CNN Features off-the-shelf: an Astounding Baseline for Recognition

Ali Sharif Razavian   Hossein Azizpour   Josephine Sullivan   Stefan Carlsson
CVAP, KTH (Royal Institute of Technology)
Stockholm, Sweden
{razavian,azizpour,sullivan,stefanc}@csc.kth.se

Abstract

Recent results indicate that the generic descriptors extracted from the convolutional neural networks are very powerful. This paper adds to the mounting evidence that this is indeed the case. We report on a series of experiments conducted for different recognition tasks using the publicly available code and model of the OverFeat *network which was trained to perform object classification on ILSVRC13. We use features extracted from the* OverFeat *network as a generic image representation to tackle the diverse range of recognition tasks of object image classification, scene recognition, fine grained recognition, attribute detection and image retrieval applied to a diverse set of datasets. We selected these tasks and datasets as they gradually move further away from the original task and data the* OverFeat *network was trained to solve. Astonishingly, we report consistent superior results compared to the highly tuned state-of-the-art systems in all the visual classification tasks on various datasets. For instance retrieval it consistently outperforms low memory footprint methods except for sculptures dataset. The results are achieved using a linear SVM classifier (or L2 distance in case of retrieval) applied to a feature representation of size 4096 extracted from a layer in the net. The representations are further modified using simple augmentation techniques e.g. jittering. The results strongly suggest that features obtained from deep learning with convolutional nets should be the primary candidate in most visual recognition tasks.*

1. Introduction

"Deep learning. How well do you think it would work for your computer vision problem?" Most likely this question has been posed in your group's coffee room. And in response someone has quoted recent success stories [29, 15, 10] and someone else professed skepticism. You may have left the coffee room slightly dejected thinking "Pity I have neither the time, GPU programming skills nor large amount of labelled data to train my own network to
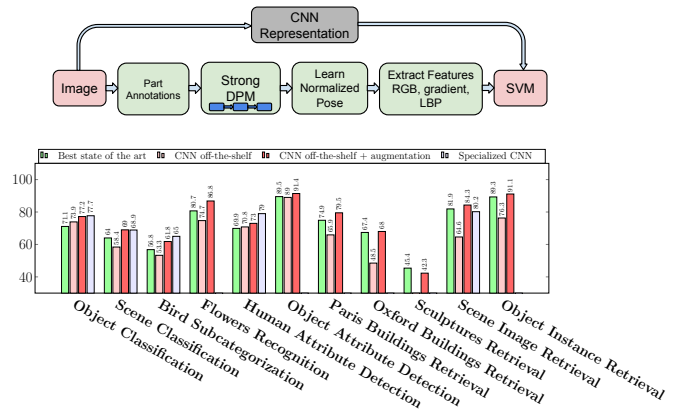
Figure 1: **top**) CNN representation replaces pipelines of s.o.a methods and achieve better results. e.g. DPD [50].
bottom) Augmented CNN representation with linear SVM consistently outperforms s.o.a. on multiple tasks. Specialized CNN refers to other works which specifically designed the CNN for their task

quickly find out the answer". But when the convolutional neural network OverFeat [38] was recently made publicly available[1] it allowed for some experimentation. In particular we wondered now, not whether one could train a deep network specifically for a given task, but if the features extracted by a deep network - one carefully trained on the diverse ImageNet database to perform the specific task of image classification - could be exploited for a wide variety of vision tasks. We now relate our discussions and general findings because as a computer vision researcher you've probably had the same questions:

Prof: First off has anybody else investigated this issue?
Student: Well it turns out Donahue *et al*. [10], Zeiler and Fergus [48] and Oquab *et al*. [29] have suggested that generic features can be extracted from large CNNs and provided some initial evidence to support this claim. But they have only considered a small number of visual recognition tasks. It would be fun to more thoroughly investigate how

---

[1] There are other publicly available deep learning implementations such as Alex Krizhevsky's ConvNet and Berkeley's Caffe. Benchmarking these implementations is beyond the scope of this paper.

powerful these CNN features are. How should we start?

Prof: The simplest thing we could try is to extract an image feature vector from the `OverFeat` network and combine this with a simple linear classifier. The feature vector could just be the responses, with the image as input, from one of the network's final layers. For which vision tasks do you think this approach would be effective?

Student: Definitely *image classification*. Several vision groups have already produced a big jump in performance from the previous sate-of-the-art methods on Pascal VOC. But maybe fine-tuning the network was necessary for the jump? I'm going to try it on Pascal VOC and just to make it a little bit trickier the MIT scene dataset.

Answer: `OverFeat` does a very good job even without fine-tuning (section 3.2 for details).

Prof: Okay so that result confirmed previous findings and is perhaps not so surprising. We asked the `OverFeat` features to solve a problem that they were trained to solve. And ImageNet is more-or-less a superset of Pascal VOC. Though I'm quite impressed by the indoor scene dataset result. What about a less amenable problem?

