Vol.2019-CG-176 No.4
Vol.2019-DCC-23 No.4
Vol.2019-CVIM-219 No.4
2019/11/7

DCNNs の中間層を統合した画像検索用インデックスの生成法

谷洸明† 和田俊和†

概要：Deep Convolutional Neural Networks (DCNNs)は画像のクラス分類において大きな成功を収めている．クラス分類に用いられる DCNNs は識別のための画像の特徴を獲得しており，その特徴は画像検索にも利用できる．DCNNs による画像検索の多くは，最終層の１つ前の層の特徴ベクトルを画像検索用インデックスとして用いるが，最終層の近くで出力される特徴ベクトルは画像中の物体の意味的な特徴を多く含み，色やテクスチャ情報に関する特徴は捨象されている．したがって，最終層の１つ前の特徴ベクトルをインデックスとする検索では，色やテクスチャを重視した商品検索などの特定物体認識のタスクに対しては十分な精度が得られない．そこで色やテクスチャ情報を含む複数の中間層を含めて画像検索を行うことが必要になる．しかし，単に中間層から複数の特徴を取り出すだけでは特徴ベクトルのバランスが取れない為，中間層の特徴ベクトルを統合し，DNN による再学習した特徴ベクトルを用いる手法を提案する．この手法を Feature Tuning と呼ぶことにする．この手法を用いて色やテクスチャ情報を重視するデータセットに対して実験を行い，有効性を確認した．

キーワード：画像検索，深層学習，CNN 中間層，再学習

Index generation for image retrieval by integrating multiple layers of DCNNs

Hiroaki TANI† Toshikazu WADA†

Abstract: Deep Convolutional Neural Networks (DCNNs) have been successfully applied to image classification tasks. Features captured by DCNN tuned for classification can be used as indices for image retrieval. In many cases, indices are extracted from the feature map preceding the last output layer of DCNN. However, this feature map has semantic (categorical) information of object without color and texture information. Image retrieval using such feature is not suitable for specific object retrieval, such as product image retrieval, because color and texture information plays important role in such retrieval task. For solving this problem, we propose a method to create image retrieval indices from feature maps extracted from multiple layers of DCNNs. Since multiple features have imbalance information, we integrate them into an index tuned for image retrieval by using other DNN. We call this method Feature Turning. Through some specific object retrieval experiments, we demonstrate the effectiveness of our method.

Keywords: Image retrieval, Deep Convolutional Neural Networks, multi-layers, fine-tuning

1. はじめに

画像検索は，画像をクエリとして与えて，データベース内の類似する画像を検索するタスクである．画像検索は多くのアプリケーションに用いられる．その例として商品検索や顔認識，ランドマーク検索などがその一部である．画像検索はクラス分類問題のような一般物体認識とは異なり，検索対象が同じ物体かどうかを判定する特定物体認識に分類される．画像検索を行う際に用いられる画像検索用インデックスとは画像ごとに生成する特徴ベクトルのことで，そのベクトル同士の距離の近さによって検索対象が決まる．

画像検索に用いられる手法は大きく２つに分類される．1 つは，SIFT[1]などの局所特徴量から求めた Bag-of-Features(BoF)を用いたものである．もう１つは精度が良く，近年より利用されるようになった Deep Convolutional Neural Networks (DCNNs)を用いたものである．DCNNs は ImageNet[2]などで学習されたネットワークを用いるクラス分類問題や，COCO データセット[3]などを用いた物体検出[4]などで幅広く利用され，いくつもの成功を収めている．DCNNs を用いた画像検索の多くは，クラス分類で用いられるネットワークを学習した際に得られた画像特徴を検索用インデックスとして利用している．

