付加情報付き局所特徴量を用いた 画像カテゴリ識別の向上*

植木 一也** 白石 洋平** 俵 直弘** 小林 哲則**

Improving Classification Accuracy of Image Categories
Using Local Descriptors with Supplemental Information

Kazuya UEKI, Youhei SHIRAISHI, Naohiro TAWARA and Tetsunori KOBAYASHI

In this paper we address the problem of image classification by embedding the spatial information into the local descriptor. In our method, we directly concatenate (x, y) coordinates of an image into the original feature vector. This simple idea can perform well in the object category classification even though the feature vector size is almost the same as the conventional approach. Results are reported for classification of the Caltech-101 dataset and our methods are found to produce consistently better results compared with traditional Bag-of-Features approaches in all experiments.

Key words: image category classification, generic object recognition, local features, bag-of-features, spacial information

1. は じ め に

インターネット上に存在する大量の画像や映像から意味情報を自動で抽出したり、どんな物体が写っているかを自動で認識したりしたいという動機から、一般物体認識というテーマで数多くの研究がなされてきた。一般物体認識技術は、画像の局所特徴量を記述する SIFT 特徴量^{1)–3)}、SURF 特徴量⁸⁾の開発に加え、画像を局所特徴量の集合とみなし、テキストマイニングの技術を適用した Bag-of-Features でエンコーディングする概念が用いられるようになり、識別精度は飛躍的に向上した⁴⁾。現在では、画像のカテゴリ識別は、

- (1) 画像からの特徴抽出 (SIFT 特徴量等)、
- (2) エンコーディング処理 (Bag-of-Features 等)、
- (3) 識別器による識別 (SVM 等)

というパイプラインで行われるのが一般的になっている。

近年では、エンコーディング方法が着目され、LLC (Locality-constrained Linear Coding)¹¹⁾、Super vector coding¹²⁾、Fisher vector¹³⁾ など、様々な方法が提案され、従来の Bag-of-Features によるエンコーディング方法を大きく上回る性能を実現している。特に Fisher vector によるエンコーディングは、様々なデータセットにおいて、他の方法を上回る性能を示している^{14) 16)}。

しかしながら、前述したパイプラインにおける (1) 画像からの特徴抽出、(2) エンコーディング処理、のようにデータ処理を行う度に元の情報は失われていく。多くの手法では、局所特徴量が画像中のどの位置で抽出されたかという情報を無視しているため、意図的に埋め込まない限り、局所特徴量をエンコーディングした段階で位置情報は失われてしまう。失われた情報は元に戻すことはできないため、画像のカテゴリ識別に有益な情報を保持しておくことが重要である。

この問題を補うために、画像中の位置情報を保持していく方法として、画像を縦、横、何段階かに分割し、それぞれの部分領域毎にヒストグラムを計算するという Spatial Pyramid Matching (SPM)⁵⁾ が提案され、Pascal VOC 等のベンチマークにおいても有効であることが示されている⁷⁾。

ただし、Spatial Pyramid Matching では、最終的に識別を行う際に使用する特徴ベクトルの次元数が大きくなり過ぎて、識別に時間がかかってしまう、分割数をいくつにすればいいかを探索する必要があるという問題点がある。これらの問題点については、2 章で詳しく述べる。

これらの問題を解決するため、本論文では局所特徴量に直接、画像の位置情報を埋め込むことで、特徴量の次元数を抑制しながら、画像中の位置情報を保持する手法を提案する。

以下、2 章では、関連研究の Spatial Pyramid Matching について、3 章と 4 章は提案手法と、本手法に用いたエンコーディング手法について説明する。5 章では、本手法の有効性を確認するための実験について述べ、6 章で、結論と今後の展望について述べる。

2. 関 連 研 究

2.1 Spatial Pyramid Matching^{5) 7)}

SIFT 特徴量等の局所特徴量を、特別な処理をせずに Bag-of-Features や Fisher vector でエンコーディングしてしまうと、画像中のどの位置から局所特徴が抽出されたかという情報が失われてしまう。この問題に対応するため、画像を部分領域に分割し、その部分領域毎にエンコーディングすることにより位置情報を保有しようとする Spatial Pyramid Matching^{5) 7)} が提案された。例えば、Bag-of-Features を用いた Spatial Pyramid Matching では、画像をいくつかの部分領域に分割し、分割された部分領域毎にヒストグラムを計算する。代表的な分割方法として、図 1 に示すように、2x2、1x3、4x4 などが用いられている。その後、各部分領域毎に計算されたヒストグラムのベクトルをつなぎ合わせて特徴ベクトルとする。

