# 大規模一般画像認識と画像表現
# Large-Scale Generic Image Recognition and Image Representation

原田達也（Tatsuya Harada）[1,2]
[1]The University of Tokyo, [2]JST PRESTO
harada@isi.imi.i.u-tokyo.ac.jp

## Abstract

近年のインターネットの発達により大量の画像とそれに付随するタグなどの付加的情報が容易に入手可能となり，この大規模な情報を用いて一般画像認識を構築する試みが盛んになってきている．本稿では大規模画像データセットを用いた一般画像認識の潮流を紹介する．また，大規模画像認識を行うにはスケーラビリティを維持するために線形識別機を用いることが多い．線形の識別機であっても十分な識別能力を発揮するためには画像表現が鍵となるため，近年の画像表現手法に関して解説を行う．

## 1. はじめに

任意の画像を柔軟に認識する知能の構築はコンピュータビジョンの究極の目標の一つである．この画像認識技術として自動画像アノテーションが注目されている．従来の画像アノテーションは限られたスモールワールドで機能するものであった．しかしながら，近年のインターネットの発達により大量の画像や動画とそれに付随するタグなどの付加的情報が容易に入手可能となり，この大規模な情報を用いて画像アノテーションエンジンを構築する試みが盛んになってきている．大規模なデータセットから構築された画像アノテーションエンジンを利用することで，今まで不可能であった多種多様な実世界画像を認識可能なレベルに押し上げている．

自動画像アノテーションとは，タグが付与されていない画像に対して複数のタグを自動的に付与する手法を指す．本稿における画像認識は断りがない限りタグの自動付与を指す．画像認識では物体や景色（シーン）のカテゴリを識別する「一般物体認識 [1]」と，対象そのものの固有の特性を特定する「特定物体認識 [2]」に一般的に分類される．画像アノテーションはこの分類よりも幅広い概念であり，付与する複数タグを一つのカテゴリに限定すれば一般物体認識となり，固有の特性に限定すれば特定物体認識となる．実際に画像認識のトップカンファレンスに掲載された論文を見渡しても，画像アノテーションという言葉が一般物体認識・特定物体認識と明確に区別して使用されていないようである．そこで本稿でも幅広い概念で画像アノテーションを取り上げる．

画像アノテーションの枠組みは，画像とそれに付随するタグのペアのデータセットの構築，画像とタグの特徴抽出，画像とタグの関係性モデルの構築，モデルを用いた画像のタグ予測となっている．つまり，データ，画像特徴，モデルが画像のタグ予測性能に貢献するため，この 3 つのサブテーマを取り上げる．また，画像アノテーションを大規模データセットに適応するには，スケーラビリティが特に重要となるため，この点ついても言及する．

また，大規模画像認識を行うにはスケーラビリティを維持するために線形識別機を用いることが多い．線形の識別機であっても十分な識別能力を発揮するためには画像表現が鍵となるため，近年の画像表現手法に関して解説を行う．

## 2. データセット

ここではデータセットを小規模と大規模に分類するが分類の明確な基準はなく，従来よりもデータ数やカテゴリ数が一桁大きければ大規模と呼ぶ場合が多い．大規模データセットに関し，Web から得られるデータに限定する．また，一般的な Web 画像とタグを持つデータセットを基準とし，安定したタグが得られるデータセットの例としてランドマークアノテーション，タグの信頼性が低い例として Web 上の動画像アノテーションを紹介する．

### 2.1. 小規模データセット

狭義の画像アノテーションで標準的に利用されているデータセットは Corel5K[3] である．これは 5,000 枚の画像から構成され，4,500 枚の訓練画像と 500 枚のテスト画像に分かれている．各訓練画像にはあらかじめ 1 から 4 つのタグが付与されており，訓練画像には 371 種類のタグとテスト画像には 260 種類のタグが用いられている．また，約 2 万枚の画像を含む IAPR-TC12[4] がある．これは各画像に付与されたキャプションから一般的な名詞を抽出して利用され，約 300 種のタグから各画像に平均 5 つ程度割り当てる．この他にもインターネット越しの対戦型タグ当てゲームから作成した ESP Game[1] [5] があ

[1]http://www.espgame.org/gwap/

る．この画像のサブセットを利用する場合が多く [6, 7]，2 万枚の画像と 270 種類のタグ，各画像には平均して 5 つが付与されている．

上記は 1 枚の画像に複数タグが付与されたデータセットだが，物体カテゴリ識別用として背景画像を含む 102 カテゴリ，約 1 万枚の画像を含む Caltech101[8] が最も有名である．この他に 256 カテゴリ，約 3 万枚の画像を含む Caltech256[9]，約 20 カテゴリ，約 1 万枚の画像を扱う Pascal VOC[2]がある．シーン識別では 15 カテゴリ，4,492 枚の画像を含む LSP15[10] が標準的に利用される．

従来用いられてきたデータセットは概ね数十から数百カテゴリに対し，画像総数が数千枚から数万枚の規模である．これらの問題は決して易しくないが，多くの研究者がこぞって参加する性能競争となり十分に高い識別性能を得られている．例えば Caltech101 において，2004 年の提案当時，各カテゴリあたり 15 枚の学習画像で識別性能が 16%程度 [8] であったが，2009 年には 75.8%[11] にも達している．限られたデータセット上で高い性能を絞り出すことはスモールワールドに極度に最適化された汎用性のない手法の危険性もはらみ，より一般的で挑戦的な課題に取り組む必然性が出てきている．

### 2.2. 一般的画像アノテーション用データセット

TinyImages[3][12] は 8,000 万枚の $32 \times 32$ の低解像度画像から構成される．これらの画像は WordNet[13] に含まれる全てのカテゴリを Flickr[4]や Google などの画像検索エンジンで検索し収集している．カテゴリ数は 75,062 である．WordNet を利用することでカテゴリに偏りのない画像を収集可能である．大規模な画像データセットを扱うには画像の次元圧縮が必要だが，このデータセットでは単純に解像度を低下させることで実現している．画像の低解像度表現はストレージへの負荷を少なくするだけではなく，識別において重要な情報を失っていないことが実験的に調べられている．

ImageNet[5][14] は WordNet の階層的構造を利用した大規模な画像のオントロジである．2010 年 4 月 30 日現在，11,231,732 枚の画像と 15,589 カテゴリが収集されている．1 カテゴリあたり 500 から 1000 枚の画像が含まれており，TinyImages と異なり画像の質が統制されている点，高解像度の画像（$400 \times 350$ 程度）を扱っている点で異なる．画像検索エンジンの検索精度は約 10%であるという知見 [12] から，各カテゴリの単語と WordNet の上位単語やその単語に対応する他言語を画像検索エンジンに入力して各カテゴリあたり 1 万枚程度（目標収集画像数の 10 倍）の画像を収集する．その画像群から Amazon Mechanical Turk（AMT）[6]を通じ，人力でカテゴリに属する適切な画像を選別して質の高いデータセットを構築している．

ARISTA[15] は 20 億枚の画像を有し，論文で情報が公開されている画像データセットの中で最大である．この論文で，数百万枚のデータセットは Web 画像の一部を表現しているに過ぎず，人の生活になじみの深い絵画，有名人，映画や商品のカテゴリが欠落している点や，WordNet を基盤としたデータセットは，これらのカテゴリを含まないために日常的に利用される画像を反映できていない点を指摘している．オントロジとして Open Dictionary Project[7]の方が Web 検索を反映していると述べている．さらに，従来ノイズとして扱ってきたほぼ類似な画像群（Near Duplicate Image, NDI）のアノテーションにおける重要性を指摘してる点は興味深い．人々が関心を集める対象は NDI を持ちやすく，NDI とそれに付随するタグが大量に得られればその画像に関する有益なタグのみ抽出可能となり，関心を集める画像のアノテーション性能を向上させることができる．

シーン識別に特化したデータセットとして SUN[8][16] がある．このデータセットには 899 カテゴリと 130,519 枚の画像を含む．前述のデータセットと比較すれば小規模であるが，シーン識別に限れば最大である．シーンカテゴリは基本的に WordNet から選択されているが，WordNet に含まれないシーンカテゴリも追加されている．

ImageNet は高品質な画像のオントロジ作成が目的であるが，TinyImages と ARISTA の主張は，大規模な画像データセットがあれば複雑な機械学習手法を利用せずともセマンティックギャップ [17] を回避できる点にある．セマンティックギャップとは画像から得られる特徴と画像に映し出されている意味との間に存在するギャップであり，長い間解決されていない問題である．TinyImages や ARISTA のアプローチは大規模な画像があれば画像特徴間の類似度がそのまま画像間の意味類似度に近づいていくというアイデアに基づく．

### 2.3. ランドマークアノテーション用データセット

Web にアップロードされている画像にはその土地を象徴するランドマークが写っている場合が多く，それらの画像を大規模に収集することで自動的にランドマークアノテーションを行う試みがある．この画像には GPS から得られる撮影時の緯度経度情報（Geotag）を活用できる場合がある．Geotag は画像のアピアランスや画像に付与されている他のタグよりも信頼がおける．さらに同じ場所に大量の画像があれば，その画像はランドマークである可能性が高い．[18, 19] において緯度経度を用いたクラスタリングを行い，生成されたクラスタをランドマークの候補として用いている．[20] では，地理空間をグリッド状に分割し，分割されたタイルの中心位置をクエリとして画像収集を行っている．このように集められた画像は，画像のアピアランスを利用したクラスタリング [18, 20] や，複数の異なる人間が撮影している条件を用いる [18, 20, 19] ことでさらに選抜される．[18] ではデータを集めるために Web 上にあるトラベルガイドを利

[2] http://pascallin.ecs.soton.ac.uk/challenges/VOC/
[3] http://groups.csail.mit.edu/vision/TinyImages/
[4] http://www.flickr.com/
[5] http://www.image-net.org/
[6] https://www.mturk.com/mturk/welcome
[7] http://www.dmoz.org/
[8] http://groups.csail.mit.edu/vision/SUN/

用し，[20] ではデータ収集後に Wikipedia を利用して検証を行っている点が特徴的である．信頼のおける Geotag が利用できることや，ランドマークは一般的な物体よりもアピアランスの分散が小さいことを考慮すると，ランドマークアノテーションは一般的な物体識別よりも一段簡略化された問題設定となる．

### 2.4. 動画像アノテーション用データセット

動画理解のワークショップとして TRECVID[9]が有名であるが，最近では YouTube[10]などの動画共有サイトのデータを大規模に収集し，動画像アノテーションに用いる例が出てきている．動画共有サイトにアップロードされる動画にはユーザが付与したタグが存在するが，短すぎたり，もしくは欠落しているため動画の内容を十分に表現できていない問題が存在する．つまりタグは内容の一部しか表現しておらず，適切な訓練データが不足している．また，アップロードされた動画の質の低さや扱うテーマの幅の広さから従来用いられている認識アルゴリズムは適さないとの報告もある [21]．よって Web 上の画像認識と比較してもチャレンジングな課題と言える．不足情報は一般的な Web ページの情報を集めて補う場合が多く，[22] においては 8 万本の YouTube 動画と関連する 7 万の Web ページを収集し，[23] に至っては 5300 万本の YouTube 動画と 1 億枚の Web ページを収集して実験を行っている．