画像検索では特徴空間にあるベクトル間の距離によって相違度を評価する．そのため，クラス分類を行う全結合層での内積計算に対するしきい値処理を前提とした画像特徴を学習しても，距離計算による検索は正しく行われない．これは同一クラスの特徴ベクトル間の内積を大きくし，異なるクラス間では内積の値が小さくなるように学習が進むため，特徴ベクトル間の距離の大小で正しい検索が行えなくなるためある．これを解決するためには，特徴ベクトルの L2 ノルムを一定にすればよい．これは，次式のようにベクトルx，yを超球上に射影した場合，ベクトル間の角度と超球上のベクトル間の大小関係が一致するためである．

$$\left\|\frac{x}{\|x\|}-\frac{y}{\|y\|}\right\|^2 = 2\left(1-\frac{x}{\|x\|}\cdot\frac{y}{\|y\|}\right) = 2(1-\cos\theta)$$

さらに，特徴ベクトル間のクラス内分散を小さくし，クラ

† 和歌山大学院システム工学研究科
Wakayama University, Faculty of System Engineering

Vol.2019-CG-176 No.4
Vol.2019-DCC-23 No.4
Vol.2019-CVIM-219 No.4
2019/11/7

ス間分散を大きくするように学習させる計量学習は画像検索でよく用いられ，検索においても有益であると考えられる．Deep Neural Network(DNN)を用いて，計量学習を行う手法は Deep Metric Learning と呼ばれている．本論文では，上述の２つの条件を満足する Deep Metric Learning のうちの１つである CosFace[5]を利用したすることを検討する．

通常の DCNNs を用いた画像検索では最終層の１つ前の層の特徴ベクトルを抽出し，画像検索用インデックスとして利用する．しかし，その結果，意味的には近くても物体としては異なる対象が検索されてしまうという誤りが発生しやすくなる．

この問題を解決するため，本論文では最終層以外の複数の中間層から特徴ベクトルを取り出し，これらを統合して検索インデックスとして利用する Feature Tuning を提案する. DCNNs は層の深さによって捉える画像特徴の性質が異なる．入力に近い層では色やテクスチャ情報を反映した特徴，出力に近い層では物体の意味的情報を反映した特徴が含まれる．これらの特徴を全て用いるために，DCNNs の複数の中間層から抽出した特徴ベクトルを統合し，再学習を行なって検索用インデックスを生成する手法が Feature Tuning である．この手法は色やテクスチャ情報を重視する特定物体認識のタスクに対して特に効果的であり，UKBench[6]や Holidays[7]などのデータセットでは，非常に高い性能が得られることを確認した．

2. 関連研究

画像検索用インデックスの生成法は局所特徴量を用いたものと DCNNs を用いたものとの大きく２つに分かれる．2.1 では DCNNs が現れるまで，主に利用されていた局所特徴量を用いた画像検索の関連研究について述べる. 2.2 では DCNNs を用いた画像検索の関連研究について述べる．

2.1 局所特徴量を用いた画像検索

画像検索の分野では SIFT や SURF[8]といったキーポイント検出手法で表現される局所特徴量がよく用いられる．これらの手法は被写体の隠れ，回転，照明変化に対して頑健であり，高速かつ安定して画像検索を行えることが多くの実験で実証されている．また，局所特徴量を用いた画像検索の代表例に Bag-of-Feature(BoF)アプローチを用いたものがある．BoF アプローチとは画像を１つの高次元かつスパースな BoF ベクトルで表現する手法である．BoF ベクトルは画像から得られる局所特徴をクラスタリングし，クラスタ中心である Visual Words ごとの出現回数をカウントすることによって生成される．この BoF ベクトル同士の距離計算によって画像検索が行われる．

しかし，局所特徴量を用いた画像検索はキーポイントの検出が困難, あるいは不安定な場合には有効ではないため，近年では DCNNs が主に利用されるようになっている[9].

