Is object localization for free? –
Weakly-supervised learning with convolutional neural networks

Maxime Oquab*
INRIA Paris, France

Léon Bottou†
MSR, New York, USA

Ivan Laptev*
INRIA, Paris, France

Josef Sivic*
INRIA, Paris, France

Abstract

Successful methods for visual object recognition typically rely on training datasets containing lots of richly annotated images. Detailed image annotation, e.g. by object bounding boxes, however, is both expensive and often subjective. We describe a weakly supervised convolutional neural network (CNN) for object classification that relies only on image-level labels, yet can learn from cluttered scenes containing multiple objects. We quantify its object classification and object location prediction performance on the Pascal VOC 2012 (20 object classes) and the much larger Microsoft COCO (80 object classes) datasets. We find that the network (i) outputs accurate image-level labels, (ii) predicts approximate locations (but not extents) of objects, and (iii) performs comparably to its fully-supervised counterparts using object bounding box annotation for training.

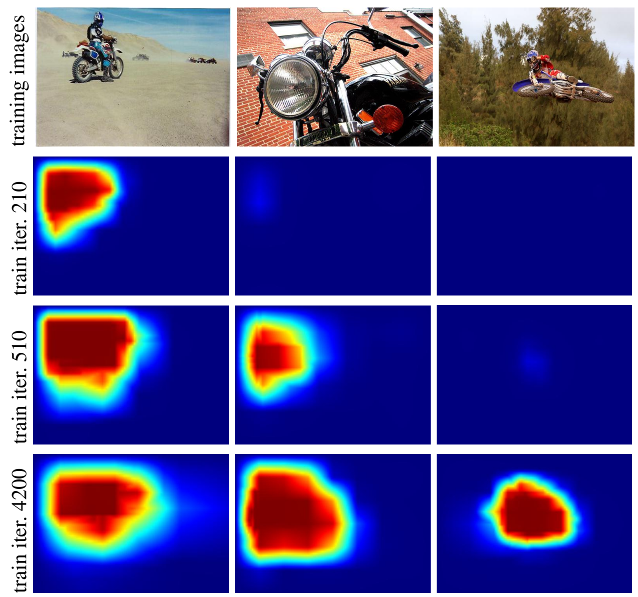

Figure 1: Evolution of localization score maps for the `motorbike` class over iterations of our weakly-supervised CNN training. Note that the network learns to localize objects despite having no object location annotation at training, just object presence/absence labels. Note also that locations of objects with more usual appearance (such as the motorbike shown in left column) are discovered earlier during training.

1. Introduction

Visual object recognition entails much more than determining whether the image contains instances of certain object categories. For example, each object has a location and a pose; each deformable object has a constellation of parts; and each object can be cropped or partially occluded.

Object recognition algorithms of the past decade can roughly be categorized in two styles. The first style extracts local image features (SIFT, HOG), constructs *bag of visual words* representations, and runs statistical classifiers [12, 41, 49, 61]. Although this approach has been shown to yield good performance for image classification, attempts to locate the objects using the position of the visual words have been unfruitful: the classifier often relies on visual words that fall in the background and merely describe the context of the object.

The second style of algorithms detects the presence of objects by fitting rich object models such as *deformable part models* [19, 59]. The fitting process can reveal useful attributes of objects such as location, pose and constellations of object parts, but the model is usually trained from images with known locations of objects or even their parts. The combination of both styles has shown benefits [25].

A third style of algorithms, *convolutional neural networks* (CNNs) [31, 33] construct successive feature vectors that progressively describe the properties of larger and larger image areas. Recent applications of this framework to natural images [30] have been extremely successful for a variety of tasks including image classification [6, 30, 37, 43, 44], object detection [22, 44], human pose estimation [52] and others. Most of these methods, however, require detailed image annotation. For example bounding box super-

*WILLOW project, Departement d'Informatique de l'École Normale Supérieure, ENS/INRIA/CNRS UMR 8548, Paris, France
†Léon Bottou is now with Facebook AI Research, New York.

vision has been shown highly benefitial for object classification in cluttered and complex scenes [37].

Labelling a set of training images with object attributes quickly becomes problematic. The process is expensive and involves a lot of subtle and possibly ambiguous decisions. For instance, consistently annotating locations and scales of objects by bounding boxes works well for some images but fails for partially occluded and cropped objects. Annotating object parts becomes even harder since the correspondence of parts among images in the same category is often ill-posed.