### 2.5. 大規模画像データセットの構築

大規模画像データセットを自前で作成する場合，数百万枚程度であれば，画像共有サイトや画像検索サイトから提供されている API を用いることで比較的容易に実現できる．しかしながら，様々なバイアス（カテゴリ，画像アピアランスなど）や，タグや画像に含まれるノイズに悩まされる場合が多い．2.2のデータセットのように WordNet を利用してカテゴリ選択にバイアスをかからないようにしたり，収集したデータセットに AMT を通じて人力でノイズを減らすことが考えられる．一方，ジャンクデータも含め大量収集後，自動的に有用なデータのみをフィルタリングする試みがある [24]．しかし画像に付与されたタグが必ずしも画像の持つ意味に対応しているとは限らず，弱い関係を持つに過ぎないため（weakly-tagged image）自動フィルタリングは困難な課題である．[25] ではこの問題に取り組み Web から得られた大量の画像中からジャンクデータを取り除き，画像に適切なタグを付与し直す試みを行っている．

## 3. 画像特徴

まず画像特徴を局所特徴と大域特徴に分類する．局所特徴から大域的な特徴を生成するフレームワークとして Bag of Features（BoF）を説明する．さらに BoF を基準として特徴ベクトルの疎密性に着目する．

[9]http://trecvid.nist.gov/
[10]http://www.youtube.com/

### 3.1. 局所特徴と大域特徴

画像特徴は，局所特徴と大域特徴に分類できる．よく用いられる局所特徴として，輝度勾配ヒストグラムを基盤とした SIFT[26]，HOG[27]，SURF[28]，ある着目領域と周囲の類似度を表現した Self Similarity[29]，着目点からの距離に応じて形状を曖昧に表現し，着目点と形状の曖昧な表現との関係を記述する Geometric Blur[30] などがある．大域特徴として，HLAC[31]，Gist[32]，カラーヒストグラムなどがある．特徴点の検出方法にもよるが局所特徴は一枚の画像あたり数百から数万点得られる．そのため特徴点検出手法や特徴点における記述子の計算コストは，データセットが大規模になれば大きな負担となる．しかしながら各特徴点の検出や記述は他の特徴点と独立に計算できる場合が多く，SiftGPU[11]や CUDA SURF[12]のように近年盛んに開発されている GPU を用いて高速化が実現されている [33]．

### 3.2. Bag of Features

局所特徴から一つの画像特徴を得るには多数存在する局所特徴をまとめる枠組みが必要となる．画像認識で標準的に利用されている枠組みが Bag of Features（BoF）[34] である．BoF の一般的な生成方法は，訓練データの局所特徴の集合に対し，k-means によりクラスタリングを行う．得られた $K$ 個のセントロイドをコードワードとして，画像特徴は局所特徴のコードワードに関するヒストグラムとして表現する．得られたコードワードの集合をコードブックという．BoF はコードワード数決定の不確定性があるものの画像認識において高い性能を示すことが実験的に知られている．しかしながら BoF は局所特徴のベクトル量子化であり，ベクトル量子化の過程で多くの情報が失われている．そのために「失われた情報」を補完することで BoF の表現能力を向上させる試みがなされている．BoF の一つの解釈は，局所特徴が分布する特徴空間における混合ガウス分布を用いた確率密度分布推定であり，コードワードが各ガウス分布の平均に対応する．混合ガウス分布を仮定して BoF を改善している例として [35] や [36] がある．同様にカーネル密度推定による BoF の例として [37] がある．一般的な BoF は一つの局所特徴を一つのコードワードに割り付けるが，[38] は複数コードワードに対応させる柔らかい割り付けを行っている．さらに BoF を生成モデルと解釈することで Fisher Kernel の適用が可能となる [39]．

さらに BoF は局所特徴間の画像空間情報が失われている問題がある．画像空間情報を表現する手法として Spatial Pyramid 表現 [10] がある．これは画像を一定間隔のグリッドに分割し，分割されたセル内で BoF のヒストグラムを求める．分割の粒度（レベル）を変えて得られた全てのセルのヒストグラムをつなげて一つの特徴ベクトルとする．Spatial Pyramid 表現に適したカーネル（Spatial Pyramid Matching Kernel, SPM Kernel）と SVM

[11]http://www.cs.unc.edu/ ccwu/siftgpu/
[12]http://www.mis.tu-darmstadt.de/surf

の組み合わせは現在最も利用されている一般物体認識手法の一つである．また，BoF そのものに空間情報を埋め込む手法として [40, 41] などがある．BoF は画像情報のみから作成するが，訓練データにカテゴリが与えられる場合にはその情報を用いて判別的なコードブックを設計可能である [42, 43].

BoF を大規模データに適応するにはいくつかの問題がある．一つ目は学習時におけるコードブックの生成であり，二つ目は局所特徴のコードワードへの割り当てである．BoF の表現能力を向上させるためには数千から数万のコードワードが必要である．また，1 枚の画像から得られる局所特徴が数百から数万となるため，膨大な組み合わせの距離計算を全て行うのは非現実的である．この問題には kd-木と階層的 k-means を用いた高速な近似最近傍探索手法 [44]，階層的 vocabulary tree[45]，Locality-sensitive hashing[46] などの利用が考えられる．[47] では kd-木の空間分割法を改良し，PCA kd-木 [48] よりも高速かつ高性能な最近傍探索手法を提案している．この他にも，膨大なデータの画像一枚あたり数万次元の特徴をそのまま保持しておくことは，ストレージへの負担や検索などの計算コストからも非効率であり，効率的なデータ表現手法が求められる．特に [49] は BoF と Fisher Kernel から導出されるコンパクトなデータ表現手法を提案し，高々20 byte で元の BoF と同等の検索性能を有することを示している．

### 3.3. 疎な特徴表現

BoF と SPM カーネル SVM の組み合わせは高い性能を示すが学習時における計算複雑度が訓練サンプル数 $n$ に対して $\mathcal{O}(n^2)$ から $\mathcal{O}(n^3)$ であり，大規模データへの適応は難しい．そこで [50] は訓練時の計算複雑度が $\mathcal{O}(n)$ の線形 SVM でも高い性能を示す局所特徴から一つの画像特徴を生成する枠組み（ScSPM）を提案している．BoF では局所特徴が一つのコードワードに対応するのに対し，スパースコーディングは局所特徴の少数のコードブックへの割り当てを許容する手法である．BoF と SPM の手順は，1）局所特徴の抽出，2）局所特徴のベクトル量子化（コーディング），3）画面空間上に存在する量子化された局所特徴をまとめて一つの特徴量で表現（プーリング），の三段階に分けられる．ScSPM は，コーディングの過程で $L_1$ ノルムの正則化を用いて特徴ベクトルの非ゼロ要素の数を減らすスパースコーディングを採用している．プーリングの過程では各要素の平均を特徴量として採用するのでなく，各要素の最大値を採用している．さらに画像をセルに分割し，各セルでスパースコーディングされた特徴を Spatial Pyramid 表現とすることで一つの特徴量を得る．ScSPM は標準的な物体認識データセットにおいて現在最も高い性能を示し，他の研究グループでも詳しく検証されている [51]．ただし，局所特徴群からスパースコーディングを行うには画像一枚ごとに最適化計算を行う必要があり計算コストが比較的にかかる．また，[52] において，同じ特性を表現する特徴を一つのグループとみなす group sparsity[53] を用いて画像特徴を改良しアノテーション性能を向上させている．RGB と HSV は色情報を表現しているので同じグループ，Gabor 特徴はテクスチャを表現しているので RGB や HSV とは異なるグループと考える．同じグループに属する場合 $L_1$ ノルムの正則化，異なるグループの場合は $L_2$ ノルムの正則化を用いることで特徴選択を行っている．

### 3.4. 密な特徴表現

密な特徴表現をまじめに行うには，3.2で述べたように混合ガウス分布等の表現自由度が高いモデルを用いるのが一般的である．一般的に特徴空間は高次元（SIFT であれば 128 次元）であるのに対し，一枚の画像から得られる局所特徴は高々数万点である．高次元の特徴空間において，限られたサンプル数から自由度の高いモデルを適切に推定できるか分からない．[54] では，特徴空間における局所特徴の分布を一つのガウス分布で推定し，その平均と分散の要素を特徴量とする Global Gaussian アプローチを提案している．これに情報幾何から得られる類似度を合わせることでシーン識別において最新の手法と同等の性能が出ることが示されている．また，[55] において，任意の局所記述子に局所空間情報と大域空間情報を埋め込む密な特徴表現手法を提案している．まず，任意の局所記述子の画像の部分領域における平均，分散，局所自己相関を計算し，それらの要素を並べた領域特徴量を生成する．平均，分散までの計算は Global Gaussian アプローチと同じになる．次に各領域特徴の重み付け平均を計算し，一つの画像特徴を得るが，重みの計算にカテゴリを利用した判別的な手法を利用している．上記二つの手法は混合ガウス分布のパラメータ推定のような繰り返し計算が必要なく，特徴抽出の計算コストが比較的低い．また，確率的線形判別分析をベースとした識別機で最新の手法と同等の性能を出しているためスケーラビリティが高い．

## 4. モデル

ここでは，識別機の学習，識別機によるアノテーション手法を含めてモデルと呼ぶことにする．モデルに関し，小規模と大規模データセットに用いられるものをそれぞれあげ，大規模データセットには複雑なモデルよりもスケーラビリティが重視される現状を見ることにする．

### 4.1. 小規模データセットに用いられるモデル

狭義の画像アノテーションで多く用いられているモデルはノンパラメトリックなモデル [6, 7, 56] である．ノンパラメトリックなモデルとは，入力画像と訓練画像との距離を計算し，距離に応じた重みで各訓練画像に付与されたタグを足しあわせ，入力画像のタグを予測するモデルである．また，タグ毎にカテゴリ識別機を構成する判別的モデルがある [57]．しかしながら訓練画像には複数タグが付与されており，タグ同士の関係性が失われる問題がある．そこで複数のタグの共起頻度を用いて作成されたグラフ構造と条件付き確率場（CRF）を用いて判別的なモデルを生成する手法が提案されている [58]．さ

らに，画像とタグから得られる潜在的な話題を扱うことでタグを予測するトピックモデルがある．[59] では，画像の潜在的な話題とテキストの潜在的な話題の関係性を線形ガウシアン回帰で表現するモデルを提案している．これは [60] と比較して，異なるモダリティで異なる話題を扱え，複数の画像領域からタグに影響を与えることができるといったメリットがある．

カテゴリ識別で最も広く用いられるモデルは Multiple Kernel Learning（MKL） [61, 62] である．これは各画像特徴毎に最適なカーネルを準備し 1-vs-all の多クラス SVM を構築する．各特徴のカーネルに識別への貢献度に応じた重み付けを行い統合することで一つの識別機を構成する．また，ノンパラメトリックな手法として Naive Bayes Nearest Neighbor（NBNN）[63] がある．NBNN はクエリ画像の全ての局所特徴に対して，各クラスに属する全ての局所特徴との最近傍点を求める．各クラスの最近傍点との距離の総和を計算し，距離の総和が最も短いカテゴリをクエリ画像のカテゴリとする．単純な枠組みでありながら高い性能を示すことが実験により確かめられている．この他にも，ローカルラーニングを用いることで性能向上が期待できる．局所識別機の学習をマルチタスク学習の問題としてとらえ物体認識に利用した例として [11] がある．

### 4.2. 大規模一般画像データセットに用いられるモデル

TinyImages では，Nearest Neighbor（NN）法を利用してアノテーションを行っている．TinyImages にはタグにノイズが多く含まれているため，アノテーション時に WordNet の意味階層構造を活用したノイズ低減手法を提案している．ある画像類似度のもとでの $K$ 最近傍点を探し，近傍点のカテゴリの属する枝に存在するカテゴリに等価な重み付けで投票を行う．そして，各意味レベルで最も得票を得たカテゴリを入力画像のカテゴリとして採用する．この手法は単純であるが，$K$ 最近傍にノイズが多くあったとしても正確な識別が可能なことが示されている．また，データセットの数が増加すれば識別性能が向上することや，カテゴリがより特定のものになると性能が下がることが報告されている [12].