2.2 DCNNs を用いた画像検索

DCNNs を用いた画像検索はバイナリコードを学習するものとクラス分類のネットワークを利用するものがある[10]．前者は検索用のバイナリコード自体を学習するので，ハミング距離を用いた効率的な検索を可能とするというメリットがある．後者はクラス分類のネットワークをそのまま応用することができるというメリットがあり，本論文ではクラス分類のネットワークを用いた画像検索を扱う．

クラス分類用の DCNNs を用いた画像検索の多くは ImageNet を用いた事前学習済みネットワークをファインチューニングしたものを利用している[9]．クラス分類とは異なり，画像検索では特定物体毎に画像を用意する必要があり，多くの画像を集めることは困難である．学習に用いることができる画像が十分に無く, 精度が向上しないため，事前学習済みネットワークを用いる必要がある．

画像検索用インデックスとして用いるのはネットワークの最終層から１つ前の層の出力である．最終層の出力はクラス分類を行う為の出力であるので，その１つ前の特徴が識別に有効な特徴であると言える. この特徴ベクトルの L2 ノルムを正規化するように DCNNs を学習しておけば，前述の通りこの特徴ベクトルは距離を用いた画像検索に利用することができる．

画像検索は各画像の特徴の距離間を用いて相違度を求めている. DCNNs を用いるこのようなタスクには Deep Metric Learning が用いられることがある．Deep Metric Learning は大きく分けて２つに分類できる．１つは Triplet Loss[11]や Contrastive Loss[12]と呼ばれる手法である. これらの手法は画像を複数枚同時に学習させ，その画像ペアが類似サンプル同士なら, 特徴空間の距離を小さくするように学習させ，非類似サンプル同士なら，距離を大きくするように学習させる．しかし，このような手法は画像のペアを大量に作ることが難しく，それらをバランス良く学習させることも難しい[5]. もう１つは CosFace や ArcFace[13]などの通常のクラス分類用ネットワークを用いて Deep Metric Learning を行う手法である．これらの手法はネットワークの最終層の一つ前の出力を L2 ノルムで正規化し，ベクトルの角度評価を行う為，距離による検索にも用いることができる．その角度評価にマージンを加えることで，クラス内分散が小さく，クラス外分散を大きくする学習が行える．この手法は，Triplet Loss などとは異なり，ペアを作る必要もなく，比較的容易に学習が可能である．本研究では Deep Metric Learning の１つである CosFace を用いる．

本論文では DCNNs の中間層を利用するが，中間層のそれぞれの役割を Wei[14]らの図 1 を用いて説明する．図 1 は AlexNet の各中間層を可視化したもので，入力に近い層ではエッジや色，テクスチャの情報が抽出され，出力に近い層ではクラス分類に必要な意味的な情報が抽出されている．このように各中間層の役割は DCNNs の層の深さによ

Vol.2019-CG-176 No.4
Vol.2019-DCC-23 No.4
Vol.2019-CVIM-219 No.4
2019/11/7

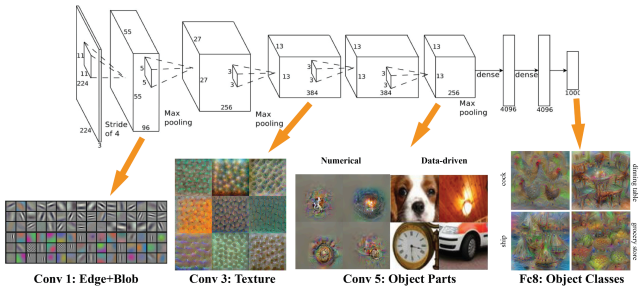

図 1　AlexNet 中間層の役割([14]から引用)

って異なっている．これらの中間層を用いて画像検索を行なったのが Yue-Hei ら[15]である．Yue-Hei らは各中間層の出力を用いてインデックスを作成している．データセットによっては出力に近い層よりも，入力側に近い層の出力が良い精度を出すことが確認されている．これは入力側に近い層の方がテクスチャを鮮明に解析できるからである．また，Husain [16]らは最終層に近い 3 層を組み合わせて特徴ベクトルを作成している．これは各 3 層で画像の注目している場所が異なる為，組み合わせることによって精度が向上したと考察している．出力に近い 3 層の組み合わせ方を複数のパターンで実験をしているが，一番良い精度が出たのが 3 層全てを使うという結果であった．Rongsheng[17]らは，ランドマーク検索用のデータセットに対して，入力に近い中間層も含めて足し合わせたインデックスの作成を行なったが，入力に近い層ではノルムが小さく，また悪影響を及ぼし，精度が悪くなったという考察を述べている．最終的には出力側に近い層の組み合わせが良い結果だったと結論を出している．