Student: I know *fine-grained classification*. Here we want to distinguish between sub-categories of a category such as the different species of flowers. Do you think the more generic `OverFeat` features have sufficient representational power to pick up the potentially subtle differences between very similar classes?

Answer: It worked great on a standard bird and flower database. In its most simplistic form it didn't beat the latest best performing methods but it is a much cleaner solution with ample scope for improvement. Actually, adopting a set of simple data augmentation techniques (still with linear SVM) beats the best performing methods. Impressive! (Section 3.4 for details.)

Prof: Next challenge *attribute detection*? Let's see if the `OverFeat` features have encoded something about the semantic properties of people and objects.

Student: Do you think the global CNN features extracted from the person's bounding box can cope with the articulations and occlusions present in the H3D dataset. All the best methods do some sort of part alignment before classification and during training.

Answer: Surprisingly the CNN features on average beat poselets and a deformable part model for the person attributes labelled in the H3D dataset. Wow, how did they do that?! They also work extremely well on the object attribute dataset. Maybe these `OverFeat` features do indeed encode attribute information? (Details in section 3.5.)

Prof: Can we push things even further? Is there a task `OverFeat` features should struggle with compared to more established computer vision systems? Maybe *instance retrieval*. This task drove the development of the SIFT and VLAD descriptors and the bag-of-visual-words approach

followed swiftly afterwards. Surely these highly optimized engineered vectors and mid-level features should win hands down over the generic features?

Student: I don't think CNN features have a chance if we start comparing to methods that also incorporate 3D geometric constraints. Let's focus on descriptor performance. Do new school descriptors beat old school descriptors in the old school descriptors' backyard?

Answer: Very convincing. Ignoring systems that impose 3D geometry constraints the CNN features are very competitive on building and holiday datasets (section 4). Furthermore, doing standard instance retrieval feature processing (*i.e.* PCA, whitening, renormalization) it shows superior performance compared to low memory footprint methods on all retrieval benchmarks except for the sculptures dataset.

Student: The take home message from all these results?

Prof: It's all about the features! SIFT and HOG descriptors produced big performance gains a decade ago and now deep convolutional features are providing a similar breakthrough for recognition. Thus, applying the well-established computer vision procedures on CNN representations should potentially push the reported results even further. In any case, if you develop any new algorithm for a recognition task then it **must** be compared against the strong baseline of *generic deep features + simple classifier*.

2. Background and Outline

In this work we use the publicly available trained CNN called `OverFeat` [38]. The structure of this network follows that of Krizhevsky *et al.* [22]. The convolutional layers each contain 96 to 1024 kernels of size 3×3 to 7×7. Half-wave rectification is used as the nonlinear activation function. Max pooling kernels of size 3×3 and 5×5 are used at different layers to build robustness to intra-class deformations. We used the "large" version of the `OverFeat` network. It takes as input color images of size 221×221. Please consult [38] and [22] for further details.

`OverFeat` was trained for the image classification task of ImageNet ILSVRC 2013 [1] and obtained very competitive results for the classification task of the 2013 challenge and won the localization task. ILSVRC13 contains 1.2 million images which are hand labelled with the presence/absence of 1000 categories. The images are mostly centered and the dataset is considered less challenging in terms of clutter and occlusion than other object recognition datasets such as PASCAL VOC [12].

We report results on a series of experiments we conducted on different recognition tasks. The tasks and datasets were selected such that they gradually move further away from the task the `OverFeat` network was trained to perform. We have two sections for visual classification (Sec. 3) and visual instance retrieval (Sec. 4) where we review different tasks and datasets and report the final results. The crucial

thing to remember is that the CNN features used are trained only using ImageNet data though the simple classifiers are trained using images specific to the task's dataset.

Finally, we have to point out that, given enough computational resources, optimizing the CNN features for specific tasks/datasets would probably boost the performance of the simplistic system even further [29, 15, 51, 43, 41].

3. Visual Classification

Here we go through different tasks related to visual classification in the following subsections.

3.1. Method

For all the experiments, unless stated otherwise, we use the first fully connected layer (layer 22) of the network as our feature vector. Note the max-pooling and rectification operations are each considered as a separate layer in `OverFeat` which differs from Alex Krizhevsky's ConvNet numbering. For all the experiments we resize the whole image (or cropped sub-window) to 221×221. This gives a vector of 4096 dimensions. We have two settings:

- The feature vector is further $L2$ normalized to unit length for all the experiments. We use the 4096 dimensional feature vector in combination with a Support Vector Machine (SVM) to solve different classification tasks (CNN-SVM).

- We further augment the training set by adding cropped and rotated samples and doing component-wise power transform and report separate results (CNNaug+SVM).