画像中の局所特徴量を部分領域毎にエンコーディングするこ

* 原稿受付 平成 26 年 5 月 8 日

掲載決定 平成 26 年 8 月 18 日

** 早稲田大学 理工学術院

(東京都新宿区早稲田町 27 早稲田大学 40 号館 701 号室)

Fig.1 Variations of image partition using spatial pyramid matching. Top: 2x2 grids, center: 1x3 grids, and bottom: 4x4 grids

Fig.2 Image partitioning using spatial pyramid matching. Two similar local descriptors extracted at two points (x_1, y_1) , (x_2, y_2) are encoded in totally different sub-regions by the Bag-of-Features encoding, whereas the proposed method can consider these two feature vectors to be very close to each other

とにより、自動車が写っている画像の下には道路が写っていることが多い、飛行機が写っている画像の上には空が写っていることが多いなど、おおよその位置関係を保持することが可能となる。そのため、画像のカテゴリ認識の問題において、多くの場合は精度向上することが報告されている。

Fig.3 Examples of misclassified images: wild cat (left), wheelchair (right)

2.2 Spatial Pyramid Matching の問題点

Spatial Pyramid Matching は、画像中の位置情報を考慮しているため、画像のカテゴリ識別の精度を向上させる効果があることが実証されているが、大きく分けて3つの問題があると考えている。

まず1つ目の問題は、局所特徴量が部分領域に対してハードに割り当てられてしまうという点である。例えば、図2における部分領域の境界面近くに存在する2点 (x_1, y_1) , (x_2, y_2) において局所特徴量が抽出され、Bag-of-Features でエンコーディングすることを考える。その場合、画像中の座標間には大きな差がないものの、別な部分領域にそれぞれハードに割り当てられ、エンコーディングされてしまう。つまり領域の区切り方次第で、エンコーディングされる区分が大きく変わってしまうという不自然なことが起きてしまう。実際に Spatial Pyramid Matching を用いて正しく識別できていない画像を調べてみると、図3のように、識別対象のオブジェクトが画像の中心ではなく、外側に寄っている画像が少なくないことがわかった。図3の左の図では、対象物 (wild cat) が中心よりも上にあり、2x2 の領域に分割した場合、右下の小領域には殆ど対象物が写っていないことがわかる。また図3の右の図では、対象物 (wheelchair) が、画像の左下に偏っており、右上の小領域には殆ど存在しない。Spatial Pyramid Matching では、領域毎にエンコーディングするため、この位置ずれが大きな影響を及ぼしてしまっことが考えられる。

2つ目の問題は、最適な領域の分割方法を求めるのが困難な点である。分割方法としては、前述した通り、2x2, 1x3, 4x4 など均等に分割する方法が一般的だが、均等に分割するのが最適とは限らず、さらにどの分割数が良いかは実験で試してみないとわからないのが現状である。

3つ目の問題は、領域毎にエンコーディングすることから、識別するための特徴ベクトルの次元が大きくなり過ぎてしまう点である。例えば、1x1, 2x2, 1x3 の3つのレベル、8個の部分領域を利用した場合には、特徴ベクトルのサイズが8倍、1x1, 2x2, 4x4 の3つのレベル、21個の部分領域を利用した場合には、特徴ベクトルのサイズが21倍になってしまう。コードブックのサイズが6,000個の場合を考えると、最終的な特徴ベクトルサイズは126,000次元になってしまい、多くのメモリと計算量が必要になってしまう¹⁴⁾。例えば、携帯端末等で撮影した画像を入力とし、そのカテゴリを認識して回答を返すアプリケーションを想定した場合、リアルタイムでの応答も困難になると考えられる。そのため5章の実験では、メモリ使用量や計算量についても考察することとする。