本論文では複数の中間層を統合するネットワークを用いて再学習し，そのネットワークから得られた特徴ベクトルを検索用インデックスとする手法を提案する．この手法は，色やテクスチャ情報を重視するデータセットに対して特に効果的である事を，実験を通じて確認した．

3. 提案手法

本章では提案手法について述べる．3.1 では提案手法に用いる DCNNs の中間層の性質について述べる．3.2 では DCNNs の中間層から取り出した特徴ベクトルを統合し，そのベクトルを再学習する Feature Tuning について説明を行う．

3.1 各層の性質

通常の画像検索では DCNNs の最終層から 1 つ前の層の出力を取り出し，その特徴ベクトルを用いて画像検索を行う．しかし，その取り出した特徴ベクトルは物体の意味的特徴を捉えているもので，色やテクスチャなどの特徴が多く含まれていない．この為，意味的には近い物体が誤検索されるという問題がある．2.2 で述べたように，DCNNs は層の深さで画像から得る特徴が異なっている為，我々は DCNNs の中間層を用いることによって，この問題が解決されるのではないかと考えた．本節ではこの仮説を説明するために必要な各中間層の性質について UKBench[6] と ResNet18[18]を用いる予備実験にて説明を行う．

UKBench は 10200 枚，2550 クラスの画像データセットである．各クラスは 4 枚ずつで，同じ物体を角度と照明条件を変化させ撮影している．1 枚の画像をクエリと与え，クエリ画像を含め検索上位 4 枚までが同じクラスである枚数を評価(NS-Score)とするデータセットである．NS-Score が 4 に近いほど高い精度を意味する．今回の予備実験では簡単のため，データセットを 1/10 の画像枚数にしている．

ResNet とは勾配消失問題を解決するための residual network を利用した DCNNs の 1 つであり，現在でも幅広く利用されている．本論文では図 2 のような構成をした ResNet18 を用いる．conv2 から conv5 では residual network を使用している．Last Layer では全結合層，もしくは CosFace に用いる層を使用する．今回の予備実験では中間層として conv2, conv3, conv4, conv5, Global Average Pooling(GAP)[19] の出力を利用する．それぞれの特徴ベクトルの次元は 64 次元，128 次元，256 次元，512 次元，512 次元である．

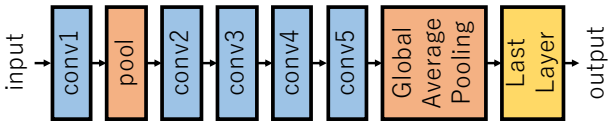

図 2　ResNet の構成

conv2，conv3，conv4，conv5 は縦横方向の空間的広がりを持つので，これらに対しては Global Average Pooling を利用してチャネル毎にスカラー値に変換し，特徴ベクトルを得ている．これらの特徴ベクトルを検索用インデックスとして利用し，UKBench に対して画像検索を行う．ResNet18 は ImageNet による学習済みネットワークを使用し，ファインチューニングを行った．各画像の検索用インデックス毎にコサイン類似度の大きいもの 4 件の画像検索を行って求めた NS-Score を表 1 に示す．

表 1　各層の NS-Score

	conv2	conv3	conv4	conv5	GAP
NS-Score	3.20	3.38	3.83	3.97	3.98

この結果から，出力側に近い層の方が良い精度が得られていることがわかる．また，各層でクエリと検索結果が一致しなかった画像の例を図 3 図 4 に示す．左がクエリ画像で，右が検索上位 1 位の画像である．図 3 のように入力に近い層では色やテクスチャが似ている画像が誤検索されていたのに対し，図 4 のように出力に近い層では意味的には同じであるが色やテクスチャが異なったものが誤検索される画像組があった．また，各層の誤検索された画像を確認したところ，各層全てにおいて誤検索された画像は 1 枚だけであった．