For the classification scenarios where the labels are not mutually exclusive (*e.g.* VOC Object Classification or UIUC Object attributes) we use a one-against-all strategy, in the rest of experiments we use one-against-one linear SVMs with voting. For all the experiments we use a linear SVM found from eq.1, where we have training data $\{(\mathbf{x}_i, y_i)\}$.

$$\underset{\mathbf{w}}{\text{minimize}} \ \frac{1}{2}\|\mathbf{w}\|^2 + C \sum_i \max(1 - y_i\mathbf{w}^T\mathbf{x}_i, 0) \qquad (1)$$

Further information can be found in the implementation details at section 3.6.

3.2. Image Classification

To begin, we adopt the CNN representation to tackle the problem of image classification of objects and scenes. The system should assign (potentially multiple) semantic labels to an image. Remember in contrast to object detection, object image classification requires no localization of the objects. The CNN representation has been optimized for the object image classification task of ILSVRC. Therefore, in this experiment the representation is more aligned with the

final task than the rest of experiments. However, we have chosen two different image classification datasets, objects and indoor scenes, whose image distributions differ from that of ILSVRC dataset.

3.2.1 Datasets

We use two challenging recognition datasets, Namely, Pascal VOC 2007 for object image classification [12] and the MIT-67 indoor scenes [36] for scene recognition.

Pascal VOC. Pascal VOC 2007 [12] contains ~10000 images of 20 classes including animals, handmade and natural objects. The objects are not centered and in general the appearance of objects in VOC is perceived to be more challenging than ILSVRC. Pascal VOC images come with bounding box annotation which are *not* used in our experiments.

MIT-67 indoor scenes. The MIT scenes dataset has 15620 images of 67 indoor scene classes. The dataset consists of different types of stores (*e.g.* bakery, grocery) residential rooms (*e.g.* nursery room, bedroom), public spaces (*e.g.* inside bus, library, prison cell), leisure places (*e.g.* buffet, fastfood, bar, movietheater) and working places (*e.g.* office, operating room, tv studio). The similarity of the objects present in different indoor scenes makes MIT indoor an especially difficult dataset compared to outdoor scene datasets.

3.2.2 Results of PASCAL VOC Object Classification

Table 1 shows the results of the `OverFeat` CNN representation for object image classification. The performance is measured using average precision (AP) criterion of VOC 2007 [12]. Since the original representation has been trained for the same task (on ILSVRC) we expect the results to be relatively high. We compare the results only with those methods which have used training data outside the standard Pascal VOC 2007 dataset. We can see that the method outperforms all the previous efforts by a significant margin in mean average precision (mAP). Furthermore, it has superior average precision on 10 out of 20 classes. It is worth mentioning the baselines in Table 1 use sophisticated matching systems. The same observation has been recently made in another work [29].

Different layers. Intuitively one could reason that the learnt weights for the deeper layers could become more specific to the images of the training dataset and the task it is trained for. Thus, one could imagine the optimal representation for each problem lies at an intermediate level of the network. To further study this, we trained a linear SVM for all classes using the output of each network layer. The result is shown in Figure 2a. Except for the fully connected

	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP
GHM[8]	76.7	74.7	53.8	72.1	40.4	71.7	83.6	66.5	52.5	57.5	62.8	51.1	81.4	71.5	86.5	36.4	55.3	60.6	80.6	57.8	64.7
AGS[11]	82.2	83.0	58.4	76.1	**56.4**	**77.5**	**88.8**	69.1	**62.2**	61.8	64.2	51.3	**85.4**	**80.2**	91.1	48.1	61.7	**67.7**	86.3	70.9	71.1
NUS[39]	82.5	79.6	64.8	73.4	54.2	75.0	77.5	79.2	46.2	62.7	41.4	74.6	85.0	76.8	91.1	53.9	61.0	67.5	83.6	70.6	70.5
CNN-SVM	88.5	81.0	83.5	82.0	42.0	72.5	85.3	81.6	59.9	58.5	66.5	77.8	81.8	78.8	90.2	54.8	71.1	62.6	87.2	71.8	73.9
CNNaug-SVM	**90.1**	**84.4**	**86.5**	**84.1**	48.4	73.4	86.7	**85.4**	61.3	**67.6**	**69.6**	**84.0**	**85.4**	80.0	**92.0**	**56.9**	**76.7**	67.3	**89.1**	**74.9**	**77.2**