これらの3つの問題を回避するため、本論文では、次元数を抑えながら、画像中の局所特徴量の位置関係を保持できる方法を提案する。

3. 位置情報付き局所特徴量

画像から抽出された SIFT, SURF 等の局所特徴ベクトル

$$\mathbf{v}^T = (v_1, v_2, \dots, v_D) \in R^D \quad (1)$$

または、特徴ベクトルを主成分分析 (PCA) 等の次元圧縮方法を用いて元の次元よりも小さい次元 $d (< D)$ に圧縮した特徴ベクトル

$$\mathbf{v}^T = (v_1, v_2, \dots, v_d) \in R^d \quad (2)$$

に対し、その特徴量が抽出された画像中の座標情報 (x, y) を直接的に埋め込んだベクトルである位置情報付き局所特徴量

$$\mathbf{v}^T = (v_1, v_2, \dots, v_D, \lambda_x \frac{x}{w}, \lambda_y \frac{y}{h}) \in R^{D+2}, \quad (3)$$

$$\mathbf{v}^T = (v_1, v_2, \dots, v_d, \lambda_x \frac{x}{w}, \lambda_y \frac{y}{h}) \in R^{d+2}, \quad (4)$$

を考える。ここで w, h はそれぞれ画像の横サイズ、縦サイズ、 λ_x, λ_y は特徴ベクトルと座標情報のスケールのバランスを調整するためのパラメータである。画像中の位置情報を埋め込む方法は¹⁰⁾にも示されているが、 λ_x と λ_y を別々に設定している点が異なる。例えば、 $\lambda_x = \lambda_y = a$ (定数) のように同じ値を設定した場合には、Spatial Pyramid Matching における 2x2 (または 4x4) に相当するが、1x3 や 3x1 のように x 方向の位置や y 方向の位置を無視したい場合に対応できない。その際は、 $\lambda_x = 0, \lambda_y = a$ のようにすることで、 x 方向の位置のみ完全に無視することができるため、図 1 の中央の図 (1x3) に相当する処理を行うことができる。また、Spatial Pyramid Matching で一般的に行われる 1x1 と 2x2 の組み合わせや、1x1, 2x2, 4x4 の組み合わせについても、パラメータ λ_x と λ_y の組み合わせを複数実施し、それぞれ個別の識別器を構成して統合することで実現可能である。これに関連する実験は 5.5 節で詳しく示すこととする。 λ_x, λ_y のパラメータ設定方法は直感的には難しいが、元の特徴ベクトルの値の範囲と同様のスケールになるように設定するのが基本設定とし、位置による変動をあまり許容したくない場合には、比較的大きな値に設定し、位置の変動の影響をあまり考慮しない場合には、小さな値に設定すればよい。

エンコーディングする際のベクトルとして式 (1)、式 (2) の代わりに、式 (3)、式 (4) を使うことで、Spatial Pyramid Matching のように巨大なサイズの特徴ベクトルを作成する必要がなく、容易に局所特徴量に位置情報を埋め込むことができる。例えば、一般的に用いられる SIFT 特徴量 (128 次元) に位置情報を組み込んだ場合の次元数は $128+2=130$ 次元、主成分分析を用いて 64 次元に圧縮した場合には 64 次元のベクトルに座標情報の 2 次元をつなげた $64+2=66$ 次元の特徴ベクトルを用いて Bag-of-Features や Fisher Vector 等のエンコーディングを行えばよい。

また、局所特徴量は、各画像から SIFT で一般的に用いられるキーポイント抽出、Harris コーナー検出などで行うことで獲得してもよいし、格子状に一定間隔に配置された局所領域から獲得してもよい。位置情報付き局所特徴量についても、従来手法と同様に、1 つの画像につき数千個から数万個抽出すればよい。

4. エンコーディング手法

画像から得られた複数の位置情報付き局所特徴量を用い、エンコードを行う。提案手法の有効性を確認するため、まずは最も基本的なエンコーディング方法であるという理由から、

画像全体の特徴量を visual words のヒストグラムで表現する Bag-of-Features による方法を用いて実験を行う。また、様々なデータセットにおいて、state-of-the-art の精度を実現^{14) 16)}している Fisher vector によるエンコーディングを使用した場合でも、さらに精度が向上することを確認する。