Vol.2019-CG-176 No.4
Vol.2019-DCC-23 No.4
Vol.2019-CVIM-219 No.4
2019/11/7

クエリ　　検索上位 1 位

図 3　conv2 での誤検索例

図 4　GAP での誤検索例

このように各層の特徴は異なり，それぞれ別の特徴を得ていることがわかった．このことから各層の特徴ベクトルを統合し，インデックスを作成することで，良い精度が出せるのではないかと考えられる．

3.2 Feature Tuning

3.1 により各層の性質がわかり，それらを統合することによって，精度向上が見込めるのではないかと考えた．しかし，ただ特徴ベクトルを統合するだけでは，特徴のバランスが取れないという問題が起きる．それは特徴によってはテクスチャを重視した方が良いものや意味的な特徴を重視したものが良い場合があるからである．その場合，単純に中間層を足し合わせるだけではうまくいかない．そこで本節では中間層から取り出した特徴ベクトルを再学習する手法の説明をする．本論文ではこの手法を Feature Tuning と呼ぶことにする．

先ほどと同様に中間層を取り出し，全て足し合わせた 1472 次元の特徴ベクトルを作成する．その特徴ベクトルを入力とした全結合層のみの DNN を作成し再学習を行う．今回，Feature Tuning を行う ResNet18 と再学習用 DNN は図 5 図 6 のようなネットワークを使用した．図 5 は全結合層(Fully Connected Layer)を 3 層使用したもので，fully1 と fully2 には各出力後に Relu 関数を使用している．図 6 も同様に全結合層を 6 層使用したもので，fully1 から fully5 までは Relu 関数を使用している．各レイヤーの上には出力するベクトルの次元数を示している．今回は DNN の最終層の 1 つ前の出力を 512 次元に設定した．これは ResNet18 の最終層一つ前の出力と条件を合わせる為である．

Feature Tuning の学習方法とテストの方法を図 5 を用いて説明を行う．はじめに ResNet18 の部分をクラス分類と同様の学習を行う．次にその学習済みの ResNet18 に画像を入力する．それによって得られる特徴ベクトルを足し合わせたものを全結合層のみの DNN へ入力し，再学習を行う．再学習に用いる最適化関数や損失関数などの環境は，ResNet18 の学習時と同様のものを利用する．これが Feature Turing の大まかな流れである．実験を行う際は ResNet18 と DNN の二つのネットワークを利用して，DNN の最終層一つ前の出力を利用して画像検索用インデックスを作成する．

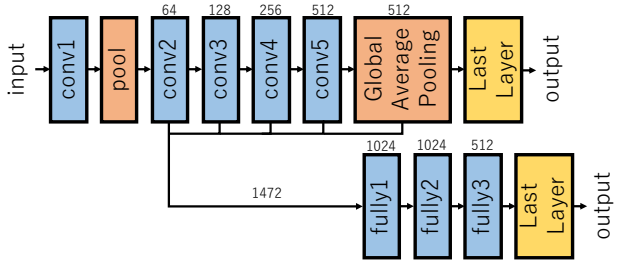

図 5　3 層の DNN による Feature Tuning

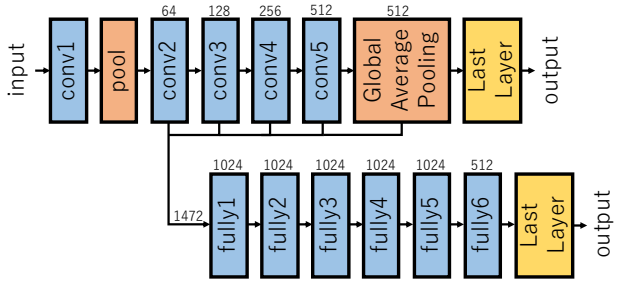

図 6　6 層の DNN による Feature Tuning

4. 実験

3 章で説明した提案手法と通常の画像検索との精度の比較実験を行う．提案手法である Feature Turning の他に中間層を足し合わせた addition と L2 ノルムで正規化した normalized addition も比較実験に含める．画像検索によく用いられる Deep Metric Learning の内の 1 つである CosFace を用いた実験も行う．