ImageNet では 2009 年時点で 320 万枚の画像と 5,247 種類のカテゴリで実験を行っている．ImageNet は TinyImages と比較して高解像度の画像が含まれるために画像特徴量ベースの認識を試すことが可能である．そこで [12] における手法と NBNN の比較を行い，この規模のデータセットにおいても局所特徴（SIFT 記述子）を活用した方が高い性能を示すことが報告されている [14]. また，ImageNet の階層構造を利用することで物体識別性能を向上させる手法（tree-max 識別機）が提案されている．具体的には対象カテゴリの識別時，そのカテゴリの子カテゴリ全てに対して識別を行い，対象カテゴリのスコアはその対象カテゴリと全ての子カテゴリの識別機のスコアの中で最高の値を採用する．AdaBoost を基盤とした識別機 [64] を用いた結果，単独のカテゴリ識別機を用いるよりも tree-max 識別機が高い性能を示すことが報告されている．各カテゴリ識別機の負サンプルは他のカテゴリからランダムに 10 枚ずつ集めることで，訓練サンプルを減らしている．

ARISTA では，NDI を用いたアノテーションによって従来困難であったカテゴリを扱うことが可能になると主張している．画像に付随する情報として，画像のキャプション，URL，Web ページのタイトル，画像周辺の Web ページのテキストが利用されている．アノテーション手法として Search Result Clustering (SRC)[65] と Majority Voting (MV) の二つが提案されている．SRC は [66] でも有効性が示されている手法で次のように計算する．(1) NDI 群に付与されたテキストの n-gram を抽出し，複数種類の突出度（TF-IDF[13]，クラスタ間類似度，クラスタのエントロピなど）を計算する．(2) 複数種類の突出度を一本のベクトルとして，事前に学習しておいた回帰モデル（線形，ロジスティック回帰など）に入力し，一つの突出度スコアとする．(3) (2) で計算された突出度で n-gram をランキングし，ランキングの高い n-gram をタグとして画像に付与する．MV は，NDI 群に付与されているテキストの頻度を用いてタグをランキングする手法である．データセットの規模が大きくなれば SRC，MV ともに平均再現率が上昇するが，平均適合率は頭打ち（SRC）もしくは下降（MV）することが実験により示されている．また，8,000 万枚のデータセット規模は認識性能の観点や画像が表現する概念を網羅する観点からも不十分であると指摘している．

SUN データセットでは，複数の特徴とそれぞれに適したカーネルを用いてシーン識別を行っている．カーネル法を直接用いた数少ない例であるが，データセットの規模は 899 カテゴリ，130,519 枚の画像であり TinyImages や ARISTA と比較すれば相当コンパクトである．識別機としてヒストグラムベースの特徴には $\chi^2$ カーネルを用い，1-vs-all の多クラス SVM を構成する．カーネルの重み付けには各識別機の精度に応じた値を利用している．

[67] において，Flickr から収集した約 300 万枚のタグ付きデータセットを用い画像とタグから得られる潜在空間を獲得し，その潜在空間においてノンパラメトリックにアノテーションを行っている．[56] との違いは，潜在空間の確率的構造を活用しサンプル間の距離尺度を適切に改善している点にある．この手法はナイーブな実装でも学習，アノテーションとも訓練サンプル数 $n$ に対し $\mathcal{O}(n)$ の計算複雑度であり逐次学習や近似最近傍探索によって高速化が可能である．そのためセマンティックギャップを緩和しつつスケーラブルな手法となっている．また，ここで用いられた距離尺度は他の線形次元圧縮に基づく距離尺度と比較しても高いアノテーション性能を示すことが実験的に分かっている [68]．これと似たアプローチとして [69] がある．これは，単語と画像の両方が表現された低次元の空間への写像を学習し，この空間を利用することで意味的に一貫したアノテーションを実現している．

[13] 単語の出現頻度（TF）と文書の逆出現頻度（IDF）を用いた文章中の単語の判別的なスコアの計算手法

### 4.3. ランドマークデータセットに用いられるモデル

[19] では，最初に Flickr から収集した 3,000 万枚の画像群を Geotag の位置情報を用いてクラスタリングし，空間上の分布のピークをカテゴリと見なしている．ユーザが付与したタグではなく Geotag がカテゴリの教師になっている．この操作により 200 万枚の訓練画像にカテゴリが付与される．次に，画像特徴とユーザタグのテキスト特徴から，各カテゴリごとに多クラス SVM[70] を用いて識別機を構築する．画像特徴のみ，テキスト特徴のみの識別結果よりもそれらを合わせた特徴の性能が高いと報告している．また，画像特徴のみならずテキスト特徴や撮影者の時系列情報を structured SVM[71] を用いて識別機を構成すれば，より高性能な識別が可能であると述べている．

[18] では，まず位置情報を持つ画像や旅行ガイド Web ページを利用し，ランドマーク候補画像群を作成している．この候補画像群から局所特徴マッチングとクラスタリングを用いてランドマークの視覚モデルを構築する．視覚モデル作成には，画像データセットの任意のペアを取りだし，局所特徴群の比較により画像間で合致する領域とそのスコアを計算する．局所特徴には LoG[72] と Gabor wavelet を利用する．そして合致した画像領域を頂点とする重み付け無向グラフを生成する．この無向グラフをクラスタリングし，得られたクラスタがランドマークとその視覚モデルに相当する．クエリ画像に対し，局所特徴のマッチングにより近傍画像群を探索する．クエリ画像と最近傍画像群との間でグラフを用いたスコアを計算し，クエリ画像のランドマーク識別を行う．

[20] では，はじめに Geotag と Wikipedia を用いた検証により収集されたデータから物体クラスタ（ランドマーク）を作成する．この物体クラスタは物体やイベントを表現する写真で構成され，Wikipedia の記事とも関連するタグが付与されている．クエリ画像に対する物体クラスタの検索は BoF による画像インデッキシングにより実施される [73]．検索された物体クラスタはコードブックの TF-IDF を用いたスコアにより順位付けが行われる．さらに面白いことに，物体クラスタに属する画像群を用いることで物体セグメンテーションが可能となる．物体クラスタに含まれる画像内の局所特徴が他の画像の局所特徴とマッチした回数をカウントする．マッチした回数の多い局所特徴のみを抽出し，それらを囲う Bounding Box を推定することでセグメンテーションが行われる．画像のアノテーションでは，前述した手法によりクエリ画像に対し物体クラスタの順位付けを行い，一位の物体クラスタに付与されているタグをそのままクエリ画像に付与する．同時に物体クラスタを用いてクエリ画像の Bounding Box を推定することで，セグメンテーションとアノテーションの同時実行が可能となる．

### 4.4. Web 上の動画像データセットに用いられるモデル

Web の動画共有サイトのアノテーションでは不完全なタグが問題となる．このように教師データが不足する場合 co-training[74, 75] や半教師付学習 [76] の枠組みを利用するのが一般的であるが，高次元かつ膨大なデータを用いた学習には効率が悪い．そのため不足したタグを補完する目的で別のドメインの情報も積極的に利用する場合が多い．

[23] では，次の手順でタグやタイトルから抽出される n-gram と動画との関係性を学習する．(1) 動画から映像，音声に関する特徴量を計算し，付与されていたタイトルやタグから n-gram を抽出する．(2) n-gram と動画との対応表を作成する．(3) 各 n-gram に対し，動画像がその n-gram に対応するか判定する識別機を AdaBoost を用いて学習する．一方，動画に付与されていないカテゴリにも対応するため，Web から大量の文章を収集し，n-gram とカテゴリの関係を学習する．入力動画に対して，カテゴリに属する全ての構成要素（n-gram）の識別を行い，それらのスコアの平均をカテゴリのスコアとする．これにより動画に対してカテゴリを付与することが可能となる．実験では 1,500 名の評価者の結果とアルゴリズムにより出力した結果を比較し，人と同様のカテゴリ付与が可能であると述べている．

[22] では動画に付与されたタグを補うため，異なるソースの情報の統合手法（tree-DRF[77]）を提案している．まず動画に人手でタグを付与し種データとする．種データの動画を閲覧した直後に閲覧した動画データ，タグを用いて検索して得られた動画データを収集する．これとは別に人手でタグを付与した Web ページデータも収集する．追加収集したデータはそれぞれ種データと統合される．各データを映像とテキストに分割し，映像を入力とする識別機，テキストを入力とする識別機を構成する．Web ページデータはテキストのみなので，各カテゴリの 3 データに関し，合計 5 種の識別機が得られる．各カテゴリの識別機出力をまとめ，そのカテゴリの特徴ベクトルとする．そしてカテゴリ間の関係を表現した条件付き確率場を用い，全特徴ベクトルと全カテゴリを関連させることで，複数ソースの情報を統合した最終的な識別機が得られる．このカテゴリ間の階層的な構造の利用は，ノイズの軽減につながり，SVM と逐次的 co-training よりも高い性能を示したと報告している．

### 4.5. カーネル法の利用

大規模データセットにカーネル法を直接用いることは困難であり，今まで紹介した大規模データセットにもほとんど利用されていないことが分かる．しかし，カーネル法は高い性能を示すことが分かっており大規模データへの拡張が望まれる．例えば訓練データから少数のランダムサンプルを抽出し，カーネル PCA を構築する．この少数サンプルで構築されたカーネル PCA を用いて射影された空間でスケーラブルな識別機を構成することが考えられる．またカーネル SVM において，特徴空間への陽なマッピングが得られるのであれば，特徴空間における線形 SVM の構築問題となるので大規模データへの可能性が広がる．[78] では，BoF の特徴ベクトルの平方根を計算した特徴ベクトル（square-rooting BoF）が Bhattacharyya カーネルに対応するマッピングであること

を明らかにしている．そのために square-rooting BoF と線形 SVM の組み合わせは，元の BoF と線形 SVM の組み合わせよりも大幅に性能が向上することを示している．

## 5. 画像表現

ここでは，前の章と説明が重複する部分もあるが，線形識別機であっても高い性能を示す画像表現手法に関して少し詳しく述べる．画像表現手法では局所記述子は与えられたとして（一般には SIFT 記述子），そこから一枚の画像を表現する特徴ベクトルの獲得過程に焦点を当てている．

