して正解クラスが選択された。

一方で動物、食べ物のクラス分類というのは難しく、花クラス内だけ見てもコスモスに引っ張られてしまっていた。動物クラスの認識では多くが花クラスに引っ張られてしまっている。動物画像の多くが背景に草原や木といった情報を含んでいるために花クラスの草に引っ張られている。たしかに草原や木といった特徴は「草木」の特徴に投票はされているということが考えられるが、もし動物クラスだけでの実験の場合そういった「草木」特徴というのが動物クラスの中で散らばるので、もしかしたらあまり結果に関わってこない特徴となりうると考えられるが、今回の実験では花クラスが存在するので結果に大きく関わってくるノイズ特徴となってしまったと考えられる。ゾウ画像で花画像に多数投票されてしまった例を図 10 で示す。

そういった「草木」に関するような特徴は食べ物クラスの認識でも関わってきていると考えられ、例えばピザが多くコスモスに出てしまうのは、ピザにのっている野菜などの特徴が取れてしまったように思う。こうした多数のクラスにまたがって出てきてしまう特徴の投票を極力減らす努力をする必要がある。

7.4 必要メモリ量に関して

データベースを増やし、特徴点数を増やしていくと必要なメモリ量というの増大してってしまう。今回の実験では SIFT 実験の場合約 20GB, PCA-SIFT 実験の場合約 26GB のメモリを必要としたが、特徴点数が増えるにつれて、kd-tree を作成する枝情報といったものものも比例して重くなってしまふ。PCA-SIFT と SIFT の比較から、次元を削減するというのはあまり動かないが、例えば SIFT の各次元に関して bit 数の削減というのは試してみる価値があるように思える。

8. ま と め

本研究では、特徴点マッチングに基づく特定物体認識手法を Web から収集した大量の画像に適用すること



図 9 ピアノ分類成功例



図 10 ゾウ分類失敗例

で、カテゴリー分類である一般物体認識が実現可能かどうか検証した。独自に Web から収集した 5 カテゴリー約 14 万枚の画像からなるデータベースの特徴点ベクトルに対して、未知画像の特徴点ベクトルの近似最近傍探索 (Approximate Nearest Neighbor) による投票を行い、投票数の結果から k-最近傍分類 (k-Nearest Neighbor) による多数決による未知画像のクラス分類を行った。

実験の結果、最高で上位クラスに対する分類率 60.3%、下位クラスに対する分類率 32.5% を得ることができた。下位クラス認識では、トラ 78%、ピアノ 70%、ギター 58% などの再現率が得られ、トラの縞模様、ピアノの鍵盤やギターの弦といった特定の特徴を必ず持ったような下位クラスに対しては認識ができた。また上位クラス認識に関しては、車クラスに関して、適合率については 56% で良いとはいえないが、再現については 91% と比較的良い結果が得られた。しかし、動物クラスに関しては背景特徴である「草木」といった特徴に関して花クラスの特徴に多く引っ張られてしまうなど、複数のクラスに共通して出てくるような特徴についての扱いが難しい問題となっていることがわかった。

9. 今後の課題

今後の課題としては、まずは、さらにデータ量を増やした場合の実験が挙げられる。現時点では計算機のメモリの都合で単体の計算機ではこれ以上枚数を増やすことは難しいが、複数の計算機を並列に用いることによって、大規模化は可能であり、今後は多数の計算機を用いて大規模な実験を行う予定である。

また、登録する特徴点の絞り込むことも利用する画像枚数を増やすためには有効である。クラスに関して有益な特徴、例えばトラを認識したい場合背景画像の草原といったものではなくトラ自体の特徴だけを投票できるようにすることが考えられる。他クラスに共通して出てしまう類似特徴を減らすために、作成したデータベースの特徴に関して同じデータベースをクエリとしてすべての特徴に対して探索を行い、ANN の近傍探索範囲 $n = 1$ 以外の投票結果を調べることで投票数の多かった

他クラスからも探索されてしまったと思われる特徴が抽出される。未知のクエリ画像を認識する際に、その特徴に投票された特徴を無視することで、有効な特徴に関する投票ができる可能性がある。

文 献

- [1] 柳井啓司, “一般物体認識の現状と今後,” 情報処理学会論文誌: コンピュータビジョン・イメージメディア, no.48, pp. 1-24, 1997
- [2] 本道貴行, 黄瀬浩一, “大規模画像認識のための局所特徴量の性能比較,” 画像と認識・理解シンポジウム (MIRU2008) 論文集, pp.580-587, 2008
- [3] 黄瀬浩一, “特定物体認識,” 電子情報通信学会パターン認識・メディア理解研究会 (PRMU), pp.79-87, 2009
- [4] 黄瀬浩一, 岩村雅一, “3 日で作る高速特定物体認識システム,” 情報処理, Vol.49, No.9 pp.1082-1089, 2008
- [5] J. Sivic, and A. Zisserman, “Video Google: A text retrieval approach to object matching in videos,” Proc.of IEEE Inter. Conf. on Computer Vision, pp.1470-1477, 2008
- [6] haoS. Gammeter, L. Bossard, T. Quack, and L. Van-Gool, “I know what you did last summer: object-level auto-annotation of holiday snaps,” Proc.of IEEE Inter. Conf. on Computer Vision, 2009
- [7] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, “Object retrieval with large vocabularies and fast spatial matching,” Proc.of CVPR Computer Vision and Pattern Recognition, Vol.3613, pp.1575-1589, 2007
- [8] D. Lowe, “Distinctive image features from scale-invariant keypoints,” International journal of computer vision, vol.60, no.2, pp.91-110, 2004
- [9] Y.T. Zheng, M. Z, Y. Song, H. Adam, U. Budde-meier, A. Bissacco, F. Brucher, T.S. Chua, and H. Neven, “Tour the World: building a web-scale landmark recognition engine,” Proc. of IEEE Inter. Conf. on Computer Vision, 2009
- [10] Y. Ke, R. Sukthankar, “PCA-SIFT: A more distinctive representation for local image descriptors,” IEEE Computer Vision and Pattern Recognition, Vol.2, 2004
- [11] D. Mount, “ANN: A Library for Approximate Nearest Neighbor Searching”, <http://www.cs.umd.edu/~mount/ANN/>,
- [12] A. Torralba, R. Fergus, and W. T. Freeman, “80 million tiny images: a large dataset for non-parametric object and scene recognition”, IEEE trans. on Pattern Analysis and Machine Intelligence, vol.30, no.11, pp.1958-1970, 2009