In this paper, we investigate whether CNNs can be trained from complex cluttered scenes labelled only with lists of objects they contain and not their locations. This is an extremely challenging task as the objects may appear at different locations, different scales and under variety of viewpoints, as illustrated in Figure 1 (top row). Furthermore, the network has to avoid overfitting to the scene clutter co-occurring with objects as, for example, motorbikes often appear on the road. How can we modify the structure of the CNN to learn from such difficult data?

We build on the successful CNN architecture [30] and the follow-up state-of-the-art results for object classification and detection [6, 22, 37, 43, 44], but introduce the following modifications. First, we treat the last fully connected network layers as convolutions to cope with the uncertainty in object localization. Second, we introduce a max-pooling layer that hypothesizes the possible location of the object in the image, similar to [32, Section 4] and [28]. Third, we modify the cost function to learn from image-level supervision. Interestingly, we find that this modified CNN architecture, while trained to output image-level labels only, localizes objects or their distinctive parts in training images, as illustrated in Figure 1. *So, is object localization with convolutional neural networks for free?* In this paper we set out to answer this question and analyze the developed weakly supervised CNN pipeline on two object recognition datasets containing complex cluttered scenes with multiple objects.

2. Related work

The fundamental challenge in visual recognition is modeling the intra-class appearance and shape variation of objects. For example, what is the appropriate model of the various appearances and shapes of "chairs"? This challenge is usually addressed by designing some form of a parametric model of the object's appearance and shape. The parameters of the model are then learnt from a set of instances using statistical machine learning. Learning methods for visual recognition can be characterized based on the required input supervision and the target output.

Unsupervised methods [34, 48] do not require any supervisory signal, just images. While unsupervised learning is appealing, the output is currently often limited only to frequently occurring and visually consistent objects. Fully supervised methods [18] require careful annotation of object location in the form of bounding boxes [18], segmentation [58] or even location of object parts [5], which is costly and can introduce biases. For example, should we annotate the dog's head or the entire dog? What if a part of the dog's body is occluded by another object? In this work, we focus on *weakly supervised* learning where only image-level labels indicating the presence or absence of objects are required. This is an important setup for many practical applications as (weak) image-level annotations are often readily available in large amounts, e.g. in the form of text tags [23], full sentences [38] or even geographical meta-data [15].

The target output in visual recognition ranges from image-level labels (object/image classification) [23], location and extent of objects in the form of bounding boxes (object detection) [18], to detailed object segmentation [5, 58, 24] or even predicting an approximate 3D pose and geometry of objects [26, 45]. In this work, we focus on predicting accurate image-level labels indicating the presence/absence of objects. However, we also find that the weakly supervised network can predict the approximate location (in the form of a x, y position) of objects in the scene, but not their extent (bounding box). Furthermore, our method performs on par with alternative fully-supervised methods both on the classification and location prediction tasks. We quantify these findings on the Pascal VOC 2012[17] and Microsoft COCO [36] datasets that both depict objects in complex cluttered scenes.

Initial work [2, 8, 11, 20, 57] on weakly supervised object localization has focused on learning from images containing prominent and centered objects in scenes with limited background clutter. More recent efforts attempt to learn from images containing multiple objects embedded in complex scenes [4, 13, 39, 50, 55, 9], from web images [7, 14] or from video [42]. These methods typically aim to localize objects including finding their extent in the form of bounding boxes. They attempt to find parts of images with visually consistent appearance in the training data that often contains multiple objects in different spatial configurations and cluttered backgrounds. While these works are promising, their performance is still far from the fully supervised methods such as [22, 44].

Our work is related to recent methods that find distinctive mid-level object parts for scene and object recognition in unsupervised [47] or weakly supervised [15, 27] settings. The proposed method can also be seen as a variant of Multiple Instance Learning [21, 29, 54] if we refer to each image as a "bag" and treat each image window as a "sample".

