

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. Remarkably we report better or competitive results compared to the state-of-the-art in all the tasks on various datasets. The results are achieved using a linear SVM classifier applied to a feature representation of size 4096 extracted from a layer in the net. The results strongly suggest that features obtained from deep learning with convolutional nets should be the primary candidate in most visual classification 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 [27, 16, 11] 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 quickly find out the answer”. But when the convolutional neural network *OverFeat* [36] was recently made publicly available 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.¹ We now relate our discussions and general findings because as a computer vision researcher you’ve probably had the same questions:

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 state-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 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. It didn’t beat the latest best performing methods but it is a much cleaner solution with ample scope for improvement. Impressive. (Section 4 for details.)

¹Donahue *et al.* [11] addressed this issue mainly for fine-grained classification, but in this paper we consider a much wider range of tasks.

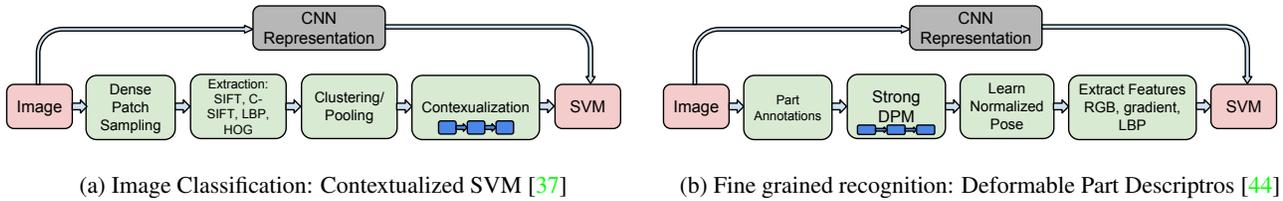


Figure 1: A CNN representation replaces pipelines of state-of-the-art methods and achieve better or comparable results for many tasks.

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 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 6).

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. 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` [36]. The structure of this network follows that of Krizhevsky *et al.* [21]. 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 [36] and [21] for further details. 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. This vector has 4096 dimensions. The feature vector is further L_2 normalized to unit length for all the experiments.

`OverFeat` was trained for the image classification task of ImageNet ILSVRC 2013 [1] and won the 2013 challenge. 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 [13].

In this paper we report 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. For each task, we have a section where we explain the datasets, a simple method of learning, and report the final results. In the final set of experiments for image retrieval we explore the results in more detail. 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.

3. Image Classification

To begin, we adopt the CNN representation to tackle the problem of object classification. 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 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 scene whose image distributions differ from the ILSVRC dataset.

	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP
GHM[9]	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[12]	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[37]	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

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 [9] learns from VOC a joint representation of bag-of-visual-words and contextual information. AGS [12] learns a second layer of representation by clustering the VOC data into subcategories. NUS [37] trains a codebook for the SIFT, HOG and LBP descriptors from the VOC dataset.

3.1. Datasets

We use two challenging recognition datasets, Namely, Pascal VOC 2007 for object image classification [13] and the MIT-67 indoor scenes [34] for scene recognition.

Pascal VOC. Pascal VOC 2007 contains ~ 10000 images of 20 classes including animals, handmade and natural objects. Images frequently have multiple annotations. The objects are typically not centered and are heavily cluttered. In general the appearance of objects in Pascal VOC is perceived to be more distorted than those in ILSVRC. Pascal VOC images come with bounding box annotation for each occurrence of an object. However, in the standard evaluation regime for image classification, as in this experiment, these annotations are not used during training.

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. In fact some of the scenes are even hard for a human to discriminate between (*e.g.* library from book store).

3.2. Method

Object classification. The Pascal VOC object image classification dataset includes multiple labels for each image and the standard evaluation measure is the average precision (AP) over precision recall (PR) curve. Therefore we adopt a one-versus-all classification regime. We use a standard linear SVM unconstrained formulation, equation (1), to train an individual linear classifier for each class from binary labelled training data $\{(\mathbf{x}_i, y_i)\}$:

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

When training an SVM for one class we used all the images containing an instance of that class as the positive samples and the rest as negative samples. We used libsvm [8] with

the trade-off parameter set to $C=5$ for all classes and chosen by cross-validation on the training set. Additionally, we augmented the set of positive samples by mirroring the positive images. We also expanded the set of negative images by adding the mirror and 2×2 sub-windows of each negative image to the set. This helps the classification by a margin of 5% in mAP.