Table 1: **Pascal VOC 2007 Image Classification Results** compared to other methods which also use training data outside VOC. The CNN representation is not tuned for the Pascal VOC dataset. However, GHM [8] learns from VOC a joint representation of bag-of-visual-words and contextual information. AGS [11] learns a second layer of representation by clustering the VOC data into subcategories. NUS [39] trains a codebook for the SIFT, HOG and LBP descriptors from the VOC dataset. Oquab *et al.* [29] fixes all the layers trained on ImageNet then it adds and optimizes two fully connected layers on the VOC dataset and achieves better results (**77.7**) indicating the potential to boost the performance by further adaptation of the representation to the target task/dataset.
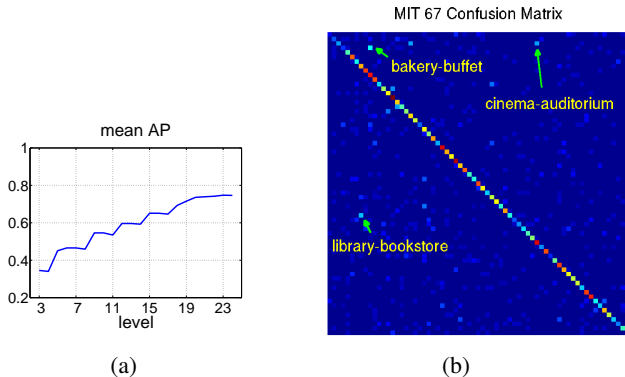

(a)   (b)

Figure 2: **a)** Evolution of the mean image classification AP over PASCAL VOC 2007 classes as we use a deeper representation from the `OverFeat` CNN trained on the ILSVRC dataset. `OverFeat` considers convolution, max pooling, nonlinear activations, etc. as separate layers. The re-occurring decreases in the plot is of the activation function layer which loses information by half rectifying the signal. **b)** Confusion matrix for the MIT-67 indoor dataset. Some of the off-diagonal confused classes have been annotated, these particular cases could be hard even for a human to distinguish.

last 2 layers the performance increases. We observed the same trend in the individual class plots. The subtle drops in the mid layers (*e.g.* 4, 8, etc.) is due to the "ReLU" layer which half-rectifies the signals. Although this will help the non-linearity of the trained model in the CNN, it does not help if immediately used for classification.

3.2.3 Results of MIT 67 Scene Classification

Table 2 shows the results of different methods on the MIT indoor dataset. The performance is measured by the average classification accuracy of different classes (mean of the confusion matrix diagonal). Using a CNN off-the-shelf representation with linear SVMs training significantly outperforms a majority of the baselines. The non-CNN baselines benefit from a broad range of sophisticated designs. confusion matrix of the CNN-SVM classifier on the 67 MIT classes. It has a strong diagonal. The few relatively bright off-diagonal points are annotated with their ground truth and estimated labels. One can see that in these examples the two labels could be challenging even for a human to distin-

Method	mean Accuracy
ROI + Gist[36]	26.1
DPM[30]	30.4
Object Bank[24]	37.6
RBow[31]	37.9
BoP[21]	46.1
miSVM[25]	46.4
D-Parts[40]	51.4
IFV[21]	60.8
MLrep[9]	64.0
CNN-SVM	58.4
CNNaug-SVM	**69.0**
CNN(AlexConvNet)+multiscale pooling [16]	68.9

Table 2: **MIT-67 indoor scenes dataset**. The MLrep [9] has a fine tuned pipeline which takes weeks to select and train various part detectors. Furthermore, Improved Fisher Vector (IFV) representation has dimensionality larger than 200K. [16] has very recently tuned a multi-scale orderless pooling of CNN features (off-the-shelf) suitable for certain tasks. With this simple modification they achieved significant average classification accuracy of **68.88**.

guish between, especially for close-up views of the scenes.

3.3. Object Detection

Unfortunately, we have not conducted any experiments for using CNN off-the-shelf features for the task of object detection. But it is worth mentioning that Girshick *et al.* [15] have reported remarkable numbers on PASCAL VOC 2007 using off-the-shelf features from `Caffe` code. We repeat their relevant results here. Using off-the-shelf features they achieve a mAP of **46.2** which already outperforms state of the art by about 10%. This adds to our evidences of how powerful the CNN features off-the-shelf are for visual recognition tasks.

Finally, by further fine-tuning the representation for PASCAL VOC 2007 dataset (not off-the-shelf anymore) they achieve impressive results of **53.1**.

3.4. Fine grained Recognition

Fine grained recognition has recently become popular due to its huge potential for both commercial and cataloging applications. Fine grained recognition is specially inter-

esting because it involves recognizing subclasses of the same object class such as different bird species, dog breeds, flower types, etc. The advent of many new datasets with fine-grained annotations such as Oxford flowers [27], Caltech bird species [45], dog breeds [1], cooking activities [37], cats and dogs [32] has helped the field develop quickly. The subtlety of differences across different subordinate classes (as opposed to different categories) requires a fine-detailed representation. This characteristic makes fine-grained recognition a good test of whether a generic representation can capture these subtle details.

3.4.1 Datasets

We evaluate CNN features on two fine-grained recognition datasets CUB 200-2011 and 102 Flowers.