### 5.1. Bag of Features

データベースに存在しない物体のカテゴリレベルの認識を**一般物体認識（Generic Object Recognition）**と呼ぶが，一般物体認識を行うには局所特徴同士の「堅い」比較で**類似度（Similarity）**評価を行うのではなく，画像の持つ局所特徴の統計量を画像特徴と見なして類似度評価を行えば「柔らかい」比較が可能となる．局所特徴群から，その統計量を計算する手法として **Bag of Features（BoF）**[34, 79] が広く利用されている．BoF は文章特徴である **Bag of Words（BoW）**のアナロジーから生まれた特徴である．BoW は単語の並び順，文法などを考慮しない文書特徴であり，例えば文章中に出てきた単語のヒストグラムが利用される．BoF は訓練集合から代表的ないくつかの局所特徴を取り上げ，画像の中に代表的な局所特徴がいくつ出現するかヒストグラムで表現したものである．BoF は **Bag of Visual Words（BoVW）**とも呼ばれる．BoF の計算プロセスを以下に示す．

1. 訓練画像集合が与えられているとする．
2. 訓練画像群から各画像に対して局所特徴を抽出する．
3. 全ての局所特徴から $K$ 個の代表的な局所特徴を選択する．選択した代表的な局所特徴を**コードワード（codeword）**と呼び，選択されたコードワードの集合を**コードブック（codebook）**と呼ぶ．コードワードにそれぞれ $w_1, \ldots, w_K$ とラベルを付与する．コードブックは**辞書（dictionary）**とも呼ばれる．
4. 全ての局所特徴をいずれかのコードワードに対応させる．この操作により，全ての局所特徴に $w_1, \ldots, w_K$ のラベルが付与される．
5. 各画像において，コードワードに関するヒストグラムを計算する．つまり，ある画像に $w_k$ とラベル付与された局所特徴の数をカウントする．コードワードのヒストグラムをその画像の特徴量（特徴ベクトル）とする．つまり特徴ベクトルの次元は $K$ となる．

上記の (3)，(4) のステップで局所特徴群から代表的な局所特徴の選択，局所特徴群を代表的な局所特徴への割り当てを行っているが，BoF のフレームワークでは $K$-means を利用することが多い．BoF を利用したカテゴリ識別手法では**カーネル法（kernel method）**と**サポートベクトルマシン（Support Vector Machine, SVM）**の組み合わせが広く利用されている．カーネルとして，**カイ 2 乗カーネル（$\chi^2$ kernel）**や**インターセクションカーネル（intersection kernel）**が BoF との相性がいいことが実験的に示されている．

### 5.2. 混合ガウス分布による Bag of Features の改良

BoF においてコードワードを作成する目的はいくつかある．

- 類似した局所記述子を最近傍のコードワードに割り当てることで局所記述子の表現にある程度のロバストネスを持たせる．
- 全ての画像に対して，局所特徴を同じコードブックに適用することで同じ長さの特徴ベクトルを得る．
- 局所特徴間の幾何学的関係は視点依存であるので，局所特徴の相対的な位置関係を無視することで識別のロバストネスが向上できる．

さらに BoF の利用によりテキスト分類のテクニックをそのまま画像分類に適用できるメリットもある．

しかしながら BoF には，識別性能がコードワードの選び方に依存する．また，局所記述子のヒストグラムを特徴空間における局所記述子の確率密度分布推定と考えると，ヒストグラムによる表現は粗い推定と言える．そのために確率密度分布をより正確にすれば識別性能向上につながると考えられる．そこで [35] では**混合ガウス分布（Gaussian Mixture Model, GMM）**を用いることで，BoF 表現を改善する試みを行っている．

混合ガウス分布はガウス分布の線形重ね合わせで書ける．

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \sum_{k=1}^{K} \pi_k p_k(\boldsymbol{x}) \qquad (1)$$

ここで $\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), p_k(\boldsymbol{x})$ は**混合要素（mixture component）**であり，平均 $\boldsymbol{\mu}_k$ と分散 $\boldsymbol{\Sigma}_k$ を持つ．$\pi_k$ は混合係数である．混合ガウス分布のパラメータ集合を $\boldsymbol{\theta} = \{\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^{K}$，データ集合 $\mathcal{X} = \{\boldsymbol{x}_n \in R^D\}_{n=1}^{N}$ とすると，混合ガウス分布の**最尤法（maximum likelihood estimation）**によるパラメータ推定は次のように求められる．

$$\boldsymbol{\theta}_{ml} = \arg\max_{\theta} \log p(\mathcal{X}|\boldsymbol{\theta}) = \arg\max_{\theta} \sum_{n=1}^{N} \log p(\boldsymbol{x}_n|\boldsymbol{\theta}) \quad (2)$$

この対数尤度関数を最大化するパラメータは閉形式の解析解で得られないために **EM アルゴリズム**を用いてパラメータを求める．混合ガウス分布のための EM アルゴリズムは以下の通りである．

**E-step**

$$\gamma_n(k) = p(k|\boldsymbol{x}_n, \boldsymbol{\theta}^{(t)}) = \frac{\pi_k p_k(\boldsymbol{x}_n)}{\sum_{j=1}^{K} \pi_j p_j(\boldsymbol{x}_n)} \tag{3}$$

**M-step**

$$\pi_k^{(t+1)} = \frac{N_k}{N} \tag{4}$$

$$\boldsymbol{\mu}_k^{(t+1)} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_n(k) \boldsymbol{x}_n \tag{5}$$

$$\boldsymbol{\Sigma}_k^{(t+1)} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_n(k) (\boldsymbol{x}_n - \boldsymbol{\mu}_k^{(t+1)})(\boldsymbol{x}_n - \boldsymbol{\mu}_k^{(t+1)})^\top \tag{6}$$

$$N_k = \sum_{n=1}^{N} \gamma_n(k) \tag{7}$$

ここで事後確率 $\gamma_n(k) = p(k|\boldsymbol{x}_n, \boldsymbol{\theta}^{(t)})$ は**負担率（responsibility）**とも見なすことができる．

混合ガウス分布を利用するメリットとして，BoF は局所特徴とコードワードへの距離が単なるユークリッド距離で計量されるが，混合ガウス分布を構成する各ガウス分布がそれぞれ共分散を持つために共分散を考慮した距離計量を利用できることがあげられる．また，BoF は局所特徴が一つのコードワードのみに割り当てられるが，混合ガウス分布では局所特徴と多くのコードワードとの関係を表現できるので，特徴空間における局所特徴の位置に関する情報をエンコードできるメリットもある．しかしながら，デメリットして混合ガウス分布表現は BoF と比較してパラメータが多い．混合ガウス分布は $\mathcal{O}(K(D^2/2 + D))$ のパラメータ数であるが，BoF は $\mathcal{O}(KD)$ ですむ．そのため，混合ガウス分布は訓練データに対して**過剰適合（overfitting）**する可能性があり，混合ガウス分布の学習時に**正則化（regularization）**を行う必要がある．

そこで混合ガウス分布のパラメータに関する事前知識を導入し，**事後確率最大化（maximum a-posterior, MAP）**によりパラメータを求めることにする．

$$\boldsymbol{\theta}_{map} = \arg\max_{\theta} \log p(\mathcal{X}, \boldsymbol{\theta}|\mathcal{H}) \tag{8}$$

$$= \arg\max_{\theta} \left( \log p(\mathcal{X}|\boldsymbol{\theta}) + \log p(\boldsymbol{\theta}|\mathcal{H}) \right) \tag{9}$$

ここで，$\mathcal{H}$ は事前知識を決定する**ハイパーパラメータ（hyper parameter）**である．また式 (9) の導出には $\mathcal{X}$ はハイパーパラメータ $\mathcal{H}$ と条件付き独立であるという仮定を利用している．式 (9) の第一項は対数尤度であり，第二項はパラメータの事前知識のペナルティとなっている．

式 (9) 問題を解くためには事前知識に関する表現を定める必要あり，ここでは**共役事前分布（conjugate prior）**を導入する．具体的には，平均には正規分布 $\mathcal{N}(\boldsymbol{\mu}_k|\boldsymbol{\nu}_k, \eta_k^{-1}\boldsymbol{\Sigma}_k)$，共分散には**ウィシャート分布（Wishart distribution）** $\mathcal{W}(\boldsymbol{\Sigma}_k^{-1}|\alpha_k, \boldsymbol{\beta}_k)$，混合係数には**ディリクレ分布（Dirichelet distribution）** $\mathcal{D}(\boldsymbol{\pi}|\boldsymbol{\gamma})$ を利用する．これらの分布を利用することで混合ガウス分布のパラメータに対する事前分布は次のようになる [80].

$$p(\boldsymbol{\theta}|\mathcal{H}) = p(\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}|\boldsymbol{\nu}, \boldsymbol{\eta}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$$

$$= \mathcal{D}(\boldsymbol{\pi}|\boldsymbol{\gamma}) \prod_{k=1}^{K} \mathcal{N}(\boldsymbol{\mu}_k|\boldsymbol{\nu}_k, \eta_k^{-1}\boldsymbol{\Sigma}_k) \mathcal{W}(\boldsymbol{\Sigma}_k^{-1}|\alpha_m, \boldsymbol{\beta}_m) \tag{4}$$

式 (9) を最大化するパラメータを EM アルゴリズムによって求める．これを **MAP-EM** と呼ぶ．

**E-step**

$$\gamma_n(k) = p(k|\boldsymbol{x}_n, \boldsymbol{\theta}^{(t)}) = \frac{\pi_k p_k(\boldsymbol{x}_n)}{\sum_{j=1}^{K} \pi_j p_j(\boldsymbol{x}_n)} \tag{10}$$

**M-step**

$$\pi_k^{(t+1)} = \frac{\sum_{n=1}^{N} \gamma_n(k) + \gamma_k - 1}{N + \sum_{k=1}^{K} \gamma_k - K} \tag{11}$$

$$\boldsymbol{\mu}_k^{(t+1)} = \frac{\sum_{n=1}^{N} \gamma_n(k) \boldsymbol{x}_n + \eta_k \boldsymbol{\nu}_k}{\sum_{n=1}^{N} \gamma_n(k) + \eta_k} \tag{12}$$

$$\boldsymbol{\Sigma}_k^{(t+1)} = \frac{1}{C_k} \left( \sum_{n=1}^{N} \gamma_n(k) (\boldsymbol{x}_n - \boldsymbol{\mu}_k)(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^\top + \eta_k (\boldsymbol{\mu}_k - \boldsymbol{\nu}_k)(\boldsymbol{\mu}_k - \boldsymbol{\nu}_k)^\top + 2\boldsymbol{\beta}_k \right) \tag{13}$$

ここで，$C_k = \sum_{n=1}^{N} \gamma_n(k) + 2\alpha_k - d$ である．

このように訓練データの局所特徴群から MAP-EM により混合ガウス分布のパラメータを推定した後に，画像一枚の特徴量 $\boldsymbol{f}$ を，混合ガウス分布の各コンポーネント $k$ に対する対象画像に含まれる局所特徴群の負担率の和で表現する．

$$\boldsymbol{f} = \frac{1}{N} \sum_{n=1}^{N} [\gamma_n(1), \cdots, \gamma_n(K)]^\top \in R^K \tag{14}$$

さらに [35] では，カテゴリを意識した特徴を生成するためにカテゴリ毎に混合ガウス分布を学習し，カテゴリ毎の特徴を結合することで一つの特徴ベクトルを作成している．つまり，カテゴリ $c$ の特徴を $\boldsymbol{f}(c)$ とすると次のように表現できる．

$$\boldsymbol{f} = [\boldsymbol{f}(1)^\top, \cdots, \boldsymbol{f}(C)^\top]^\top \in R^{CK} \tag{15}$$