4.1 実験環境

実験には ImageNet による学習済みの ResNet18 を使用する．画像の入力サイズは 224x224 にリサイズしてから入力する．バッチサイズは 64，エポックは 120，moumentum は 5e-4，Learning Rate の初期値は 0.01 と設定した．Learning Rate はエポックが 60，90，110 の時の 0.2 倍するように変化させる．CosFace のマージンとスケールのパラメータは 0.45 と 60 である． CosFace を使用する場合は図 5 と図 6 の Last Layer の部分を全結合層から CosFace 用の層に入れ替える．

4.2 データセット

データセットは UKBench，Holidays，Oxford5k[20]を使用した．UKBench は上記において説明を行ったので省略する．

Vol.2019-CG-176 No.4
Vol.2019-DCC-23 No.4
Vol.2019-CVIM-219 No.4
2019/11/7

表 2　softmax を用いた学習による実験結果

	softmax	addition	normalized addition	Feature Tuning（3 層の DNN）	Feature Tuning（6 層の DNN）
UKBench (NS-Score)	3.73	3.78	3.84	3.84	**3.91**
Holidays (mAP)	0.855	0.881	0.900	0.906	**0.977**
Oxford5k (mAP)	**0.485**	0.456	0.448	0.435	0.0167

表 3　CosFace を用いた学習による実験結果

	cosface	addition	normalized addition	Feature Tuning（3 層の DNN）	Feature Tuning（6 層の DNN）
UKBench (NS-Score)	3.93	3.94	3.93	**4.00**	**4.00**
Holidays (mAP)	0.955	0.960	0.963	**0.998**	**0.998**
Oxford5k (mAP)	**0.658**	0.585	0.573	0.617	0.576

Holidays はあらゆる景色の画像データセットである．画像枚数は 1491 枚でクラス数は 500 である．各クラス 1 枚をクエリとして与えて検索し，同クラスが検索上位に出てくるかで評価を行う．Oxford5k は Oxford にあるランドマークの画像データセットである．画像枚数は 5062 枚でクラス数は 11 である．各クラス 5 つのクエリを用いて検索し，同クラスが検索上位に出るかで評価を行う．

全てのデータセットの画像はクエリも含め学習に用いる．今回の学習には画像に対するデータオーギュメントを行っていない．UKBench は NS-Score を使用し，Holidays と Oxford5k は公開されている評価方法である mAP を使用している．

4.3 中間層の統合

本節では Feature Turning を行わずに，特徴ベクトルを足し合わせた手法である addition と normalized addition の説明を行う．

統合する層は Feature Turning と同様に ResNet18 の conv2，conv3，conv4，conv5，GAP の出力を利用する．それらを単純に足し合わせ統合したものを addition と呼ぶことにする．全て足し合わせた次元は 1472 次元となる．しかし，単純に足し合わせるだけでは各層の出力のノルムの差が影響する．UKBench を用いて各層の出力の平均ノルムを求めたものが表 4 である．

表 4　各層の出力の平均ノルム

	conv2	conv3	conv4	conv5	GAP
ノルム	4.44	2.25	2.47	32.3	47.9

このように入力に近い層の出力はノルムが小さく為，影響が小さくなると考えた．そこで各層の出力を正規化してから足し合わせる normalized addition と呼ぶ．正規化することによって各層の出力の影響を等しくすることが目的である．