Caltech-UCSD Birds (CUB) 200-2011 dataset [45] is chosen since many recent methods have reported performance on it. It contains 11,788 images of 200 bird subordinates. 5994 images are used for training and 5794 for evaluation. Many of the species in the dataset exhibit extremely subtle differences which are sometimes even hard for humans to distinguish. Multiple levels of annotation are available for this dataset - bird bounding boxes, 15 part landmarks, 312 binary attributes and boundary segmentation. The majority of the methods applied use the bounding box and part landmarks for training. In this work we only use the bounding box annotation during training and testing.

Oxford 102 flowers dataset [27] contains 102 categories. Each category contains 40 to 258 of images. The flowers appear at different scales, pose and lighting conditions. Furthermore, the dataset provides segmentation for all the images.

3.4.2 Results

Table 3 reports the results of the CNN-SVM compared to the top performing baselines on the CUB 200-2011 dataset. The first two entries of the table represent the methods which only use bounding box annotations. The rest of baselines use part annotations for training and sometimes for evaluation as well.

Table 4 shows the performance of CNN-SVM and other baselines on the flowers dataset. All methods, bar the CNN-SVM, use the segmentation of the flower from the background. It can be seen that CNN-SVM outperforms all basic representations and their multiple kernel combination even without using segmentation.

3.5. Attribute Detection

An attribute within the context of computer vision is defined as some semantic or abstract quality which different instances/categories share.

Method	Part info	mean Accuracy
Sift+Color+SVM[45]	✗	17.3
Pose pooling kernel[49]	✓	28.2
RF[47]	✓	19.2
DPD[50]	✓	51.0
Poof[5]	✓	56.8
CNN-SVM	✗	53.3
CNNaug-SVM	✗	**61.8**
DPD+CNN(DeCaf)+LogReg[10]	✓	**65.0**

Table 3: **Results on CUB 200-2011 Bird dataset.** The table distinguishes between methods which use part annotations for training and sometimes for evaluation as well and those that do not. [10] generates a pose-normalized CNN representation using DPD [50] detectors which significantly boosts the results to **64.96**.

Method	mean Accuracy
HSV [27]	43.0
SIFT internal [27]	55.1
SIFT boundary [27]	32.0
HOG [27]	49.6
HSV+SIFTi+SIFTb+HOG(MKL) [27]	72.8
BOW(4000) [14]	65.5
SPM(4000) [14]	67.4
FLH(100) [14]	72.7
BiCos seg [7]	79.4
Dense HOG+Coding+Pooling[2] w/o seg	76.7
Seg+Dense HOG+Coding+Pooling[2]	80.7
CNN-SVM w/o seg	74.7
CNNaug-SVM w/o seg	**86.8**

Table 4: **Results on the Oxford 102 Flowers dataset.** All the methods use segmentation to subtract the flowers from background unless stated otherwise.

3.5.1 Datasets

We use two datasets for attribute detection. The first dataset is the UIUC 64 object attributes dataset [13]. There are 3 categories of attributes in this dataset: shape (*e.g.* is 2D boxy), part (*e.g.* has head) or material (*e.g.* is furry). The second dataset is the H3D dataset [6] which defines 9 attributes for a subset of the person images from Pascal VOC 2007. The attributes range from "has glasses" to "is male".

3.5.2 Results

Table 5 compares CNN features performance to state-of-the-art. Results are reported for both across and within categories attribute detection (refer to [13] for details).

Table 6 reports the results of the detection of 9 human attributes on the H3D dataset including poselets and DPD [50]. Both poselets and DPD use part-level annotations during training while for the CNN we only extract one feature from the bounding box around the person. The CNN representation performs as well as DPD and significantly outperforms poselets.

Method	within categ.	across categ.	mAUC
Farhadi *et al.* [13]	83.4	-	73.0
Latent Model[46]	62.2	79.9	-
Sparse Representation[44]	89.6	**90.2**	-
att. based classification[23]	-	-	73.7
CNN-SVM	91.7	82.2	89.0
CNNaug-SVM	**93.7**	84.9	**91.5**

Table 5: **UIUC 64 object attribute dataset results**. Compared to other existing methods the CNN features perform very favorably.

Method	male	lg hair	glasses	hat	tshirt	lg slvs	shorts	jeans	lg pants	mAP
Freq[6]	59.3	30.0	22.0	16.6	23.5	49.0	17.9	33.8	74.7	36.3
SPM[6]	68.1	40.0	25.9	35.3	30.6	58.0	31.4	39.5	84.3	45.9
Poselets[6]	82.4	**72.5**	**55.6**	60.1	51.2	74.2	45.5	54.7	90.3	65.2
DPD[50]	83.7	70.0	38.1	**73.4**	49.8	78.1	64.1	**78.1**	93.5	69.9
CNN-SVM	83.0	67.6	39.7	66.8	52.6	82.2	78.2	71.7	95.2	70.8
CNNaug-SVM	**84.8**	71.0	42.5	66.9	**57.7**	**84.0**	**79.1**	75.7	**95.3**	**73.0**

Table 6: **H3D Human Attributes dataset results.** A CNN representation is extracted from the bounding box surrounding the person. All the other methods require the part annotations during training. The first row shows the performance of a random classifier. The work of Zhang *et al.* [51] has adapted the CNN architecture specifically for the task of attribute detection and achieved the impressive performance of **78.98** in mAP. This further highlights the importance of adapting the CNN architecture for different tasks given enough computational resources.