もちろん，これはクラス数が増えれば巨大なベクトルとなるので**主成分分析（PCA）**や **PLS** などの次元削減手法を用いて圧縮後に識別機に入力する．

[36] でも同様に混合ガウス分布を用いて BoF の改良を行っている．[35] との違いは，はじめに全てのカテゴリを包括する一般的なコードブックを作成し，この一般

的なコードブックを元にカテゴリに特化したコードブックを生成させる点にある．

全てのカテゴリを包括する一般的なコードブックは全てのデータを用いて最尤推定により求める．つまり，式(3)〜式(7)を利用する．次に，全てのデータを用いて最尤推定されたパラメータを事前知識として，カテゴリ毎の訓練データからカテゴリに特化したコードブックを作成する．混合ガウス分布の各コンポーネントの共分散行列を対角行列と仮定し（$\boldsymbol{\sigma}_k^2 = \mathrm{diag}(\boldsymbol{\Sigma}_k)$），最尤推定で得られた平均と分散をそれぞれ $\boldsymbol{\mu}_k^{ml}$，$\boldsymbol{\sigma}_k^{ml}$ とする．また対象とする訓練データをカテゴリ $c$ に属するデータとする $\mathcal{X}_c = \{\boldsymbol{x}_n\}_{n=1}^{N_c}$．この学習方法を次に示す．

**E-step**

$$\gamma_n(k) = p(k|\boldsymbol{x}_n, \boldsymbol{\theta}^c) \tag{16}$$

**M-step**

$$\begin{aligned}
\pi_k^c &= \frac{\sum_{n=1}^{N_c}\gamma_n(k) + \tau_k^\pi}{N_c + \sum_{k=1}^{K}\tau_k^\pi}, \\
\boldsymbol{\mu}_k^c &= \frac{\sum_{n=1}^{N_c}\gamma_n(k)\boldsymbol{x}_n + \tau_k^m\boldsymbol{\mu}_k^{ml}}{\sum_{n=1}^{N_c}\gamma_n(k) + \tau_k^m}, \\
(\boldsymbol{\sigma}_k^c)^2 &= \frac{\sum_{n=1}^{N_c}\gamma_n(k)\boldsymbol{x}_n^2 + \tau_k^s((\boldsymbol{\sigma}_k^{ml})^2 + (\boldsymbol{\mu}_k^{ml})^2)}{\sum_{n=1}^{N_c}\gamma_n(k) + \tau_k^s} - (\boldsymbol{\mu}_k^c)^2.
\end{aligned}$$

ここで，$\boldsymbol{x}^2$ は $\mathrm{diag}(\boldsymbol{x}\boldsymbol{x}^\top)$ の略記であり，$\tau_k^w, \tau_k^m, \tau_k^s$ は，事前に与えた情報とカテゴリに特化したデータから獲得されるエビデンスの間のバランスを取るパラメータである．

このようにカテゴリに依存しない一般的なコードブックとカテゴリに特化したコードブックが得られたとする．それぞれのコードブックから生成される特徴ベクトルは次のようになる．

$$\begin{aligned}
\boldsymbol{f}^u &= \frac{1}{N}\sum_{n=1}^{N}[p(1|\boldsymbol{x}_n, \boldsymbol{\theta}^{ml}), \cdots, p(K|\boldsymbol{x}_n, \boldsymbol{\theta}^{ml})]^\top \in R^K \\
\boldsymbol{f}^c &= \frac{1}{N}\sum_{n=1}^{N}[p(1|\boldsymbol{x}_n, \boldsymbol{\theta}^{c}), \cdots, p(K|\boldsymbol{x}_n, \boldsymbol{\theta}^{c})]^\top \in R^K
\end{aligned}$$

識別機は各カテゴリとその他を識別する**一対他（one-vs-all）**識別機によって構成される．各カテゴリ識別機にはそれぞれ別の特徴量が利用される．利用される特徴量はカテゴリに依存しない特徴とカテゴリに特化した特徴を結合したものである．

$$\boldsymbol{f}(c) = [\boldsymbol{f}^{u\top}, \boldsymbol{f}^{c\top}]^\top \in R^{2K} \tag{17}$$

これは式(15)と比較してコンパクトな表現となっている．

### 5.3. フィッシャーベクトル

局所特徴の混合ガウス分布を用いた正確な確率密度分布推定による BoF の改良を述べた．混合ガウス分布は**生成モデル（generative model）**と見なせるが，生成モデルを識別的なアプローチに適応可能なより洗練された手法があれば識別性能の改善につながるはずである．**フィッシャーカーネル（Fisher Kernel）**[81] は**生成的アプローチ（generative approach）**と**識別的アプローチ（discriminative approach）**を結合させる強力な枠組みである．フィッシャーカーネルでは，まず局所特徴を生成する確率密度分布から導出される勾配ベクトルを計算し，画像を表現する一つの特徴ベクトルとする．そしてこの特徴ベクトルを識別的分類機に入力する．

BoF と比較してフィッシャーカーネルを利用するメリットは，コードブックサイズが同じであればフィッシャーカーネルの方がより要素数の多い特徴ベクトルが得られる点にある．つまり，特徴ベクトルの表現する情報が多いため計算コストの高いカーネル法を利用して高次元空間へ射影する必要がなく，線形識別機でも十分な識別性能を出すことが可能となる．

ここで，$u_\theta$ をあらゆる画像の内容を表現する**確率密度関数（probability density function, pdf）**とし $\boldsymbol{\theta}$ を確率密度関数のパラメータとする．局所特徴群を $\mathcal{X}$ とすると，このデータを次に示す勾配ベクトルで表現する．

$$G_\theta^{\mathcal{X}} = \frac{1}{N}\nabla_\theta \log u_\theta(\mathcal{X}|\boldsymbol{\theta}) \tag{18}$$

対数尤度の勾配はデータに最も適合するように確率密度関数のパラメータが修正すべき方向を表現している．また異なるデータサイズの $\mathcal{X}$ をパラメータ数に依存した決まった長さの特徴ベクトルに変換する．

この勾配ベクトルは様々な識別機に利用できるが，内積を利用する識別機ではベクトルを正規化する必要がある．この正規化には**フィッシャー情報行列（Fisher information matrix）**が利用できる．

$$F_\theta = E_X[\nabla_\theta \log u_\theta(\mathcal{X}|\boldsymbol{\theta})\nabla_\theta \log u_\theta(\mathcal{X}|\boldsymbol{\theta})^\top]. \tag{19}$$

フィッシャー情報行列を用いて正規化された勾配ベクトルは次のように与えられる．

$$\mathcal{G}_\theta^X = F_\theta^{-1/2}\nabla_\theta \log u_\theta(\mathcal{X}|\boldsymbol{\theta}) \tag{20}$$

このようにしてできた画像の特徴ベクトルを局所特徴群 $\mathcal{X}$ の**フィッシャーベクトル（Fisher Vector）**と呼ぶ[39]．計算コストの観点からフィッシャー情報行列を単位行列と近似する場合もあるが，[39] では対角行列として近似している．

フィッシャーカーネルをコードブックに適応するにあたり，局所特徴の特徴空間における確率密度分布を混合ガウス分布で表現する．一枚の画像から得られる局所特徴の集合を $\mathcal{X}$ とする．$\gamma_n(k)$ を式(3)で示した局所特徴 $\boldsymbol{x}_n$ が $k$ 番目のコンポーネントから生成される確率とす

る．この時，対数尤度 $\mathcal{L}(\mathcal{X}|\boldsymbol{\theta}) = \log u_\theta(\mathcal{X}|\boldsymbol{\theta})$ の微分は以下のようになる．

$$\frac{\partial \mathcal{L}(\mathcal{X}|\boldsymbol{\theta})}{\partial \pi_k} = \sum_{n=1}^{N}\left[\frac{\gamma_n(k)}{\pi_k} - \frac{\gamma_n(1)}{\pi_1}\right] \quad (21)$$

$$\frac{\partial \mathcal{L}(\mathcal{X}|\boldsymbol{\theta})}{\partial \boldsymbol{\mu}_k^d} = \sum_{n=1}^{N}\gamma_n(k)\left[\frac{\boldsymbol{x}_n^d - \boldsymbol{\mu}_k^d}{(\boldsymbol{\sigma}_k^d)^2}\right] \quad (22)$$

$$\frac{\partial \mathcal{L}(\mathcal{X}|\boldsymbol{\theta})}{\partial \boldsymbol{\sigma}_k^d} = \sum_{n=1}^{N}\gamma_n(k)\left[\frac{(\boldsymbol{x}_n^d - \boldsymbol{\mu}_k^d)^2}{(\boldsymbol{\sigma}_k^d)^3} - \frac{1}{\boldsymbol{\sigma}_k^d}\right] \quad (23)$$

ここで，ベクトルの上付き文字 $d$ はベクトルの $d$ 番目の要素を示す．また混合ガウス分布の共分散行列は対角行列（$\boldsymbol{\sigma}_k^2 = \mathrm{diag}(\boldsymbol{\Sigma}_k)$）と仮定している．

フィッシャー情報行列を対角行列と仮定し，$\frac{\partial \mathcal{L}(\mathcal{X}|\theta)}{\partial \pi_k}$，$\frac{\partial \mathcal{L}(\mathcal{X}|\theta)}{\partial \mu_k^d}$，$\frac{\partial \mathcal{L}(\mathcal{X}|\theta)}{\partial \sigma_k^d}$ のそれぞれに対応するフィッシャー情報行列の要素を $f_{\pi_k}$，$f_{\mu_k^d}$，$f_{\sigma_k^d}$ とすると，これらは次に示すように閉じた解として近似的に求められる．

$$f_{\pi_k} = N\left(\frac{1}{\pi_k} + \frac{1}{\pi_1}\right) \quad (24)$$

$$f_{\mu_k^d} = \frac{N\pi_k}{(\boldsymbol{\sigma}_k^d)^2} \quad (25)$$

$$f_{\sigma_k^d} = \frac{2N\pi_k}{(\boldsymbol{\sigma}_k^d)^2} \quad (26)$$

式 (14) に示したように，BoF や混合ガウス分布による特徴ベクトルは負担率を用いて

$$\boldsymbol{f} = \frac{1}{N}\sum_{n=1}^{N}[\gamma_n(1), \cdots, \gamma_n(K)]^\top \in R^K \quad (27)$$

と表現できるが，これは混合比が一定の仮定を設けると式 (21) に示すフィッシャーベクトルの 0 次の統計量と同じとなる．一方フィッシャーベクトルは 0 次だけではなく平均（1 次），分散（2 次）の統計量を考慮している．コードブックのサイズを $K$ とすると，BoF は $K$ 次元のベクトルとなるがフィッシャーベクトルは $(2d+1)K-1$ 次元となる．つまりフィッシャーベクトルは小さなコードブックサイズで豊かな表現が可能となる．

### 5.4. フィッシャーベクトルの改良

フィッシャーベクトルは BoF と比較して画像を豊かに表現しているにも関わらず，そのまま画像識別に利用しても BoF とさほど性能に差がない．そこで [82] ではフィッシャーベクトルの改良を提案している．

ここで一枚の画像から得られた局所特徴群 $\mathcal{X} = \{\boldsymbol{x}_n\}_{n=1}^N$ は確率密度分布を $p(\boldsymbol{x})$ に従っているとする．十分大きな $N$ のとき大数の法則から式 (18) は以下のように近似できる．

$$G_\theta^{\mathcal{X}} \approx \nabla_\theta \int_x p(\boldsymbol{x}) \log u_\theta(\boldsymbol{x}) d\boldsymbol{x} \quad (28)$$