以上、代表的な 2 つのエンコーディング方法において有効性を確認することで、他のエンコーディング手法においても有効に機能することを期待している。

4.1 Bag-of-Features によるエンコーディング

Bag-of-Features のエンコーディング方法は、まず K 個の visual vocabulary を含む辞書 (codebook) を作ることから始める。つまり、位置情報付き局所特徴量の空間 R^{D+2} 上で、学習画像から抽出した大量の位置情報付き局所特徴量を用いて、 k -means クラスタリングを実施し、ベクトル量子化における K 個のベクトルを探し、それをコードブックとする。次に、1 枚の画像から得られる位置情報付き局所特徴量の集合 $\mathbf{v}_1, \dots, \mathbf{v}_n \in R^{D+2}$ を最も近いクラスに分類し、ヒストグラムを作成する。

4.2 Fisher vector によるエンコーディング

Fisher vector のエンコーディング方法は、あらゆる画像を表現するような確率密度分布モデルを作成することから開始する。学習画像を用いて抽出した大量の位置情報付き局所特徴量を用い、 $D+2$ 次元の空間上で、局所特徴の確率密度分布を混合ガウス分布 (GMM):

$$p(\mathbf{X}|\theta) = \sum_{k=1}^K w_k p(\mathbf{X}|\mu_k, \Sigma_k) \quad (5)$$

$$p(\mathbf{X}|\mu_k, \Sigma_k) = \frac{1}{(2\pi)^{\frac{D+2}{2}} |\Sigma_k|^{\frac{1}{2}}} e^{-\frac{1}{2}(\mathbf{x}-\mu_k)^T \Sigma_k^{-1}(\mathbf{x}-\mu_k)} \quad (6)$$

$$\forall k, w_k \geq 0, \sum_{k=1}^K w_k = 1 \quad (7)$$

で表現する。モデルパラメータの推定には EM アルゴリズムを用いる。ここで

$$\theta = (w_1, \mu_1, \Sigma_1, \dots, w_K, \mu_K, \Sigma_K) \quad (8)$$

はモデルのパラメータで、 $w_k \in R$ は混合比、 $\mu_k \in R^{D+2}$ は平均値、 $\Sigma_k \in R^{(D+2) \times (D+2)}$ は分散共分散行列である。

次に、1 枚の画像から得られる位置情報付き局所特徴量の集合 $\mathbf{v}_1, \dots, \mathbf{v}_n \in R^{D+2}$ を $\mathbf{V}$ とする。このとき、Fisher vector によるエンコーディングは、共分散行列を対角行列と仮定すると、各パラメータに対する対数尤度の勾配

$$\frac{\partial L(\mathbf{V}|\lambda)}{\partial w_i} = \sum_{t=1}^n (\gamma_t(i) - w_k), \quad (9)$$

$$\frac{\partial L(\mathbf{V}|\lambda)}{\partial \mu_i^{(d)}} = \sum_{t=1}^n \gamma_t(i) \left[\frac{v_t^{(d)} - \mu_i^{(d)}}{(\sigma_i^{(d)})^2} \right] \quad (10)$$

$$\frac{\partial L(\mathbf{V}|\lambda)}{\partial \sigma_i^{(d)}} = \sum_{t=1}^n \gamma_t(i) \left[\frac{(v_t^{(d)} - \mu_i^{(d)})^2}{(\sigma_i^{(d)})^3} - \frac{1}{\sigma_i^{(d)}} \right] \quad (11)$$

をそれぞれ計算して、それらを K 個のコンポーネントについてをつなぎ合わせることで次元数 $(2D+5)K$ の固定長のベクトルを得ることができる。ここで $\gamma_t(k)$ は局所特徴量 $\mathbf{v}_n$ が GMM のコンポーネント k に属する確率、 $v_t^{(d)}$ は t 番目の特徴ベクトルの d 次元目、 $\mu_i^{(d)}$ は i 番目のコンポーネントにおける平均ベクトルの d 次元目、 $\sigma_i^{(d)}$ は i 番目のコンポーネントに