3.6. Implementation Details

We have used precomputed linear kernels with libsvm for the CNN-SVM experiments and liblinear for the CNNaug-SVM with the primal solver (#samples $\gg$ #dim). Data augmentation is done by making 16 representations for each sample (original image, 5 crops, 2 rotation and their mirrors). The cropping is done such that the subwindow contains 4/9 of the original image area from the 4 corners and the center. We noted the following phenomenon for all datasets. At the test time, when we have multiple representations for a test image, taking the sum over all the responses works outperforms taking the max response. In CNNaug-SVM we use *signed* component-wise power transform by raising each dimension to the power of 2. For the datasets which with bounding box (*i.e.* birds, H3D) we enlarged the bounding box by 150% to include some context. In the early stages of our experiments we noticed that using one-vs-one approach works better than structured SVM for multi-class learning. Finally, we noticed that using the imagemagick library for image resizing has slight adverse effects compared to matlab imresize function. The cross-validated SVM parameter (C) used for different datasets are as follows. VOC2007:**0.2**, MIT67:**2** , Birds:**2**, Flowers:**2**, H3D:**0.2** UIUCatt:**0.2**.[2]

---

[2]The details of our system including extracted features, scripts and updated tables can be found at our project webpage: http://www.csc.kth.se/cvap/cvg/DL/ots/

4. Visual Instance Retrieval

In this section we compare the CNN representation to the current state-of-the-art retrieval pipelines including VLAD[4, 52], BoW, IFV[33], Hamming Embedding[17] and BoB[3]. Unlike the CNN representation, all the above methods use dictionaries trained on similar or same dataset as they are tested on. For a fair comparison between the methods, we only report results on representations with relevant order of dimensions and exclude post-processing methods like spatial re-ranking and query expansion.

4.1. Datasets

We report retrieval results on five common datasets in the area as follows:

Oxford5k buildings[34] This is a collection of 5063 reference photos gathered from flickr, and 55 queries of different buildings. From an architectural standpoint the buildings in Oxford5k are very similar. Therefore it is a challenging benchmark for generic features such as CNN.

Paris6k buildings[35] Similar to the Oxford5k, this collection has 55 queries images of buildings and monuments from Paris and 6412 reference photos. The landmarks in Paris6k have more diversity than those in Oxford5k.

Sculptures6k[3] This dataset brings the challenge of smooth and texture-less item retrieval. It has 70 query images and contains 6340 reference images which is halved to train/test subsets. The results on this dataset highlights the extent to which CNN features are able to encode shape.

Holidays dataset[19] This dataset contains 1491 images of which 500 are queries. It contains images of different scenes, items and monuments. Unlike the first three datasets, it exhibits a diverse set of images. For the above datasets we reported mAP as the measurement metric.

UKbench[28] A dataset of images of 2250 items each from four different viewpoints. The UKbench provides a good benchmark for viewpoint changes. We reported recall at top four as the performance over UKBench.

4.2. Method

Similar to the previous tasks we use the $L2$ normalized output of the first fully connected layer as representation.

Spatial search. The items of interest can appear at different locations and scales in the test and reference images making some form of spatial search necessary. Our crude search has the following form. For each image we extract multiple sub-patches of different sizes at different locations. Let h (the number of levels) represent the number of different sized patches we extract. At level i, $1 \leq i \leq h$, we extract i^2 overlapping sub-patches of the same size whose union covers the whole image. For each extracted sub-patch we compute its CNN representation. The distance between a query sub-patch and a reference image is defined as the min-

	Dim	Oxford5k	Paris6k	Sculp6k	Holidays	UKBench
BoB[3]	N/A	N/A	N/A	**45.4**[3]	N/A	N/A
BoW	200k	36.4[20]	46.0[35]	8.1[3]	54.0[4]	70.3[20]
IFV[33]	2k	41.8[20]	-	-	62.6[20]	83.8[20]
VLAD[4]	32k	55.5[4]	-	-	64.6[4]	-
CVLAD[52]	64k	47.8[52]	-	-	81.9[52]	89.3[52]
HE+burst[17]	64k	64.5[42]	-	-	78.0[42]	-
AHE+burst[17]	64k	66.6[42]	-	-	79.4[42]	-
Fine vocab[26]	64k	74.2[26]	74.9[26]	-	74.9[26]	-
ASMK*+MA[42]	64k	80.4[42]	77.0[42]	-	81.0[42]	-
ASMK+MA[42]	64k	**81.7**[42]	78.2[42]	-	82.2[42]	-
CNN	4k	32.2	49.5	24.1	64.2	76.0
CNN-ss	32-120k	55.6	69.7	31.1	76.9	86.9
CNNaug-ss	4-15k	**68.0**	**79.5**	42.3	**84.3**	**91.1**
CNN+BOW[16]	2k	-	-	-	80.2	-