Scene classification. Images from the MIT indoor dataset are labelled with only one semantic class. The standard measure on this dataset is the mean of the confusion matrix’s diagonal elements. To perform the multi-class classification we use a one-against-one approach and train $K(K-1)/2$ SVM binary classifiers, where K is the number of classes. A simple voting procedure then determines the winner class. We have trained individual SVMs with $C=1$ for this experiment. It should be noted that we tried a structured SVM formulation of multi-class SVM using the SVMstruct package [19], but the one-against-one approach performs better.

3.3. Results

3.3.1 PASCAL VOC Object Classification

Final Results. 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 [13]. 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 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 [27].

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

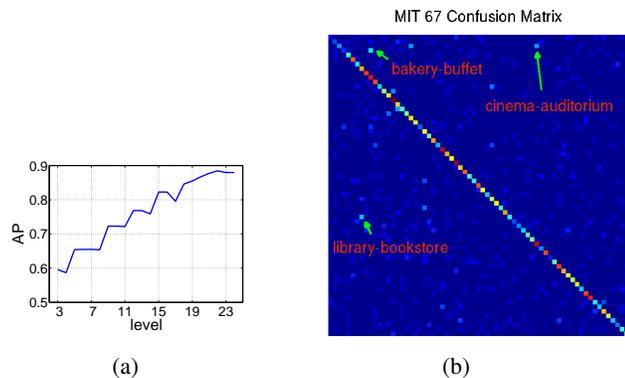


Figure 2: **a)** Evolution of the mean image classification AP over PASCAL VOC 2007 classes as we use a deeper representation from the OverFeat CNN trained on the ILSVRC 2011 dataset. **b)** Confusion matrix for the MIT-67 indoor dataset.

for all classes using the output of each network layer. The result is shown in Figure 2a. As can be seen except for the fully connected 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.

3.3.2 MIT 67 Scene Classification

Final Results. 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 representation off the shelf with linear SVMs training significantly outperforms majority of the baselines. The non-CNN baselines benefit from a broad range of sophisticated designs. [23] uses various object detectors on the LabelMe dataset and learns a sparse representation by fusing the results. [29] trains a discriminative large region based model which learns the distribution of appearances in each region/scene. The discriminative optimization is initialized with a generative approximation. The BoP approach [20] trains thousands of small part exemplar SVMs and expands a selection of those by retraining, it finally fuses them into a single representation and learns a new classifier on top of that. The improved Fisher Vectors (IFV) of [20] uses a $\sim 200,000$ dimensional representation derived from a GMM modeling of extracted SIFT patches. Finally, [10] uses a similar approach to [20] with an even more sophisticated selection and learning which takes several days to train.

Confusion Matrix. Figure 2b depicts the confusion matrix of the CNN-SVM classifier on the 67 MIT classes. It has a strong diagonal. The few relatively bright off-diagonal

Method	mean Accuracy
ROI + Gist[34]	26.05
DPM[28]	30.40
Object Bank[23]	37.60
RBow[29]	37.93
BoP[20]	46.10
miSVM[24]	46.40
D-Parts[38]	51.40
IFV[20]	60.77
MLrep[10]	64.03
CNN-SVM	58.44

Table 2: **MIT-67 indoor scenes dataset.** It should be noted the MLrep [10] takes weeks to train various part classifiers and the IFV feature vectors have dimensionality greater than 200,000.

points are annotated with their ground truth and estimated labels. One can see that in these examples these two labels can be challenging even for a human to distinguish between, especially for close-up views of the scenes.

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 interesting 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 [25], Caltech bird species [40], dog breeds [1], cooking activities [35], cats and dogs [30] 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.

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 [40] 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 and report different results with or without using the part annotations during evaluation. In this work we only use the bounding box annotation during training and testing.

Method	Part info	mean Accuracy
Sift+Color+SVM[40]	✗	17.31
CNN-SVM	✗	53.29
Pose pooling kernel[43]	✓	28.20
RF[42]	✓	19.20
DPD[44]	✓	50.98
Poof[5]	✓	56.78

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.

Oxford 102 flowers dataset [25] contains 102 categories. Each category contains 40 to 258 of images. The flowers appear in different scales, pose and lighting conditions. Furthermore, the dataset provides segmentation for all the images.

4.2. Method

There is a single label associated to each bird and flower image in this dataset and the standard evaluation procedure is by measuring the mean accuracy. This is aligned with that of MIT-67 indoor dataset so we use the same classification and training procedure as for the scene recognition experiments, section 3.2. The penalty cost used for the CUB dataset is $C=3$ and $C=8$ for the Oxford 102 flowers dataset. These parameters were optimized using a small subset of each dataset.