おける分散の d 次元目の値である。上記の対数尤度の勾配は、データに適合するように GMM のパラメータが修正すべき方向を表現している。Fisher vector の利点は、異なるデータサイズ集合を特定の長さの特徴ベクトルに変換できる点である。しかしながら、そのままでは Bag-of-Features と同等レベルの精度しか出ないことも知られている。そのため識別に使用する際の特徴ベクトルは Power 正規化と L2 正規化を行うのが一般的である¹³⁾。

5. 実験

5.1 データセット

提案手法の有効性を確認するため、一般物体認識でよく用いられる Caltech-101 データベース⁶⁾を用いて精度の確認を行った。Caltech-101 データベースは 101 種類の物体^{*1}を含む、8677 枚の画像から構成されている。1 つのカテゴリに含まれる画像数は 31 から 800 までとばらつきがある。実験では、データベースを半分に分け、片方を学習に、残りをテストに用いて評価を行った。

5.2 実験条件

局所特微量には SIFT 特微量を用い、サイズを 4 段階に変えながら、画像中の格子点から均等な間隔で (5 ピクセル毎に) 密にサンプリングを行った。1 枚の画像から抽出された SIFT 特微量の数は平均で約 5000 個であった。SIFT 特微量の各要素の値は、L2 正規化を行った後、512 倍した値を用いた。その後、学習データの中からランダムに 100 万個の SIFT 特微量 (128 次元ベクトル) を抽出し、主成分分析を行い、次元圧縮のための行列を計算し、最終的に 128 次元のベクトルを 64 次元に圧縮した。次に、式 (4) に示すように、それぞれの 64 次元のベクトルに対応する座標情報を加えて、66 次元の座標情報付き特徴ベクトルを作成した。その際、ベクトルの各次元の値とスケールをおおよそ合わせるために、 $\lambda_x = \lambda_y = 256$ に設定した。

提案手法の有効性を確認するため、従来の座標情報なしの特徴ベクトル (64 次元) と座標情報付きの特徴ベクトル (66 次元) との比較に加え、Spatial Pyramid Matching との比較実験を行った。この際、エンコーディング方法には、Bag-of-Features と Fisher vector の 2 つの方法を用いた。Spatial Pyramid Matching については、Bag-of-Features に対しては、 $1 \times 1 + 2 \times 2$, $1 \times 1 + 1 \times 3$, $1 \times 1 + 2 \times 2 + 1 \times 3$, $1 \times 1 + 2 \times 2 + 4 \times 4$ の 4 種類、Fisher vector については、 $1 \times 1 + 2 \times 2$ で実験を行った。

Bag-of-Features の辞書作成には、66 次元 (位置情報付きの場合)、64 次元 (位置情報なしの場合) の空間上で、 k -means クラスタリングを用い、クラスタ中心を計算した。visual word の数 (k -means クラスタリングの k の値) は、値を 250 から 10000 まで変更しながら精度比較を行った。

また、Fisher vector を計算するための確率密度分布にはガウス混合モデル (GMM) を用いた。ここでも、66 次元 (位置情報付きの場合)、64 次元 (位置情報なしの場合) の空間を考え、それぞれの空間上で GMM を作成した。ガウス混合モデルのパラメータは、Bag-of-Features の辞書作成に用いたものと同様の 100 万個のデータを用いて、EM アルゴリズムにより推定した。混合数については、32 から 512 まで変化させ、精度比較を行った。Fisher vector によるエンコーディングでは、¹³⁾ に示される

Table 1 Image classification rates of the Bag-of-Features encoding on Caltech-101 dataset. The number of dimensions is shown in brackets (). Leading approach is shown in bold for each number of visual words. The underlined number indicates the best performance in all experiments

visual word の数	位置情報 なし	位置情報 付き (提案手法)	Spatial Pyramid Matching			
			1×1 2×2	1×1 1×3	1×1 2×2 1×3	1×1 2×2 4×4
250	61.25% (250)	63.12% (250)	64.41% (1250)	63.93% (1000)	64.69% (2000)	64.46% (5250)
500	64.30% (500)	65.58% (500)	66.11% (2500)	66.63% (2000)	66.59% (4000)	66.45% (10500)
2000	69.18% (2000)	69.96% (2000)	69.68% (10000)	70.23% (8000)	70.07% (16000)	68.49% (42000)
4000	70.39% (4000)	71.47% (4000)	70.57% (20000)	70.80% (16000)	70.46% (32000)	68.03% (84000)
6000	71.10% (6000)	72.32% (6000)	70.46% (30000)	70.39% (24000)	70.21% (48000)	67.21% (126000)
10000	70.44% (10000)	71.97% (10000)	69.16% (50000)	69.48% (4000)	69.13% (80000)	— (210000)