In contrast to the above methods we develop a weakly supervised learning method based on *end-to-end training* of a convolutional neural network (CNN) [31, 33] from image-level labels. Convolutional neural networks have recently demonstrated excellent performance on a number of visual recognition tasks that include classification of entire images [16, 30, 60], predicting presence/absence of objects in cluttered scenes [6, 37, 43, 44] or localizing objects by bounding boxes [22, 44]. However, most of the current CNN architectures assume in training a single prominent object in the image with limited background clutter [16, 30, 35, 44, 60] or require fully annotated object locations in the image [22, 37, 44]. Learning from images containing multiple objects in cluttered scenes with only weak object presence/absence labels has been so far mostly limited to representing entire images without explicitly searching for location of individual objects [6, 43, 60], though some level of robustness to the scale and position of objects is gained by jittering. Recent concurrent effort [56] also investigates CNNs for learning from weakly labelled cluttered scenes. Their work confirms some of our findings but does not investigate location prediction. Our work is also related to recent efforts aiming to extract object localization by examining the network output while masking different portions of the input image [3, 46, 60, 62], but these methods consider already pre-trained networks at test time.

Contributions. The contributions of this work are twofold. First, we develop a weakly supervised convolutional neural network end-to-end learning pipeline that learns from complex cluttered scenes containing multiple objects by explicitly searching over possible object locations and scales in the image. Second, we perform an extensive experimental analysis of the network's classification and localization performance on the Pascal VOC 2012 and the much larger Microsoft COCO datasets. We find that our weakly-supervised network (i) outputs accurate image-level labels, (ii) predicts approximate locations (but not extents) of objects, and (iii) performs comparably to its fully-supervised counterparts that use object bounding box annotation for training.

3. Network architecture for weakly supervised learning

We build on the fully supervised network architecture of [37] that consists of five convolutional and four fully connected layers and assumes as input a fixed-size image patch containing a single relatively tightly cropped object. To adapt this architecture to weakly supervised learning we introduce the following three modifications. First, we treat the fully connected layers as convolutions, which allows us to deal with nearly arbitrary-sized images as input. Second, we explicitly search for the highest scoring object position in the image by adding a single global max-pooling layer at the output. Third, we use a cost function that can explicitly model multiple objects present in the image. The three modifications are discussed next and the network architecture is illustrated in Figure 2.

Convolutional adaptation layers. The network architecture of [37] assumes a fixed-size image patch of 224×224

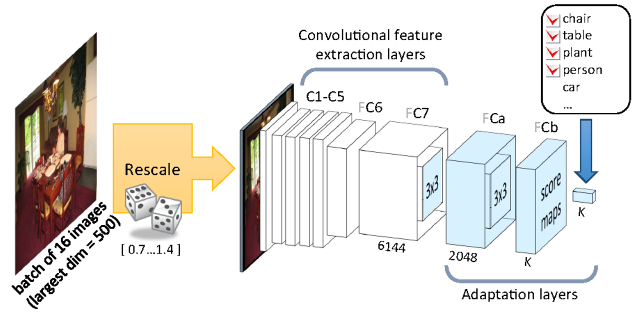

Figure 2: Network architecture for weakly supervised training.

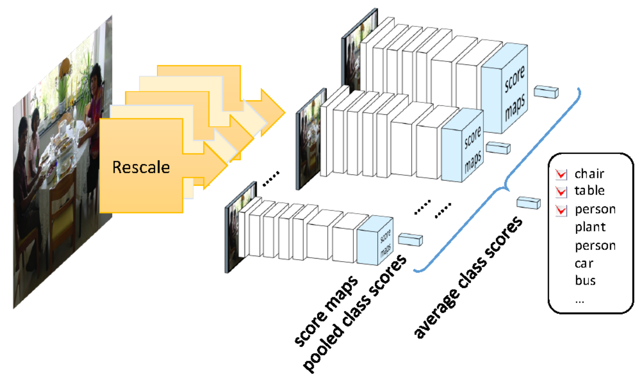

Figure 3: Multiscale object recognition.