4.4 評価結果

表 2 表 3 が実験結果である．表の Softmax と CosFace は通常の画像検索を行なった結果である．UKBench と Holidays では通常の学習よりも提案手法が良いという結果となった．addition や normalized addition の精度も向上しているが，Feature Tuning を使用した場合が最も良い精度となっている．また，CosFace を使用して学習させた場合も同様に提案手法の精度が向上していて，UKBench では NS-Score が 4，すなわち精度 100%の結果が得られた．

しかし，Oxford5k に対しては提案手法の精度が悪くなっていることがわかる．この提案手法には不向きなデータセットがあり，次節でその説明を行う．

4.5 不向きなデータセット

Oxford5k のみ提案手法の精度が悪くなった原因を Grad-CAM[21]を利用して説明を行う．Grad-CAM とは CNN の判断根拠の可視化ツールである．CNN がクラス分類を行う際に，画像のどこに注目しているかを理解するために使用される．図 7 が UKBecnch と Oxford5k の一部の画像に対する可視化の結果である．各行の中央が Resnet18 の conv2 に対する可視化で，右側が conv5 に対する可視化である．ヒートマップが赤い方が強く注目していることを表している．図 7 を見るとわかるように，出力に近い層である conv5 では物体に対して強く注目していることが分かる．それに比べ，入力に近い層の conv2 では画像全体に対してまばらに注目している．Oxford5k の conv2 の結果を見ると，建物だ

Vol.2019-CG-176 No.4
Vol.2019-DCC-23 No.4
Vol.2019-CVIM-219 No.4
2019/11/7

けではなく,周りの門の縁にも注目していることがわかる．このように，Oxford5k では対象物体以外にも遮蔽物などが写っている場合があり，提案手法のように画像全体に対して注目する入力に近い層を用いると，対象物体以外の特徴を得ることになる．その為，今回の実験では精度の向上につながらなかったと考えらえる．一方で UKBench や Holidays のようなデータセットは対象物体の撮影角度や照明条件は変化しているが，同じ対象物体が写っている画像は同じ場所で撮影されているので，背景の変化や遮蔽物がない．その為，提案手法の精度が向上したと考えられる．

このことから本研究の提案手法では検索対象の背景の環境が同じ，もしくは背景がないデータセットへの画像検索が有効であることが分かる．本論文では，使用データの公開許諾条件により,結果を公開することができなかったが,物体検出と本手法を組み合わせた商品検索のようなタスクでは大きな有効性を確認することができた．これは物体検出を行い，対象物体だけを抽出することで本手法の弱点を消すことができるからである．

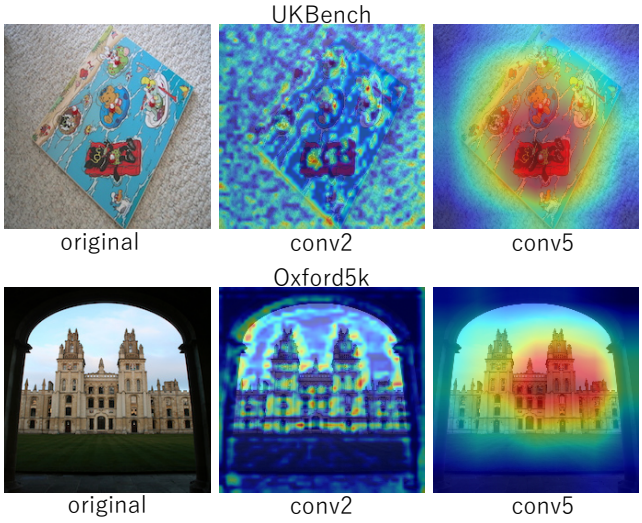

図 7　Grad-CAM による可視化

5. おわりに

本論文では DCNNs の複数の中間層を取り出し，足し合わせ再学習を行なった Feature Tuning を提案した．この手法は色やテクスチャ情報を重視したデータセットでは通常の DCNNs を用いた画像検索よりも精度が良いということを示した．また，物体検出と本手法を組み合わせたタスクにおいても有効性があることを確認している．