Table 7: **The result of object retrieval on 5 datasets.** All the methods except the CNN have their representation trained on datasets similar to those they report the results on. The spatial search result on Oxford5k, Paris6k and Sculpture6k, are reported for $h_r = 4$ and $h_q = 3$. It can be seen that CNN features, when compared with low-memory footprint methods, produce consistent high results. ASMK+MA [42] and fine-vocab [26] use in order of million codebooks but with various tricks including binarization they reduce the memory foot print to 64k.

imum $L2$ distance between the query sub-patch and respective reference sub-patches. Then, the distance between the reference and the query image is set to the average distance of each query sub-patch to the reference image. In contrast to visual classification pipelines, we extract features from the smallest square containing the region of interest (as opposed to resizing). In the reset of the text, h_r denotes to the number of levels for the reference image and similarly h_q for the query image.

Feature Augmentation. Successful instance retrieval methods have many feature processing steps. Adopting the proposed pipeline of [18] and followed by others [16, 42] we process the extracted 4096 dim features in the following way: $L2$ normalize $\rightarrow$ PCA dimensionality reduction $\rightarrow$ whitening $\rightarrow$ $L2$ renormalization. Finally, we further use a signed component wise power transform and raise each dimension of the feature vector to the power of 2. For all datasets in the PCA step we reduce the dimensionality of the feature vector to 500. All the $L2$ normalizations are applied to achieve unit length.

4.3. Results

The result of different retrieval methods applied to 5 datasets are in table 7. Spatial search is only used for the first three datasets which have samples in different scales and locations. For the other two datasets we used the same jittering as explained in Sec. 3.1

It should be emphasized that we only reported the results on low memory footprint methods.

5. Conclusion

In this work, we used an off-the-shelf CNN representation, `OverFeat`, with simple classifiers to address different recognition tasks. The learned CNN model was originally optimized for the task of object classification in ILSVRC 2013 dataset. Nevertheless, it showed itself to be a strong competitor to the more sophisticated and highly tuned state-of-the-art methods. The same trend was observed for various recognition tasks and different datasets which highlights the effectiveness and generality of the learned representations. The experiments confirm and extend the results reported in [10]. We have also pointed to the results from works which specifically *optimize* the CNN representations for different tasks/datasets achieving even superior results. Thus, it can be concluded that from now on, deep learning with CNN has to be considered as the primary candidate in essentially any visual recognition task.

References

[1] Imagenet large scale visual recognition challenge 2013 (ilsvrc2013). http://www.image-net.org/challenges/LSVRC/2013/.

[2] A. Angelova and S. Zhu. Efficient object detection and segmentation for fine-grained recognition. In *CVPR*, 2013.

[3] R. Arandjelović and A. Zisserman. Smooth object retrieval using a bag of boundaries. In *ICCV*, 2011.

[4] R. Arandjelović and A. Zisserman. All about VLAD. In *CVPR*, 2013.

[5] T. Berg and P. N. Belhumeur. Poof: Part-based one-vs.-one features for fine-grained categorization, face verification, and attribute estimation. In *CVPR*, 2013.

[6] L. D. Bourdev, S. Maji, and J. Malik. Describing people: A poselet-based approach to attribute classification. In *ICCV*, 2011.

[7] Y. Chai, V. S. Lempitsky, and A. Zisserman. Bicos: A bi-level co-segmentation method for image classification. In *ICCV*, 2011.

[8] Q. Chen, Z. Song, Y. Hua, Z. Huang, and S. Yan. Hierarchical matching with side information for image classification. In *CVPR*, 2012.

[9] C. Doersch, A. Gupta, and A. A. Efros. Mid-level visual element discovery as discriminative mode seeking. In *NIPS*, 2013.

[10] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *ICML*, 2014.

[11] J. Dong, W. Xia, Q. Chen, J. Feng, Z. Huang, and S. Yan. Subcategory-aware object classification. In *CVPR*, 2013.

[12] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

[13] A. Farhadi, I. Endres, D. Hoiem, and D. A. Forsyth. Describing objects by their attributes. In *CVPR*, 2009.

[14] B. Fernando, E. Fromont, and T. Tuytelaars. Mining mid-level features for image classification. *International Journal of Computer Vision*, 2014.