RGB pixels as input and outputs a $1 \times 1 \times K$ vector of per-class scores as output, where K is the number of classes. The aim is to apply the network to bigger images in a sliding window manner thus extending its output to $n \times m \times K$ where n and m denote the number of sliding window positions in the x- and y- direction in the image, respectively, computing the K per-class scores at all input window positions. While this type of sliding was performed in [37] by applying the network to independently extracted image patches, here we achieve the same effect by treating the fully connected adaptation layers as convolutions. For a given input image size, the fully connected layer can be seen as a special case of a convolution layer where the size of the kernel is equal to the size of the layer input. With this procedure the output of the final adaptation layer FC7 becomes a $2 \times 2 \times K$ output score map for a 256×256 RGB input image. As the global stride of the network is 32[1] pixels, adding 32 pixels to the image width or height increases the width or height of the output score map by one. Hence, for example, a 2048×1024 pixel input would lead to a 58×26 output score map containing the score of the network for all classes for the different locations of the input 224×224 window with a stride of 32 pixels. While this architecture is typically used for efficient classification at test time, see e.g. [44], here we also use it at training time (as discussed in Section 4) to efficiently examine the entire

[1] or 36 pixels for the OverFeat network that we use on MS COCO

image for possible locations of the object during weakly supervised training.

Explicit search for object's position via max-pooling. The aim is to output a single image-level score for each of the object classes independently of the input image size. This is achieved by aggregating the $n \times m \times K$ matrix of output scores for $n \times m$ different positions of the input window using a global max-pooling operation into a single $1 \times 1 \times K$ vector, where K is the number of classes. Note that the max-pooling operation effectively searches for the best-scoring candidate object position within the image, which is crucial for weakly supervised learning where the exact position of the object within the image is not given at training. In addition, due to the max-pooling operation the output of the network becomes independent of the size of the input image, which will be used for multi-scale learning in Section 4.

Multi-label classification loss function. The goal of object classification is to tell whether an instance of an object class is present in the image, where the input image may depict multiple different objects. As a result, the usual multi-class mutually exclusive logistic regression loss, as used in e.g. [30] for ImageNet classification, is not suited for this set-up as it assumes only a single object per image. To address this issue, we treat the task as a separate binary classification problem for each class. The loss function is therefore a sum of K binary logistic regression losses, one for each of the K classes $k \in \{1 \cdots \mathrm{K}\}$,

$$\ell(\, f_k(\mathbf{x})\,,\, y_k \,) \;=\; \sum_k \log(1 + e^{-y_k f_k(\mathbf{x})})\,, \qquad (1)$$

where $f_k(\mathbf{x})$ is the output of the network for input image $\mathbf{x}$ and $y_k \in \{-1, 1\}$ is the image label indicating the absence/presence of class k in the input image $\mathbf{x}$. Each class score $f_k(\mathbf{x})$ can be interpreted as a posterior probability indicating the presence of class k in image $\mathbf{x}$ with transformation

$$P(k|\mathbf{x}) \approx \frac{1}{1 + e^{-f_k(\mathbf{x})}}\;. \qquad (2)$$

Treating a multi-label classification problem as K independent classification problems is often inadequate because it does not model label correlations. This is not an issue here because the classifiers share hidden layers and therefore are not independent. Such a network can model label correlations by tuning the overlap of the hidden state distribution given each label.

4. Weakly supervised learning and classification

In this section we describe details of the training procedure. Similar to [37] we pre-train the convolutional feature extraction layers C1-C7 on images from the ImageNet dataset and keep their weights fixed. This pre-training procedure is standard and similar to [30]. Next, the goal is to train the adaptation layers Ca and Cb using the Pascal VOC or MS COCO images in a weakly supervised manner, i.e. from image-level labels indicating the presence/absence of the object in the image, but not telling the actual position and scale of the object. This is achieved by stochastic gradient descent training using the network architecture and cost function described in Section 3, which explicitly searches for the best candidate position of the object in the image using the global max-pooling operation. We also search over object scales (similar to [40]) by training from images of different sizes. The training procedure is illustrated in Figure 2. Details and further discussion are given next.

Stochastic gradient descent with global max-pooling. The global max-pooling operation ensures that the training error backpropagates only to the network weights corresponding to the highest-scoring window in the image. In other words, the max-pooling operation hypothesizes the location of the object in the image at the position with the maximum score, as illustrated in Figure 4. If the image-level label is positive (i.e. the image contains the object) the back-propagated error will adapt the network weights so that the score of this particular window (and hence other similar-looking windows in the dataset) is increased. On the other hand, if the image-level label is negative (i.e. the image does not contain the object) the back-propagated error adapts the network weights so that the score of the highest-scoring window (and hence other similar-looking windows in the dataset) is decreased. For negative images, the max-pooling operation acts in a similar manner to hard-negative mining known to work well in training sliding window object detectors [18]. Note that there is no guarantee the location of the score maxima corresponds to the true location of the object in the image. However, the intuition is that the erroneous weight updates from the incorrectly localized objects will only have limited effect as in general they should not be consistent over the dataset.