Fig. 4 Accuracy comparison of the Bag-of-Features encoding with respect to the number of dimensions

ように L2 正規化とパワー正規化を行い、精度の向上を図った。

カテゴリの識別には線形の SVM、具体的には LIBSVM パッケージ¹⁵⁾を用いて識別モデルの作成、評価を行った。

5.3 実験結果

Bag-of-Features エンコーディングを用いたときの評価結果を表 1 に示す。特徴ベクトルに (x, y) 座標の 2 つの値を追加した位置情報付き局所特微量を用いることで、位置情報がない従来の手法に比べ、どの visual word の数に対しても、常に精度が向上していることがわかった。Spatial Pyramid Matching との比較においても、visual word の数が少ないときは、Spatial Pyramid Matching の方が高い精度を示しているが、visual word の数が大きくなるにつれて、位置情報付きの方が精度が高くなり、visual word 数が 6000 のときに、提案手法で最も良い精度 (72.32%) が得られた。また、最終的に識別に用いる特徴ベクトルの次元数に対する精度の変化を図 4 に示す。識別の際の次元数を比較してみると、少ない次元数で精度が高いことが確認できる。

Fisher vector エンコーディングによる結果を表 2 に、次元数と精度の関係を図 5 に示す。ここでも位置情報付き局所特微量を用いた手法は、位置情報なしに比べ、GMM の混合数を変化させた場合にも常に精度が上回っていて、位置情報の埋め込みの効果が見られた。同じ GMM の混合数を用いた場合の Spatial Pyramid Matching との比較においては、Spatial Pyramid Matching の方が若干良い精度が得られているが、識別の際の次元数は莫大になってしまうという問題がある。

*1 BACKGROUND.Google ディレクトリを除いた 101 カテゴリで実験を行った。

Table2 Image classification rates of the Fisher vector encoding on Caltech-101 dataset. The number of dimensions is shown in brackets (). Leading approach is shown in bold for each number of mixtures. The underlined number indicates the best performance in all experiments

GMM の 混合数	位置情報なし	位置情報付き (提案手法)	Spatial Pyramid Matching
			1 × 1 2 × 2
32	70.58% (2080)	74.01% (2208)	75.52% (10400)
64	72.91% (4160)	75.06% (4416)	76.76% (20800)
128	74.06% (8320)	76.62% (8832)	77.08% (41600)
256	74.58% (16640)	77.26% (17664)	76.30% (83200)
512	74.83% (32280)	76.25% (35328)	76.32% (166400)

Fig.5 Accuracy comparison of the Fisher vector encoding with respect to the number of dimensions

5.4 改善点に関する考察

まず、どのような画像に対して改善しているかを確認するため、Spatial Pyramid Matching では識別誤りを起こしたが、提案手法により改善できた例を確認したところ、図3に示した画像に加え、図6のように対象物が中央からずれている場合に対して向上が見られた。図6の上の4枚の画像はSpatial Pyramid Matching の2x2、下の画像は1x3において誤って識別されたが、提案手法では正しく識別できた例である。

一方、すべての方法において識別誤りが出た画像の例を図7に示す。学習データに殆ど含まれていない通常と異なる画像に対して誤りが生じていることがわかる。例えば、図7の左上の画像の正解クラスはlobsterだが、他の学習サンプルは実際の生物のlobsterが写っているのに対し、識別誤りを起こした画像は実際の生物ではないため、識別は困難だったと思われる。図7の右の図の正解クラスはcougar faceであるが、他の画像に比べて極端に大きく写っている点が大きく異なる。また、図7の下の方(正解クラス: mandolin)は、他の画像は実際のmandolinが中央に写っているのに対し、誤った画像は画像の下部に偏って写っていることに加え、実物ではなく人工的なmandolinの絵が描かれているといった点が異なるため、正しく識別できなかったと考えられる。このような画像を正しく識別するには、クラス特有の特徴を取り出すなどの根本的な改良が必要である。