4.3. 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. We can see that the CNN representation outperforms the SIFT+Color [40] representation by a large margin. RF [42] uses a bag of decision trees trained on a bag of SIFT words. DPD [44] is a strongly supervised deformable part model using LBP and color features. Finally, POOF [5] adopts several discriminative learners for the same parts from each two different subcategories.

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.

5. Attribute Detection

An attribute within the context of computer vision is defined as some semantic or abstract quality which different

Method	mean Accuracy
HSV [25]	43.00
SIFT internal [25]	55.10
SIFT boundary [25]	32.00
HOG [25]	49.60
HSV+SIFTi+SIFTb+HOG(MKL) [25]	72.80
BOW(4000) [15]	65.50
SPM(4000) [15]	67.40
FLH(100) [15]	72.70
BiCos seg [7]	79.40
Dense HOG+Coding+Pooling[2] w/o seg	76.70
Seg+Dense HOG+Coding+Pooling[2]	80.66
CNN-SVM w/o seg	74.70

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

instances/categories share. Attributes can be category level (4-legged) or instance level (wearing glasses). Detection of instance-level attributes can be challenging due to the subtlety of their appearance. Attribute detection is important since it enables the description of unknown objects in unsupervised or weakly-supervised scenarios.

5.1. Datasets

We use two datasets to investigate the performance of CNN features in attribute detection. The first dataset is the UIUC 64 object attributes dataset [14]. 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”.

5.2. Method

Multiple labels (attributes) can be assigned to each training sample. Thus, we adopt a one-versus-all approach as explained in section 3.2. The penalty cost is set to $C=3$.

5.3. Results

Table 5 compares CNN features performance to state-of-the-art methods. The results are reported for cases of across and within categories attribute detection (refer to [14] for more details).

Table 6 reports the results of the detection of 9 human attributes on the H3D dataset including poselets and DPD [44]. 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[14]	83.40	-	73.00
Latent Model[41]	62.16	79.88	-
Sparse Representation[39]	89.60	90.20	-
att. based classification[22]	-	-	73.70
CNN-SVM	91.67	82.23	89.04

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

Method	male	long hair	glasses	hat	tshirt	long slvs	shorts	jeans	long pnts	mAP
Freq[6]	59.3	30.0	22.0	16.6	23.5	49.0	17.9	33.8	74.7	36.31
SPM[6]	68.1	40.0	25.9	35.3	30.6	58.0	31.4	39.5	84.3	45.91
Poselets[6]	82.4	72.5	55.6	60.1	51.2	74.2	45.5	54.7	90.3	65.18
DPD[44]	83.7	70.0	38.1	73.4	49.8	78.1	64.1	78.1	93.5	69.88
CNN-SVM	83.0	67.6	39.7	66.8	52.6	82.2	78.2	71.7	95.2	70.78

Table 6: **H3D Human Attributes dataset results.** CNN representation is extracted from the bounding box surround the person. All the other methods require the part annotations during training. The first row shows the performance of a random classifier.



Figure 3: Some nearest neighbor results for the Sculpture6k dataset. The leftmost column shows a query image while the subsequent columns show the nearest neighbors to the query image with increasing distance.

6. Retrieval

In this section we compare the CNN representation to the current state-of-the-art representations of retrieval including VLAD[4], BoW 200k, IFV[31] and BoB[3]. Unlike our CNN representation, all the above methods use dictionaries trained on the same datasets as they are tested on. To allow for a fair comparison between the methods, we only report results on the main pipelines and exclude preprocessing/post-processing methods like spatial re-ranking.

6.1. Datasets

To investigate the feasibility of using a CNN representation for image retrieval, we chose five common datasets in this area. They are:

Oxford5k buildings[32] This is a collection of 5063 photos gathered from flickr, used as a reference set, and 55 queries of different buildings. From an architectural standpoint the buildings in Oxford5k are very similar. Therefore we feel

that this dataset poses a big challenge to the CNN representation trained on ImageNet.

Paris6k buildings[33] 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 contains 6340 images of which 3170 are provided for training purposes only, another 3170 reference images and 70 query images. We experimented on this dataset to discover if the CNN representation is capable of modeling the global shape of an item rather than just local textures.

Concatenation of Oxford5k, Paris6k and Sculpture6k Using this dataset creates the challenge of simultaneously solving both textured and texture-less item retrieval.