確率密度分布 $p(\boldsymbol{x})$ を画像に依存しない分布 $u_\theta(\boldsymbol{x})$ と画像に特定の分布 $q(\boldsymbol{x})$ に分解する．

$$p(\boldsymbol{x}) = \omega q(\boldsymbol{x}) + (1-\omega)u_\theta(\boldsymbol{x}) \quad (29)$$

ここで $\omega$ は 0 から 1 の間の値を取るパラメータである．式 (29) を式 (28) に代入する．

$$G_\theta^{\mathcal{X}} \approx \omega\nabla_\theta \int_x q(\boldsymbol{x}) \log u_\theta(\boldsymbol{x}) d\boldsymbol{x} + (1-\omega)\nabla_\theta \int_x u_\theta(\boldsymbol{x}) \log u_\theta(\boldsymbol{x}) d\boldsymbol{x} \quad (30)$$

パラメータ $\boldsymbol{\theta}$ は最尤法によって求められているとすると式 (30) の右辺第二項はゼロと見なせるので，結局，

$$G_\theta^{\mathcal{X}} \approx \omega\nabla_\theta \int_x q(\boldsymbol{x}) \log u_\theta(\boldsymbol{x}) d\boldsymbol{x} \quad (31)$$

となるため，フィッシャーベクトルを利用すると画像に依存しない部分を無視することができる．しかしながら $\omega$ の値の大小により $G_\theta^{\mathcal{X}}$ の値が変化する．これは前景と背景の割合によって $G_\theta^{\mathcal{X}}$ の値が変化することを意味する．そのために $\omega$ の影響を取り除く必要があるが，[82] では $G_\theta^{\mathcal{X}}$ の **L2 正規化（L2 normalization）** によって $\omega$ の影響を排除する手法を提案している．

また，混合ガウス分布の混合数 $K$ を増加させるとフィッシャーベクトルがスパースになる現象が観測されている．これは混合数の増加により局所特徴が複数のコードワードと接近するため，大きな負担率 $\gamma_n(k)$ を持つ局所特徴が少なくなることによる．その結果，フィッシャーベクトルの要素はゼロ近くの頻度が高くなる．L2 正規化により得られたベクトルの内積は L2 距離と同じであるが，スパースなベクトルに L2 距離を適用しても高い識別性能が得られないことが知られている [45]．このときの対処方法として，カーネル法の利用が考えられるが，一般にカーネル法は計算コストが高く大規模データへの適応は難しい．[82] では**パワー正規化（power normalization）** によりスパースネスを緩和することで内積による類似度を維持する手法を提案している．具体的にはフィッシャーベクトルの各要素 $z$ に以下に示す関数を適用する．

$$f(z) = \mathrm{sign}(z)|z|^\alpha \quad (32)$$

ここで $\alpha$ は正規化のためのパラメータである．正規化の順序はパワー正規化を行った後に L2 正規化を適用している．

三番目のフィッシャーベクトルの改良点として**空間ピラミッド（spatial pyramid）** [83] の適応を行っている．空間ピラミッドは画像を一定間隔のグリッドに分割し，分割されたセル内で BoF のヒストグラムを求める．分割の粒度（レベル）を変えて得られた全てのセルのヒストグラムをつなげて一つの特徴ベクトルとする手法である．[82] では画像を $1\times1$，$2\times2$，$3\times1$ の合計 8 個の

セルに分割し，8 個のフィッシャーベクトルを計算している．

標準的な画像識別用のデータセットを利用した実験結果より，パワー正規化，L2 正規化，空間ピラミッドの順に性能改善の効果が高いことが示されている [82].

### 5.5. VLAD

フィッシャーベクトルは画像検索にも活用され高い性能を示すことが知られている．ここでは [49] の手法を紹介する．[49] ではフィッシャーベクトルを簡略化した特徴として **VLAD (Vector of Locally Aggregated Descriptors)** を提案している．

ここで，コードブック $\mathcal{V} = \{\boldsymbol{v}_k\}_{k=1}^{K}$ が得られているとする．また局所特徴 $\boldsymbol{x} \in R^D$ は最近傍のコードワードに割り当てられるとする（$\boldsymbol{v}_i = \mathrm{NN}(\boldsymbol{x})$）．$\boldsymbol{v}_i$ に割り当てられた局所特徴の集合を $\mathcal{X}_i$ とする．VLAD（$\boldsymbol{z} \in R^{KD}$）は各コードワード $\boldsymbol{v}_i$ に対して，$\boldsymbol{v}_i$ に割り当てられた局所特徴 $\boldsymbol{x}$ との差 $\boldsymbol{x} - \boldsymbol{v}_i$ の和を計算する．

$$\boldsymbol{z}_i^d = \sum_{x \in \mathcal{X}_i} (\boldsymbol{x}^d - \boldsymbol{v}_i^d) \tag{33}$$

この後に $\boldsymbol{z}$ に対し L2 正規化を適応する（$\boldsymbol{z} = \boldsymbol{z}/||\boldsymbol{z}||$）．ここでフィッシャーベクトルの式 (22) と比較すると負担率 $\gamma_n(k)$ を最近傍のコードワード $\boldsymbol{v}_k$ に対して 1，その他を 0 とし，分散 $\boldsymbol{\sigma}_k^d$ を混合ガウス分布の全コンポーネントで一定と見なせば，両式は一致することが分かる．

[49] ではさらに，VLAD を主成分分析で次元削減を行った後，削減後のベクトルの各要素の分散のバランスを取り，**プロダクト量子化（product quantization）**[84] によって VLAD をビットコードへ変換している．クエリーベクトル（画像）が与えられると量子化されたデータベース内の画像と量子化しないクエリーベクトルの近傍探索（**ADC（asymmetric distance computation）**）を行うことで検索を実行する．これらの詳細は本稿の趣旨からやや外れるために割愛する．

### 5.6. スパース符号化

ここで，BoF のアプローチはいわゆる**ベクトル量子化（vector quantization, VQ）**であるが，$K$ 近傍法によるプロセスを最適化問題として定式化すると以下のように記述できる．

$$\begin{aligned} &\min_{U,V} \sum_{n=1}^{N} ||\boldsymbol{x}_n - \boldsymbol{V}\boldsymbol{u}_n||^2 \\ &\text{s.t. } \mathrm{Card}(\boldsymbol{u}_n) = 1, |\boldsymbol{u}_n| = 1, \boldsymbol{u}_n \succeq 0, \forall n \end{aligned} \tag{34}$$

ここで $X = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N]^\top \in R^{N \times D}$ は $D$ 次元局所特徴の集合，$V = [\boldsymbol{v}_1, \cdots, \boldsymbol{v}_K] \in R^{D \times K}$ はコードブック，$\boldsymbol{v}_k$ はコードワード，$\boldsymbol{U} = [\boldsymbol{u}_1, \cdots, \boldsymbol{u}_N]^\top \in R^{N \times K}$ は局所特徴がどのコードワードに所属するかを示す指標である．また，$\mathrm{Card}(\boldsymbol{u}_n) = 1$ は $\boldsymbol{u}_n$ の一つの要素のみが非ゼロであり，$|\boldsymbol{u}_n| = 1$ は $\boldsymbol{u}_n$ の L1 ノルムが 1，$\boldsymbol{u}_n \succeq 0$ は $\boldsymbol{u}_n$ の全ての要素が非ゼロであることを示す．学習のフェーズでは式 (34) において $\boldsymbol{U}, \boldsymbol{V}$ に関して最適化を行い，符号化のフェーズでは学習で得られたコードブック $\boldsymbol{V}$ を用いて，$\boldsymbol{U}$ のみに関して式 (34) を解く．

ベクトル量子化における $\mathrm{Card}(\boldsymbol{u}_n) = 1$ という制約は非常に厳しく，一つのコードワードのみで局所特徴を表現することになるため，局所特徴の粗い近似となる．コードワードを用いてより正確に局所特徴を表現するために [50] では $\boldsymbol{u}_n$ に **L1 正則化（L1 normalization）**の制約を与え，$\boldsymbol{u}_n$ の少数の要素が非ゼロとなるようにする手法を提案している．これは局所特徴が少数のコードワードに帰属することを意味する．これによりベクトル量子化は以下に示すように**スパース符号化（sparse coding, SC）**の問題と捉えることができる．

$$\begin{aligned} &\min_{U,V} \sum_{n=1}^{N} ||\boldsymbol{x}_n - \boldsymbol{V}\boldsymbol{u}_n||^2 + \lambda|\boldsymbol{u}_n| \\ &\text{s.t. } ||\boldsymbol{v}_k|| \leq 1, \forall k \end{aligned} \tag{35}$$

ベクトル量子化のプロセスと同様に，学習のフェーズでは式 (35) において $\boldsymbol{U}, \boldsymbol{V}$ に関して最適化を行い，符号化のフェーズでは学習で得られたコードブック $\boldsymbol{V}$ を用いて，$\boldsymbol{U}$ のみに関して式 (35) を解く．ここでの L1 正則化の重要な役割をまとめると次のようになる．

- コードブックは局所特徴の次元数よりも多く，過剰（$K > D$）なため，under determined な系である．つまり情報が不足して解を定められない状況にある．そのため L1 正則化により解を定めることが可能となる．
- スパース性の事前知識を用いることによって局所特徴の顕著なパターンを捉えることができる．
- ベクトル量子化よりもスパース符号化の方が量子化誤差を低減させられる．

符号化された局所特徴群 $\boldsymbol{U}$ から一つの特徴ベクトル $\boldsymbol{f}$ を得る手段として空間ピラミッドがよく用いられるが，これはあらかじめ定められた**プーリング関数（pooling function）**$\mathcal{F}$ によって計算される．

$$\boldsymbol{f} = \mathcal{F}(\boldsymbol{U}) \tag{36}$$

BoF の場合，$\mathcal{F}$ には平均を利用する場合が多い（$\boldsymbol{f} = \frac{1}{N}\sum_{n=1}^{N} \boldsymbol{u}_n$）．[50] ではプーリング関数 $\mathcal{F}$ として max プーリング関数を利用する手法を提案している．

$$\boldsymbol{f}^d = \max\{|\boldsymbol{u}_1^d|, |\boldsymbol{u}_2^d|, \cdots, |\boldsymbol{u}_N^d|\} \tag{37}$$

ここで $\boldsymbol{f}^d$ は $\boldsymbol{f}$ の $d$ 番目の要素である．max プーリング関数は他のプーリング関数と比べても高い性能を示すことが実験的に示されている [50, 85]．スパース符号化と空間ピラミッドマッチングを組み合わせた手法を **ScSPM** と呼ぶ．

### 5.7. 局所制約線形符号化

[86] では，データ $\boldsymbol{x}$ をそのデータ近傍に存在するいくつかの基準点の線形和で局所的に近似する手法を提案している．この結果，得られた線形和の重みはデータ $\boldsymbol{x}$ の**局所座標符号化（Local Coordinate Coding, LCC）**と呼ばれる．この中で，ある仮定の下では局所性がスパースネスよりも本質であると述べている．しかしながらスパース符号化と同じように，LCC も L1 ノルム最適化問題を解く必要があり，計算コストが高い問題を抱える．そこで [87] では，LCC の高速な実装と見なせる**局所制約線形符号化（Locality-constrained Linear Coding, LLC）**を提案している．