また、メモリ使用量や速度^{*2}に関する計測を行ったところ、識別処理で大きな差異が見られた。例えば、Bag-of-Features エンコーディングを用いて、visual word 数が6000のときを例に

^{*2} Core i7 2GHz の CPU を使用。

Fig.6 Examples of correctly-classified images by our proposed method. Class name: binocular, lamp, camera, sea horse, and starfish

Fig.7 Examples of misclassified images using both spatial pyramid matching and our proposed method. Class name: lobster, cougar face, and mandolin

取ると、識別にかかる平均処理時間は、提案手法では、0.04秒以内に収まるのに対し、Spatial Pyramid Matching の1x1, 2x2, 4x4を考慮した場合は0.87秒かかってしまい、リアルタイム性が要求されるアプリケーションに適応することが難しくなってしまう。さらに、メモリ使用量についても、SVMのサポートベクターをfloat型(4バイト)で保存することを考えると、提案手法では、学習モデルを保存するのに、79.9MBで済むのに対し、Spatial Pyramid Matchingでは、1.78GBと大量のメモリを使用しないといけないという問題が生じる。

5.5 パラメータ変更と識別器統合による精度向上

ここまでの実験では、特徴ベクトルのスケールと座標情報のスケールのバランスを調整するためのパラメータ λ_x, λ_y には固定の値 256 を用いてきた。画像中の位置情報を重視したい場合は、このパラメータの値を大きく設定すればよい。逆に座標情報の影響を低減したい場合、 λ_x, λ_y を小さな値に設定すればよい。例えば、図 1 の Spatial Pyramid Matching の中央の図のように、画像の横方向の位置情報を無視したい場合、 λ_x を 0、またはそれに近い値に設定すればよい。

これらのパラメータの値を変更することで、異なったエンコーディングを行うことができることから、誤り傾向が異なる識別器を構築することが可能と考えた。実際にパラメータを変更させ、各パラメータについて個別に識別器を作成したところ、それぞれの単体の識別器の性能はほぼ変わらないものの、違った誤り傾向を持つことが確認できた。

さらに識別精度を向上させるため、位置情報に加え、

$$\mathbf{v}^T = (v_1, v_2, \dots, v_d, \lambda_x \frac{x}{w}, \lambda_y \frac{y}{h}, \lambda_s s) \in R^{d+3}. \quad (12)$$

として局所特微量のスケールを加えることも可能であることから、 $\lambda_x, \lambda_y, \lambda_s$ の 3 つの値を変更し、ロジスティック回帰を用いて複数の識別器を作成し、各識別器から得られる確率出力の平均を取ることで識別器を統合を試みた。実験では LIBLINEAR パッケージ⁹⁾ のロジスティック回帰を用いた。

Bag-of-Features エンコーディングを用いた実験では、表 1 に示すように、コードブックサイズは最も精度が良かった 6000 に設定し、 $(\lambda_x, \lambda_y, \lambda_s)$ の組み合わせとして、 $(0, 768, 0)$, $(256, 256, 0)$, $(256, 256, 1)$, $(512, 512, 0)$, $(512, 512, 1)$, $(768, 0, 0)$, $(768, 768, 0)$, $(1024, 1024, 0)$ の 8 つの設定で行い、識別器の作成と統合を行った。結果は、統合前の 72.32% に対して 4 ポイント以上向上し、76.80% の識別精度を達成した。Bag-of-Features エンコーディングを用いただけで、Fisher vector エンコーディングの識別性能に近づけることができた。

一方、Fisher vector エンコーディングを用いた実験では、表 2 から、GMM の混合数は最も精度が良かった 256 に固定し、 (λ_x, λ_y) の組み合わせとして $(0.0, 768)$, $(256, 256)$, $(512, 512)$, $(768, 0)$, $(768, 768)$, $(1024, 1024)$ の 6 つの識別器を作成し、統合を行った。こちらは統合前の 77.26% に対して、78.50% と多少の向上に留まった。