For these four datasets we reported mAP as the measurement metric as described in [32].

Holidays dataset[17] This dataset contains 1491 images of which 500 are queries. The Holidays dataset contains images of different scenes, items and monuments. Unlike the first four datasets, this dataset provides us with a diverse image retrieval set. We reported mAP as the measurement metric for this dataset.

UKbench[26] This is composed of images of 2250 items each from 4 different viewpoints. The UKbench provides a good benchmark for viewpoint changes. We reported the performance over UKBench by the mean number of relevant images within the top four most similar photos.

6.2. Method

Similar to previous tasks we use the $L2$ normalized output of 22nd layer. For building and sculpture retrieval aspect ratio plays a vital role. In fact we found resizing query patches to square images decreased the mAP by more than 2 percent. Therefore, for each query patch we extract features of the smallest square containing it.

Spatial search The buildings can appear in different scales and locations. Therefore we employ a spatial pyramid search. To compute the representations of layer l of the spatial pyramid, we divide an image into l^2 overlapping patches of size $(2/l) \times (W, H)$ where W and H are the width and height of the image. Each patch in the pyramid is then resized to 221×221 . Hence each image r is represented by $L_r = \sum_{l=1}^{h_r} l^2$ different patches where h_r is the height of spatial pyramid. We compute the similarity between image r and image q as

$$S_{r,q} = \max_{\substack{1 \leq i \leq L_r \\ 1 \leq j \leq L_q}} \mathbf{f}_r^i \cdot \mathbf{f}_q^j \quad (2)$$

where \mathbf{f}_r^i is the feature representation for the i th patch in the spatial pyramid representation of image r , similarly for \mathbf{f}_q^j .



Figure 4: The four most similar false positive patches over the Oxford5k using the CNN-representation.

6.3. Results

The result of four different retrieval methods applied to 5 datasets and the concatenation of the Oxford5k, Paris6k and Sculpture6k are presented in table 7. Figure 4 shows the most similar false positive for the Oxford5k dataset. This gives an indication of why retrieval task for similar buildings can be difficult with a generic (not tuned) representation.

We combined Oxford5k, Paris6k and Sculp6k and applied all 180 queries. The CNN representation achieved a mAP of 60.01 while VLAD reached a mAP of 51.88 percent. VLAD’s performance falls below that of the CNN-representation in this test mainly because VLAD does not work for smooth item retrieval. A small technical detail is $L2$ normalization of each CNN feature dimension increases the mAP by approximately 1 percent over the datasets Oxford5k and Paris6k.

The CNN representation performs relatively poorly on the UKbench database as this dataset addresses the challenge of viewpoint change and CNN features are not invariant to in-plane rotations not present in the training data.

Spatial search increase the processing time and memory consumption by $O(h_q^3 \times h_r^3)$. Where h_q and h_r are the height of spatial pyramids over query and reference images respectively. Therefore, the height of the pyramid should be kept as small as possible. The query items in the UKbench and the Holidays datasets are centered and spatial search is not required in either query or reference image. On the other hand the items in the Oxford5k, Paris6k and Sculp6k reference images are not centered and searching through the reference images increases the performance of retrieval.

The difference between landmarks in the Oxford5k and Paris6k datasets are subtle. For example nuances of the window architecture are the most visual distinctive features for many buildings. Hence, intuitively spatial search over the query images also should increase the performance of retrieval over the aforementioned datasets. While the important structures in the Sculpture6k dataset are the global shape of the sculpture and small patches in a query image do not capture this information. Figure 5 shows the effect of spatial search over the Paris6k and Sculpture6k datasets.

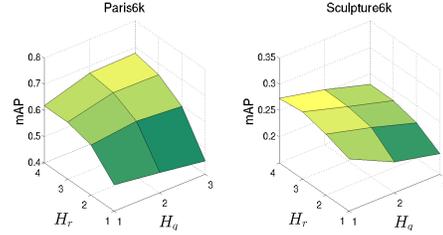


Figure 5: The effect of the spatial pyramid search for Paris6k and Sculp6k. As items in the reference sets are not centered, increasing the pyramid height for reference images constantly increases the performance. In Sculp6k complete shape of the sculpture matters. Therefore, spatial search in the query images decreases the performance. In Paris6k, a nuanced detail like the shape of a window in a building might be all the information we seek, hence increasing the height of pyramid increases the performance of retrieval.