局所制約線形符号化では，式 (35) のスパース性の制約を用いる代わりに局所制約を用いる．具体的には，コードブック $\boldsymbol{V}$ が与えられている仮定の下で，局所制約線形符号化は符号化の過程に下記の基準を用いる．

$$\min_{U} \sum_{n=1}^{N} ||\boldsymbol{x}_n - \boldsymbol{V}\boldsymbol{u}_n||^2 + \lambda||\boldsymbol{d}_n \odot \boldsymbol{u}_n||^2 \quad (38)$$
$$\text{s.t. } \mathbf{1}^\top \boldsymbol{u}_n = 1, \forall n$$

ここで $\odot$ は要素毎の積であり，$\boldsymbol{d}_n \in R^K$ は入力ベクトル $\boldsymbol{x}_n$ と各コードワードとの距離に応じて，入力ベクトル $\boldsymbol{x}_n$ から遠いコードワードにペナルティを与えるベクトルである．

$$\boldsymbol{d}_n = \exp\left(\frac{\text{dist}(\boldsymbol{x}_n, \boldsymbol{V})}{\sigma}\right) \quad (39)$$

ここで $\text{dist}(\boldsymbol{x}_n, \boldsymbol{V}) = [\text{dist}(\boldsymbol{x}_n, \boldsymbol{v}_1), \cdots, \text{dist}(\boldsymbol{x}_n, \boldsymbol{v}_K)]^\top$, $\text{dist}(\boldsymbol{x}_n, \boldsymbol{v}_j)$ は入力ベクトル $\boldsymbol{x}_n$ とコードワード $\boldsymbol{v}_j$ とのユークリッド距離，$\sigma$ は荷重減衰速度の調整に利用される．また，$\text{dist}(\boldsymbol{x}_n, \boldsymbol{V})$ から $\max(\text{dist}(\boldsymbol{x}_n, \boldsymbol{V}))$ を差し引くことで $\boldsymbol{d}_n$ の値を $(0, 1]$ に正規化する．式 (38) は L0 ノルムの意味ではスパースではないが $\boldsymbol{u}_n$ の少数の要素のみ高い値をとるという意味でスパースである．

良い識別性能を得るには類似した局所記述子には類似した符号を出力する符号化手法である必要がある．この観点から局所制約線形符号化は次の特性を持つ．

- ベクトル量子化では局所特徴を一つのコードワードに割り当てる．そのために量子化誤差が大きく類似した局所記述子であっても量子化後の符号は異なる可能性がある．またベクトル量子化はコードワード間の関係を無視している．一方，局所制約線形符号化は複数のコードワードを利用しているために，より正確に局所記述子を表現できる．また共通したコードワードによって類似した局所記述子の関係を捉えられる．
- スパース符号化ではコードワードが過剰であるためにスパース性を優先することで類似した局所記述子に対して全く異なるコードワードを選択する可能性がある．一方，局所制約線形符号化は類似した局所記述子には類似したコードを出力可能である．
- スパース符号化は最適化計算を必要とするが局所制約線形符号化には解析解が存在する．

局所制約線形符号化を近似的することで高速に符号化が可能となる．式 (38) を解くのではなく，入力 $\boldsymbol{x}_n$ の $K$ 近傍のコードワードを局所基底 $\boldsymbol{V}_n$ として採用し，より小さな線形システムを解くことで符号を得る．

$$\min_{\tilde{U}} \sum_{n=1}^{N} ||\boldsymbol{x}_n - \boldsymbol{V}_n\tilde{\boldsymbol{u}}_n||^2 \quad (40)$$
$$\text{s.t. } \mathbf{1}^\top \tilde{\boldsymbol{u}}_n = 1, \forall n$$

**局所線形埋込み（Local Linear Embedding, LLE）**と比較して，局所制約線形符号化はコードブックの学習が入る点で異なる．コードブックの学習は，以下の基準を逐次的に計算することで行う．

$$\arg\min_{U,V} \sum_{n=1}^{N} ||\boldsymbol{x}_n - \boldsymbol{V}\boldsymbol{u}_n||^2 + \lambda||\boldsymbol{d}_n \odot \boldsymbol{u}_n||^2 \quad (41)$$
$$\text{s.t. } \mathbf{1}^\top \boldsymbol{u}_n = 1, \forall n,\ ||\boldsymbol{v}_j||^2 \leq 1, \forall j$$

### 5.8. スーパーベクトル符号化

BoF や混合ガウス分布を用いた BoF の改善手法は，特徴空間における局所特徴の分布の表現を得るプロセスと解釈できた．同様にここでも高次元空間におけるなめらかな非線形関数 $f(\boldsymbol{x})$ の学習についてとりあげ，$f(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x})$ と線形関数でよく近似できる符号化手法 $\boldsymbol{\phi}(\boldsymbol{x})$ の導出問題を考える [88].

局所記述子 $\boldsymbol{x}$ をコードブック $\mathcal{V} = \{\boldsymbol{v}_k\}_{k=1}^K$ を用いて近似する．

$$\boldsymbol{x} \approx \sum_{k=1}^{K} \gamma_x(k)\boldsymbol{v}_k, \boldsymbol{\gamma}_x = [\gamma_x(1), \cdots, \gamma_x(K)], \sum_{k=1}^{K} \gamma_x(k) = 1$$

$\boldsymbol{\gamma}_x$ の非ゼロ要素の数を 1 とするとベクトル量子化となる．このとき得られる $\boldsymbol{x}$ の最近傍コードワードを $\boldsymbol{v}^x$ とする．

全ての $\boldsymbol{x}, \boldsymbol{x}' \in R^D$ に対して次の条件を満たすとき，$f(\boldsymbol{x})$ は $\beta$ Lipschitz derivative smooth と呼ぶ．

$$|f(\boldsymbol{x}) - f(\boldsymbol{x}') - \nabla f(\boldsymbol{x}')^\top(\boldsymbol{x} - \boldsymbol{x}')| \leq \frac{\beta}{2}||\boldsymbol{x} - \boldsymbol{x}'||^2 \quad (42)$$

ここで $\boldsymbol{x}' = \boldsymbol{v}^x$ とすると次式を得る．

$$|f(\boldsymbol{x}) - f(\boldsymbol{v}^x) - \nabla f(\boldsymbol{v}^x)^\top(\boldsymbol{x} - \boldsymbol{v}^x)| \leq \frac{\beta}{2}||\boldsymbol{x} - \boldsymbol{v}^x||^2 \quad (43)$$

この式から，$f(\boldsymbol{x}) = f(\boldsymbol{v}^x) + \nabla f(\boldsymbol{v}^x)^\top(\boldsymbol{x} - \boldsymbol{v}^x)$ と近似可能なことや，上限を最小化するコードブックを学習することで $f(\boldsymbol{x})$ の近似を改善できることが分かる．

さらに $f(\boldsymbol{x})$ は次に示すように線形関数で近似可能である．

$$f(\boldsymbol{x}) \approx \boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}) \quad (44)$$

ここで $\phi(\boldsymbol{x})$ を $\boldsymbol{x}$ の**スーパーベクトル符号化（Super-Vector coding）**[88] と呼ぶ．

$$\phi(\boldsymbol{x}) = \left[s\gamma_x(k), \gamma_x(k)(\boldsymbol{x}-\boldsymbol{v}_k)^\top\right]^\top_{v_k\in\mathcal{V}} \quad (45)$$

$$\boldsymbol{w} = \left[\frac{1}{s}f(\boldsymbol{v}_k), (\nabla f(\boldsymbol{v}_k))^\top\right]^\top_{v_k\in\mathcal{V}} \quad (46)$$

ここで $s$ は非負の定数，$\boldsymbol{w}$ は推定される未知のベクトルである．

スーパーベクトル符号化で得られる $\boldsymbol{w}^\top\phi(\boldsymbol{x})$ によって，非線形関数 $f(\boldsymbol{x})$ を区分的線形関数で近似することができる．一方ベクトル量子化は，$\phi(\boldsymbol{x}) = [\gamma_x(k)]^\top_{v_k\in\mathcal{V}}$ となっているので，非線形関数 $f(\boldsymbol{x})$ を区分的に定数をとる関数で近似していることになる．従って，スーパーベクトル符号化はベクトル量子化と比較して低い関数近似誤差を得る．また，フィッシャーベクトルの負担率と平均を示す式 (21) と式 (22) を比較すると，スーパーベクトル符号化と類似していることが分かる．

## 6. おわりに

本稿では大規模画像データセットを用いた自動画像アノテーションについて概観し，スケーラビリティが特に重要視されることを述べた．画像アノテーションのプロセスをデータセット構築，特徴抽出，モデル化に分類した．また近年盛んに研究されている画像表現手法について詳しく述べた．直列に処理を重ねていくことで各処理後のデータに含まれる情報量は決して増加しないため，性能向上のためにはデータセット，特徴抽出，モデルの順で高い質が必要となる．そのためデータ量と質で画像アノテーション性能が大きく左右されるが，その規模は一研究者だけでは扱いきれないレベルに到達しつつある．それではデータを握る一部のグループ以外研究できないかというとそうとは思えない．一例をあげると，Web に存在する画像は人間が意識的にアップロードしたものであり，人間という高度なフィルタを通している．つまり Web 上の画像認識と真の実世界画像認識とは依然として大きな隔たりがある．この人間が持つようなフィルタ（特徴抽出）構築は根本的問題であり，ブレークスルーを期待したい．

## References


[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とイメージメディア，vol.48，pp.1–24，2007． 1

[2] 黄瀬浩一，“局所特徴量を用いた画像照合による特定物体認識，” 人工知能学会誌，vol.25，no.6，pp.769–776，2010． 1

[3] P. Duygulu, K. Barnard, and D.F.N. Freitas, “Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary,” ECCV, 2002. 1

[4] M. Grubinger, P. Clough, H. Müller, and T. Deselaers, “The iapr benchmark: A new evaluation resource for visual information systems,” International Conference on Language Resources and Evaluation, 2006. 1

[5] L. vonAhn and L. Dabbish, “Labeling images with a computer game,” SIGCHI, 2004. 1

[6] A. Makadia, V. Pavlovic, and S. Kumar, “A new baseline for image annotation,” ECCV, 2008. 2, 4

[7] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid, “Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation,” ICCV, 2009. 2, 4

[8] 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,” CVPR 2004, Workshop on Generative-Model Based Vision, 2004. 2

[9] G. Griffin, A. Holub, and P. Perona, “Caltech-256 object category dataset,” Technical report, California Institute of Technology, 2007. 2

[10] S. Lazebnik, C. Schmid, and J. Ponce, “Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories,” CVPR, 2006. 2, 3

[11] Y.-Y. Lin, J.-F. Tsai, and T.-L. Liu, “Efficient discriminative local learning for object recognition,” ICCV, 2009. 2, 5

[12] A. Torralba, R. Fergus, and W.T. Freeman, “80 million tiny images: A large data set for nonparametric object and scene recognition,” IEEE PAMI, vol.30, no.11, pp.1958–1970, 2008. 2, 5

[13] G.A. Miller, “Wordnet: A lexical database for english,” Communications of the ACM, vol.38, no.11, pp.39–41, 1995. 2

[14] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “Imagenet: A large-scale hierarchical image database,” CVPR, 2009. 2, 5

[15] X.-J. Wang, L. Zhang, M. Liu, Y. Li, and W.-Y. Ma, “Arista - image search to annotation on billions of web photos,” CVPR, 2010. 2