6. 結 論

本論文では、画像からのカテゴリ識別の問題に対し、画像中の位置情報を特微量に直接組み込むことにより、Bag-of-Features や Fisher vector エンコーディングの中に位置情報を加えることを提案した。物体認識用データベース Caltech-101 を用いた実験により、本手法の有効性を確認できた。

提案手法では、Spatial Pyramid Matching が抱える、境界領域で極端にエンコードされてしまう問題を防ぐことができ、局所特微量の微妙な位置を保持することができたことが、精度向上につながったのではないかと考えている。また、データベース

の性質によって、Spatial Pyramid Matching の分割数を変更したり、実験により最適な分割数を決定したりする必要もないという利点もある。さらに、識別の際の特徴ベクトルを考えた場合、従来の特徴ベクトルに座標に関連する 2 つの値をつなげるだけで済むため、Spatial Pyramid Matching に比べて、計算コストやメモリ使用の削減にも役立つ。

さらに識別精度を向上させるため、位置情報の強度を調節するパラメータを変更し、複数の識別器を構築し、それらを統合することで、単体の精度からさらに向上することが確認できた。

今後、本手法は画像の中での位置が予め定まっている問題に対して有効であると考えられることから、近年研究がさかんになりつつある詳細画像カテゴリ識別 (FGVC: Fine-Grained Visual Categorization) への適応も検討していきたい。

参 考 文 献

- 1) D. G. Lowe: Object recognition from local scale invariant features, Proc. of IEEE International Conference on Computer Vision, (1999) 1150.
- 2) K. Mikolajczyk and C. Schmid: Scale & Affine Invariant Interest Point Detectors, International Journal of Computer Vision, **60**, 1, (2004) 63.
- 3) D. G. Lowe: Distinctive Image Features from Scale-Invariant Keypoints, International Journal of Computer Vision, **60**, 2, (2004) 91.
- 4) G. Csurka., C. R. Dance, L. Fan, J. Williamowski, and C. Bray: Visual categorization with bags of keypoints, Proc. of European Conference on Computer Vision Workshop on Statistical Learning in Computer Vision, (2004) 59.
- 5) S. Lazebnik, C. Schmid, and J. Ponce: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories, In Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR2006), **2**, (2006) 2169.
- 6) L. Fei-Fei, R. Fergus, and P. Perona: Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories, Computer Vision and Image Understanding, **106**, 1, (2007) 59.
- 7) M. Marszalek, C. Schmid, H. Harzallah, and J. van de Weijer: Learning object representations for visual object class recognition, Visual Recognition Challenge Workshop, in Conjunction with ICCV, (2007).
- 8) H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool: SURF: Speeded Up Robust Features, Computer Vision and Image Understanding (CVIU), **110**, 3, (2008) 346.
- 9) R.-E. Fan, K.-W. Chang and C.-J. Hsieh, X.-R. Wang, and C.-J. Lin: LIBLINEAR: A Library for large linear classification, Journal of Machine Learning Research, **9**, (2008) 1871.
- 10) O. Boiman, E. Shechtman, and M. Irani: In defense of nearest-neighbor based image classification, In Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR2008), (2008) 1.
- 11) J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong: Locality-constrained linear coding for image classification, In Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR2010), (2010) 3360.
- 12) X. Zhou, K. Yu, T. Zhang, and T. S. Huang: Image classification using super-vector coding of local image descriptors, In Proc. of European Conference on Computer Vision (ECCV2010), Part IV, (2010) 141.
- 13) F. Perronnin, J. Sánchez and T. Mensink: Improving the Fisher Kernel for Large-Scale Image Classification, In Proc. of European Conference on Computer Vision (ECCV2010), Part V, (2010) 143.
- 14) K. Chatfield, V. Lempitsky, A. Vedaldi, and A. Zisserman: The devil is in the details: an evaluation of recent feature encoding methods, In Proc. of British Machine Vision Conference (BMVC2011), (2011).
- 15) C.-C. Chang and C.-J. Lin: LIBSVM: A library for support vector machines, ACM Transactions on Intelligent Systems and Technology, **2**, 3, (2011).
- 16) J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek: Image classification with the Fisher vector: theory and practice, International Journal of Computer Vision, **105**, 3, (2013) 222.