Multi-scale sliding-window training. The above procedure assumes that the object scale (the size in pixels) is known and the input image is rescaled so that the object occupies an area that corresponds to the receptive field of the fully connected network layers (i.e. 224 pixels). In general, however, the actual object size in the image is unknown. In fact, a single image can contain several different objects of different sizes. One possible solution would be to run multiple parallel networks for different image scales that share parameters and max-pool their outputs. We opt for a different less memory demanding solution. Instead, we train from images rescaled to multiple different sizes. The intuition is that if the object appears at the correct scale, the max-pooling operation correctly localizes the object in the image and correctly updates the network weights. When the

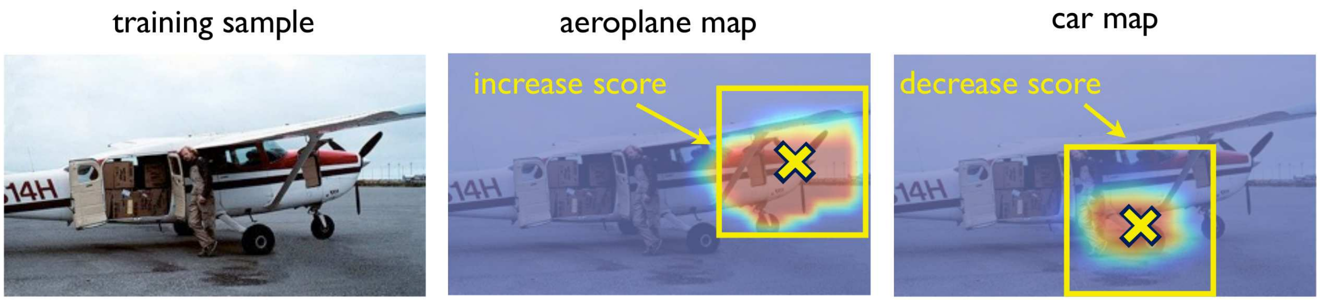

Figure 4: **Illustration of the weakly-supervised learning procedure.** At training time, given an input image with an aeroplane label (left), our method increases the score of the highest scoring positive image window (middle), and decreases scores of the highest scoring negative windows, such as the one for the car class (right).

object appears at the wrong scale the location of the maximum score may be incorrect. As discussed above, the network weight updates from incorrectly localized objects may only have limited negative effect on the results in practice.

In detail, all training images are first rescaled to have the largest side of size 500 pixels and zero-padded to 500×500 pixels. Each training mini-batch of 16 images is then resized by a scale factor s uniformly sampled between 0.7 and 1.4. This allows the network to see objects in the image at various scales. In addition, this type of multi-scale training also induces some scale-invariance in the network.

Classification. At test time we apply the same sliding window procedure at multiple finely sampled scales. In detail, the test image is first normalized to have its largest dimension equal to 500 pixels, padded by zeros to 500×500 pixels and then rescaled by a factor $s \in \{0.5, 0.7, 1, 1.4, 2.0, 2.8\}$. Scanning the image at large scales allows the network to find even very small objects. For each scale, the per-class scores are computed for all window positions and then max-pooled across the image. These raw per-class scores (before applying the soft-max function (2)) are then aggregated across all scales by averaging them into a single vector of per-class scores. The testing architecture is illustrated in Figure 3. We found that searching over only six different scales at test time was sufficient to achieve good classification performance. Adding wider or finer search over scale did not bring additional benefits.

5. Classification experiments

In this section we describe our classification experiments where we wish to predict whether the object is present / absent in the image. Predicting the location of the object is evaluated in section 6.

Experimental setup. We apply the proposed method to the Pascal VOC 2012 object classification task and the recently released Microsoft COCO dataset. The Pascal VOC 2012 dataset contains 5k images for training, 5k for validation and 20 object classes. The much larger COCO dataset contains 80k images for training, 40k images for validation and 80 classes. On the COCO dataset, we wish to evaluate whether our method scales-up to much bigger data with more classes.

We use Torch7 [10] for our experiments. For Pascal VOC, we use a network pre-trained on 1512 classes of ImageNet following [37] ; for COCO, we use the Overfeat [44] network. Training the adaptation layers was performed with stochastic gradient descent (learning rate 0.001, momentum 0.9).