[16] J. Xiao, J. Haysy, K.A. Ehinger, A. Oliva, and A. Torralba, “Sun database: Large-scale scene recognition from abbey to zoo,” CVPR, 2010. 2

[17] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, “Content-based image retrieval at the end of the early years,” IEEE PAMI, vol.22, no.12, pp.1349–1380, 2000. 2

[18] Y. taoZheng, M. Zhao, Y. Song, H. Adam, U. Buddemeier, A. Bissacco, F. Brucher, T.-S. Chua, and H. Neven, "Tour the world: Building a web-scale landmark recognition engine," CVPR, 2009. 2, 6

[19] Y. Li, D.J. Crandall, and D.P. Huttenlocher, "Landmark classification in large-scale image collections," ICCV, 2009. 2, 6

[20] S. Gammeter, L. Bossard, T. Quack, and L.V. Gool, "I know what you did last summer: object-level auto-annotation of holiday snaps," ICCV, 2009. 2, 3, 6

[21] S. Zanetti, L. Zelnik-Manor, and P. Perona, "A walk through the web 's video clips," CVPR Workshop on Internet Vision, 2008. 3

[22] Z. Wang, M. Zhao, Y. Song, S. Kumar, and B. Li, "Youtubecat: Learning to categorizewildweb videos," CVPR, 2010. 3, 6

[23] G. Toderici, H. Aradhye, M. Pasca, L. Sbaiz, and J. Yagnik, "Finding meaning on youtube: Tag recommendation and category discovery," CVPR, 2010. 3, 6

[24] F. Schroff, A. Criminisi, and A. Zisserman, "Harvesting image databases from the web," ICCV, 2007. 3

[25] J. Fan, Y. Shen, N. Zhou, and Y. Gao, "Harvesting large-scale weakly-tagged image databases from theweb," CVPR, 2010. 3

[26] D.G. Lowe, "Distinctive image features from scale-invariant keypoints," IJCV, vol.60, no.2, pp.91–110, 2004. 3

[27] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," CVPR, 2005. 3

[28] H. Bay, A. Ess, T. Tuytelaars, and L.V. Gool, "Surf: Speeded up robust features," CVIU, vol.110, no.3, pp.346–359, 2008. 3

[29] E. Shechtman and M. Irani, "Matching local self-similarities across images and videos," CVPR, 2007. 3

[30] A.C. Berg, T.L. Berg, and J. Malik, "Shape matching and object recognition using low distortion correspondence," CVPR, 2005. 3

[31] N. Otsu and T. Kurita, "A new scheme for practical, flexible and intelligent vision systems," IAPR Workshop on Computer Vision, 1988. 3

[32] A. Oliva and A. Torralba, "Modeling the shape of the scene: a holistic representation of the spatial envelope," IJCV, vol.42, no.3, pp.145–175, 2001. 3

[33] 藤吉弘亘, "画像局所特徴量 sift と最近のアプローチ," 人工知能学会誌, vol.25, pp.753–760, 2010. 3

[34] G. Csurka, C.R. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," ECCV International Workshop on Statistical Learning in Computer Vision, 2004. 3, 7

[35] J.D.R. Farquhar, S. Szedmak, H. Meng, and J. Shawe-Taylor, "Improving "bag-of-keypoints" image categorisation: Generative models and pdf-kernels," Technical report, University of Southampton, 2005. 3, 7, 8

[36] F. Perronnin, C. Dance, G. Csurka, and M. Bressan, "Adapted vocabularies for generic visual categorization," ECCV, 2006. 3, 8

[37] J.C. vanGemert, J.M. Geusebroek, C.J. Veenman, and A.W.M. Smeulders, "Kernel codebooks for scene categorization," ECCV, 2008. 3

[38] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Lost in quantization: Improving particular object retrieval in large scale image databases," CVPR, 2008. 3

[39] F. Perronnin and C. Dance, "Fisher kernels on visual vocabularies for image categorization," CVPR, 2007. 3, 9

[40] Y. Cao, C. Wang, Z. Li, L. Zhang, and L. Zhang, "Spatial-bag-of-features," CVPR, 2010. 4

[41] N. Morioka, Shin 'ichiSatoh, "Building compact local pairwise codebook with joint feature space clustering," ECCV, 2010. 4

[42] H. Cai, K. Mikolajczyk, and F. Yan, "Learning weights for codebook in image classification," CVPR, 2010. 4

[43] R. Ji, H. Yao, and X. Sun, "Towards semantic embedding in visual vocabulary," CVPR, 2010. 4

[44] M. Muja and D.G. Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration," VISAPP International Conference on Computer Vision Theory and Applications, 2009. 4

[45] D. Nistér and H. Stewénius, "Scalable recognition with a vocabulary tree," CVPR, 2006. 4, 10

[46] M. Datar, N. Immorlica, P. Indyk, and V.S. Mirrokni, "Locality-sensitive hashing scheme based on p-stable distributions," the 12th annual symposium on Computational geometry, 2004. 4

[47] Y. Jia, J. Wang, G. Zeng, and X.-S. Hua, "Optimizing kd-trees for scalable visual descriptor indexing," CVPR, 2010. 4

[48] R.F. Sproull, "Refinements to nearest-neighbor searching in k-dimensional trees," Algorithmica, vol.6, no.4, pp.579–589, 1991. 4

[49] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," CVPR, 2010. 4, 11

[50] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," CVPR, 2009. 4, 11

[51] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce, "Learning mid-level features for recognition," CVPR, 2010. 4

[52] S. Zhang, J. Huang, Y. Huang, Y. Yu, H. Li, and D.N. Metaxas, "Automatic image annotation using group sparsity," CVPR, 2010. 4

[53] M. Yuan, M. Yuan, Y. Lin, and Y. Lin, "Model selection and estimation in regression with grouped variables," Journal of the Royal Statistical Society, Series B, vol.68, pp.49–67, 2006. 4

[54] H. Nakayama, T. Harada, and Y. Kuniyoshi, "Global gaussian approach for scene categorization using information geometry," CVPR, 2010. 4

[55] T. Harada, H. Nakayama, and Y. Kuniyoshi, "Improving local descriptors by embedding global and local spatial information," ECCV, 2010. 4

[56] T. Harada, H. Nakayama, Y. Kuniyoshi, and N. Otsu, "Image annotation and retrieval for weakly labeled images using conceptual learning," New Generation Computing, vol.28, pp.277–298, 2010. 4, 5

[57] C. Cusano, G. Ciocca, and R. Schettini, "Image annotation using svm," Proceedings of the SPIE, vol.5304, pp.330–338, 2003. 4

[58] Y. Xiang, X. Zhou, Z. Liu, T.-S. Chua, and C.-W. Ngo, "Semantic context modeling with maximal margin conditional random fields for automatic image annotation," CVPR, 2010. 4

[59] D. Putthividhya, H.T. Attias, and S.S. Nagarajan, "Topic regression multi-modal latent dirichlet allocation for image annotation," CVPR, 2010. 5

[60] D.M. Blei and M.I. Jordan, "Modeling annotated data," ACM SIGIR, 2003. 5

[61] G.R.G. Lanckriet, N. Cristianini, L.E. Ghaoui, P. Bartlett, and M.I. Jordan, "Learning the kernel matrix with semi-definite programming," JMLR, vol.5, pp.27–72, 2004. 5

[62] A. Rakotomamonjy, F.R. Bach, S. Canu, and Y. Grandvalet, "Simplemkl," JMLR, vol.9, pp.2491–2521, 2008. 5

[63] O. Boiman, E. Shechtman, and M. Irani, "In defense of nearest-neighbor based image classification," CVPR, 2008. 5

[64] B. Collins, J. Deng, K. Li, and L. Fei-Fei, "Towards scalable dataset construction: An active learning approach," ECCV, 2008. 5

[65] H.-J. Zeng, Q.-C. He, Z. Chen, W.-Y. Ma, and J. Ma, "Learning to cluster web search results," ACM SIGIR, 2004. 5

[66] X.-J. Wang, L. Zhang, F. Jing, and W.-Y. Ma, "Annosearch: Image auto-annotation by search," CVPR, 2006. 5

[67] H. Nakayama, T. Harada, and Y. Kuniyoshi, "Canonical contextual distance for large-scale image annotation and retrieval," the First ACM workshop on Large-scale multimedia retrieval and mining, 2009. 5

[68] H. Nakayama, T. Harada, and Y. Kuniyoshi, "Evaluation of dimensionality reduction methods for image auto-annotation," BMVC, 2010. 5

[69] J. Weston, S. Bengio, and N. Usunier, "Large scale image annotation: learning to rank with joint word-image embeddings," Machine Learning, vol.81, pp.21–35, 2010. 5

[70] K. Crammer, Y. Singer, N. Cristianini, J. Shawe-taylor, and B. Williamson, "On the algorithmic implementation of multiclass kernel-based vector machines," JMLR, vol.2, pp.265–292, 2002. 6

[71] I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun, "Support vector machine learning for interdependent and structured output spaces," ICML, 2004. 6

[72] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," IJCV, vol.60, no.1, pp.63–86, 2004. 6

[73] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," CVPR, 2007. 6

[74] S. Gupta, J. Kim, K. Grauman, and R.J. Mooney, "Watch, listen & learn: Co-training on captioned images and videos," ECML PKDD, 2008. 6

[75] C.M. Christoudias, R. Urtasun, A. Kapoor, and T. Darrell, “Co-training with noisy perceptual observations,” CVPR, 2009. 6

[76] X. Zhu and Z. Ghahramani, “Learning from labeled and unlabeled data with label propagation,” Technical report, CMU-CALD-02-107, 2002. 6

[77] S. Kumar and M. Hebert, “Discriminative fields for modeling spatial dependencies in natural images,” NIPS, 2003. 6

[78] F. Perronnin, J. Sánchez, and Y. Liu, “Large-scale image categorization with explicit data embedding,” CVPR, 2010. 6

[79] J. Sivic and A. Zisserman, “Video Google: A text retrieval approach to object matching in videos,” ICCV, 2003. 7

[80] D. Ormoneit and V. Tresp, “Averaging, maximum penalized likelihood and bayesian estimation for improving gaussian mixture probability density estimates,” IEEE Trans. on Neural Networks, vol.9, no.4, pp.639–650, 1998. 8

[81] T. Jaakkola and D. Haussler, “Exploiting generative models in discriminative classifiers,” NIPS, 1998. 9

[82] F. Perronnin, J. Sánchez, and T. Mensink, “Improving the fisher kernel for large-scale image classification,” ECCV, 2010. 10, 11

[83] S. Lazebnik, C. Schmid, and J. Ponce, “Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories,” CVPR, 2006. 10

[84] H. Jegou, M. Douze, and C. Schmid, “Product quantization for nearest neighbor search,” IEEE PAMI, vol.33, pp.117–128, 2011. 11

[85] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce, “Learning mid-level features for recognition,” CVPR, 2010. 11

[86] K. Yu, T. Zhang, and Y. Gong, “Nonlinear learning using local coordinate coding,” NIPS, 2009. 12

[87] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, “Locality-constrained linear coding for image classification,” CVPR, 2010. 12

[88] X. Zhou, K. Yu, T. Zhang, and T.S. Huang, “Image classification using super-vector coding of local image descriptors,” ECCV, 2010. 12, 13