[15] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *arxiv:1311.2524 [cs.CV]*, 2013.

[16] Y. Gong, L. Wang, R. Guo, and S. Lazebnik. Multi-scale orderless pooling of deep convolutional activation features. *CoRR*, 2014.

[17] M. Jain, H. Jégou, and P. Gros. Asymmetric hamming embedding: taking the best of our bits for large scale image search. In *ACM Multimedia*, pages 1441–1444, 2011.

[18] H. Jégou and O. Chum. Negative evidences and co-occurences in image retrieval: The benefit of pca and whitening. In *ECCV*, pages 774–787, 2012.

[19] H. Jégou, M. Douze, and C. Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *ECCV*, 2008.

[20] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, and C. Schmid. Aggregating local image descriptors into compact codes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(9):1704–1716, 2012.

[21] M. Juneja, A. Vedaldi, C. V. Jawahar, and A. Zisserman. Blocks that shout: Distinctive parts for scene classification. In *CVPR*, 2013.

[22] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[23] C. H. Lampert, H. Nickisch, and S. Harmeling. Attribute-based classification for zero-shot visual object categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(3), 2014.

[24] L.-J. Li, H. Su, E. P. Xing, and F.-F. Li. Object bank: A high-level image representation for scene classification & semantic feature sparsification. In *NIPS*, 2010.

[25] Q. Li, J. Wu, and Z. Tu. Harvesting mid-level visual concepts from large-scale internet images. In *CVPR*, 2013.

[26] A. Mikulík, M. Perdoch, O. Chum, and J. Matas. Learning a fine vocabulary. In *ECCV*, pages 1–14, 2010.

[27] M.-E. Nilsback and A. Zisserman. Automated flower classification over a large number of classes. In *Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing*, Dec 2008.

[28] D. Nistér and H. Stewénius. Scalable recognition with a vocabulary tree. In *CVPR*, 2006.

[29] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Learning and transferring mid-level image representations using convolutional neural networks. Technical Report HAL-00911179, INRIA, 2013.

[30] M. Pandey and S. Lazebnik. Scene recognition and weakly supervised object localization with deformable part-based models. In *ICCV*, 2011.

[31] S. N. Parizi, J. G. Oberlin, and P. F. Felzenszwalb. Reconfigurable models for scene recognition. In *CVPR*, 2012.

[32] O. M. Parkhi, A. Vedaldi, A. Zisserman, and C. V. Jawahar. Cats and dogs. In *CVPR*, 2012.

[33] F. Perronnin, Y. Liu, J. Sánchez, and H. Poirier. Large-scale image retrieval with compressed fisher vectors. In *CVPR*, 2010.

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

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

[36] A. Quattoni and A. Torralba. Recognizing indoor scenes. In *CVPR*, 2009.

[37] M. Rohrbach, S. Amin, M. Andriluka, and B. Schiele. A database for fine grained activity detection of cooking activities. In *CVPR*, 2012.

[38] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. In *ICLR*, 2014.

[39] Z. Song, Q. Chen, Z. Huang, Y. Hua, and S. Yan. Contextualizing object detection and classification. In *CVPR*, 2011.

[40] J. Sun and J. Ponce. Learning discriminative part detectors for image classification and cosegmentation. In *ICCV*, 2013.

[41] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *CVPR*, 2014.

[42] G. Tolias, Y. S. Avrithis, and H. Jégou. To aggregate or not to aggregate: Selective match kernels for image search. In *ICCV*, pages 1401–1408, 2013.

[43] A. Toshev and C. Szegedy. Deeppose: Human pose estimation via deep neural networks. In *CVPR*, 2014.

[44] G. Tsagkatakis and A. E. Savakis. Sparse representations and distance learning for attribute based category recognition. In *ECCV Workshops (1)*, pages 29–42, 2010.

[45] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

[46] Y. Wang and G. Mori. A discriminative latent model of object classes and attributes. In *ECCV*, 2010.

[47] B. Yao, A. Khosla, and F.-F. Li. Combining randomization and discrimination for fine-grained image categorization. In *CVPR*, 2011.

[48] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. *CoRR*, abs/1311.2901, 2013.

[49] N. Zhang, R. Farrell, and T. Darrell. Pose pooling kernels for sub-category recognition. In *CVPR*, 2012.

[50] N. Zhang, R. Farrell, F. Iandola, and T. Darrell. Deformable part descriptors for fine-grained recognition and attribute prediction. In *ICCV*, 2013.

[51] N. Zhang, M. Paluri, M. Ranzato, T. Darrell, and L. Bourdev. Panda: Pose aligned networks for deep attribute modeling. In *CVPR*, 2014.

[52] W.-L. Zhao, H. Jégou, G. Gravier, et al. Oriented pooling for dense and non-dense rotation-invariant features. In *BMVC*, 2013.