今後は Feature Tuning に用いる全結合層のみの DNN の改善を行いたい．また，本論文では精度が向上しなかった Oxford5k などのデータセットに対して，効果があると見込まれる Attention 機構の導入も考えている．出力側に近い層では対象物体に対して注目している為，その注目を入力側に近い層に Attention させることによって，背景や遮蔽物に依存せずに複数の層から特徴を得られると考えられる．

参考文献

[1] Lowe, David G. "Distinctive image features from scale-invariant keypoints." International journal of computer vision 60.2 (2004): 91-110.
[2] Deng, Jia, et al. "Imagenet: A large-scale hierarchical image database." 2009 IEEE conference on computer vision and pattern recognition. Ieee, 2009.
[3] Lin, Tsung-Yi, et al. "Microsoft coco: Common objects in context." European conference on computer vision. Springer, Cham, 2014.
[4] Redmon, Joseph, et al. "You only look once: Unified, real-time object detection." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
[5] Wang, Hao, et al. "Cosface: Large margin cosine loss for deep face recognition." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018.
[6] Nister, David, and Henrik Stewenius. "Scalable recognition with a vocabulary tree." 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06). Vol. 2. Ieee, 2006.
[7] Jegou, Herve, Matthijs Douze, and Cordelia Schmid. "Hamming embedding and weak geometric consistency for large scale image search." European conference on computer vision. Springer, Berlin, Heidelberg, 2008.
[8] Bay, Herbert, Tinne Tuytelaars, and Luc Van Gool. "Surf: Speeded up robust features." European conference on computer vision. Springer, Berlin, Heidelberg, 2006.
[9] Zheng, Liang, Yi Yang, and Qi Tian. "SIFT meets CNN: A decade survey of instance retrieval." IEEE transactions on pattern analysis and machine intelligence 40.5 (2017): 1224-1244.
[10] Zhou, Wengang, Houqiang Li, and Qi Tian. "Recent advance in content-based image retrieval: A literature survey." arXiv preprint arXiv:1706.06064 (2017).
[11] Schroff, Florian, Dmitry Kalenichenko, and James Philbin. "Facenet: A unified embedding for face recognition and clustering." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.
[12] Chopra, Sumit, Raia Hadsell, and Yann LeCun. "Learning a similarity metric discriminatively, with application to face verification." *CVPR (1)*. 2005.
[13] Deng, Jiankang, et al. "Arcface: Additive angular margin loss for deep face recognition." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019.
[14] Wei, Donglai, et al. "mNeuron: A Matlab plugin to visualize neurons from deep models." Massachusetts Institute of Technology 2017<http://vision03.csail.mit.edu/cnn_art/index.html>.
[15] Yue-Hei Ng, Joe, Fan Yang, and Larry S. Davis. "Exploiting local features from deep networks for image retrieval." Proceedings of the IEEE conference on computer vision and pattern recognition workshops. 2015.
[16] Husain, Syed Sameed, and Miroslaw Bober. "REMAP: Multi-layer entropy-guided pooling of dense CNN features for image retrieval." IEEE Transactions on Image Processing (2019).
[17] Rongsheng Dong, Ming Liu, and Fengying Li. "Multilayer Convolutional Feature Aggregation Algorithm for Image Retrieval" Mathematical Problems in Engineering Volume 2019, Article ID 9794202, 12 pages
[18] He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
[19] Lin, Min, Qiang Chen, and Shuicheng Yan. "Network in network."

Vol.2019-CG-176 No.4
Vol.2019-DCC-23 No.4
Vol.2019-CVIM-219 No.4
2019/11/7

arXiv preprint arXiv:1312.4400 (2013).
[20] Philbin, James, et al. "Object retrieval with large vocabularies and fast spatial matching." 2007 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2007.
[21] Selvaraju, Ramprasaath R., et al. "Grad-cam: Visual explanations from deep networks via gradient-based localization." Proceedings of the IEEE International Conference on Computer Vision. 2017.