	Oxford5k	Paris6k	Sculp6k	Holidays	Comb.	UKBench
VLAD 64D[4]	0.555 [4]	0.642	-	0.646 [4]	0.519	3.38
BoW 200kD	0.364[18]	0.460[33]	0.086[3]	0.540[4]	-	2.81[18]
IFV 64D[31]	0.418[4]	-	-	0.626[4]	-	3.35[18]
BoB	N/A	N/A	0.253[3]	N/A	-	N/A
CNN	0.520	0.676	0.269	0.646	0.600	3.05

Table 7: **The result of object retrieval on 6 datasets.** All the methods except the CNN have their representation trained on same dataset that they report the results on. The result of VLAD[4] on Oxford5k with a dictionary trained on Flicker60k is 0.478[4]. ”Comb” has the results for the first three datasets combined.

7. 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 to be a strong competitor for 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 extends the results reported in [11]. 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] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011.
- [9] Q. Chen, Z. Song, Y. Hua, Z. Huang, and S. Yan. Hierarchical matching with side information for image classification. In *CVPR*, 2012.
- [10] C. Doersch, A. Gupta, and A. A. Efros. Mid-level visual element discovery as discriminative mode seeking. In *NIPS*, 2013.
- [11] 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.
- [12] J. Dong, W. Xia, Q. Chen, J. Feng, Z. Huang, and S. Yan. Subcategory-aware object classification. In *CVPR*, 2013.
- [13] 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>.
- [14] A. Farhadi, I. Endres, D. Hoiem, and D. A. Forsyth. Describing objects by their attributes. In *CVPR*, 2009.
- [15] B. Fernando, E. Fromont, and T. Tuytelaars. Mining mid-level features for image classification. *International Journal of Computer Vision*, 2014.
- [16] 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.
- [17] H. Jegou, M. Douze, and C. Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *ECCV*, 2008.
- [18] 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.
- [19] T. Joachims. Svmstruct support vector machine for complex outputs.
- [20] M. Juneja, A. Vedaldi, C. V. Jawahar, and A. Zisserman. Blocks that shout: Distinctive parts for scene classification. In *CVPR*, 2013.
- [21] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.
- [22] 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.
- [23] 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.
- [24] Q. Li, J. Wu, and Z. Tu. Harvesting mid-level visual concepts from large-scale internet images. In *CVPR*, 2013.
- [25] 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.
- [26] D. Nistér and H. Stewénius. Scalable recognition with a vocabulary tree. In *CVPR*, 2006.
- [27] 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.
- [28] M. Pandey and S. Lazebnik. Scene recognition and weakly supervised object localization with deformable part-based models. In *ICCV*, 2011.
- [29] S. N. Parizi, J. G. Oberlin, and P. F. Felzenszwalb. Reconfigurable models for scene recognition. In *CVPR*, 2012.
- [30] O. M. Parkhi, A. Vedaldi, A. Zisserman, and C. V. Jawahar. Cats and dogs. In *CVPR*, 2012.
- [31] F. Perronnin, Y. Liu, J. Sánchez, and H. Poirier. Large-scale image retrieval with compressed fisher vectors. In *CVPR*, 2010.
- [32] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *CVPR*, 2007.
- [33] 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.
- [34] A. Quattoni and A. Torralba. Recognizing indoor scenes. In *CVPR*, 2009.
- [35] M. Rohrbach, S. Amin, M. Andriluka, and B. Schiele. A database for fine grained activity detection of cooking activities. In *CVPR*, 2012.
- [36] 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.
- [37] Z. Song, Q. Chen, Z. Huang, Y. Hua, and S. Yan. Contextualizing object detection and classification. In *CVPR*, 2011.
- [38] J. Sun and J. Ponce. Learning discriminative part detectors for image classification and cosegmentation. In *ICCV*, 2013.
- [39] G. Tsagkatakis and A. E. Savakis. Sparse representations and distance learning for attribute based category recognition. In *ECCV Workshops (1)*, pages 29–42, 2010.
- [40] 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.
- [41] Y. Wang and G. Mori. A discriminative latent model of object classes and attributes. In *ECCV*, 2010.
- [42] B. Yao, A. Khosla, and F.-F. Li. Combining randomization and discrimination for fine-grained image categorization. In *CVPR*, 2011.
- [43] N. Zhang, R. Farrell, and T. Darrell. Pose pooling kernels for sub-category recognition. In *CVPR*, 2012.
- [44] N. Zhang, R. Farrell, F. Iandola, and T. Darrell. Deformable part descriptors for fine-grained recognition and attribute prediction. In *ICCV*, 2013.