Pascal VOC 2012 classification results. In Table 1, we provide classification scores on the Pascal VOC 2012 test set, for which many baseline results are available. Evaluation is performed via the Pascal VOC evaluation server. The per-class performance is measured using average precision (the area under the precision-recall curve) and summarized across all classes using mean average precision (mAP). Our weakly supervised approach (G.WEAK SUP) obtains the highest overall mAP among all single network methods outperforming other CNN-based methods trained from image-level supervision (C-G) as well as the comparable setup of [37] (B) that uses object-level supervision.

Benefits of sliding-window training. Here we compare the proposed weakly supervised method (G. WEAK SUP) with training from full images (F. FULL IMAGES), where no search for object location during training/testing is performed and images are presented to the network at a single scale. Otherwise the network architectures are identical. Results for Pascal VOC test data are shown in Table 1). The results clearly demonstrate the benefits of sliding window multi-scale training attempting to localize the objects in the training data. The largest improvements are obtained for small objects, such as bottles and potted plants, where AP increases by 15-20%. Similar results on the COCO dataset are shown in the first row of Figure 5, where sliding window weakly supervised training (blue) consistently improves over the full image training (red) for all classes.

Benefits of multi-scale training and testing. On the COCO dataset, multi-scale training improves the classification mAP by about 1% when compared to training at a single-scale $s = 1$. The intuition is that the network gets to

Object-level sup.	plane	bike	bird	boat	btl	bus	car	cat	chair	cow	table	dog	horse	moto	pers	plant	sheep	sofa	train	tv	**mAP**
A.NUS-SCM [51]	97.3	84.2	80.8	85.3	60.8	89.9	86.8	89.3	**75.4**	77.8	75.1	83.0	87.5	90.1	95.0	57.8	79.2	**73.4**	**94.5**	80.7	82.2
B.OQUAB [37]	94.6	82.9	88.2	84.1	60.3	89.0	84.4	90.7	72.1	86.8	69.0	92.1	93.4	88.6	96.1	64.3	86.6	62.3	91.1	79.8	82.8
Image-level sup.	plane	bike	bird	boat	btl	bus	car	cat	chair	cow	table	dog	horse	moto	pers	plant	sheep	sofa	train	tv	**mAP**
C.Z&F [60]	96.0	77.1	88.4	85.5	55.8	85.8	78.6	91.2	65.0	74.4	67.7	87.8	86.0	85.1	90.9	52.2	83.6	61.1	91.8	76.1	79.0
D.CHATFIELD [6]	96.8	82.5	91.5	88.1	62.1	88.3	81.9	**94.8**	70.3	80.2	76.2	92.9	90.3	89.3	95.2	57.4	83.6	66.4	93.5	81.9	83.2
E.NUS-HCP [56]	**97.5**	84.3	**93.0**	**89.4**	62.5	90.2	84.6	**94.8**	69.7	**90.2**	74.1	93.4	93.7	88.8	93.2	59.7	90.3	61.8	94.4	78.0	84.2
F.FULL IMAGES	95.3	77.4	85.6	83.1	49.9	86.7	77.7	87.2	67.1	79.4	73.5	85.3	90.3	85.6	92.7	47.8	81.5	63.4	91.4	74.1	78.7
G.WEAK SUP	96.7	**88.8**	92.0	87.4	**64.7**	**91.1**	**87.4**	94.4	74.9	89.2	**76.3**	**93.7**	**95.2**	**91.1**	**97.6**	**66.2**	**91.2**	70.0	**94.5**	**83.7**	**86.3**

Table 1: Single method image classification results on the VOC 2012 test set. Methods A,B use object-level supervision. Methods C to G use image-level supervision only. The combination of methods A and E reaches 90.3% mAP [56], the highest reported result on this data.

Setup	Classification		Location Prediction	
Dataset	VOC	COCO	VOC	COCO
H.FULL IMAGES	76.0	51.0	-	-
I.MASKED POOL	**82.3**	62.1	72.3	**42.9**
J.WEAK SUP	81.8	**62.8**	74.5	41.2
K.CENTER PRED.	-	-	50.9	19.1
L.RCNN*	79.2	-	**74.8**	-

Table 2: Classification and location prediction mean Average Precision on the validation sets for Pascal VOC and COCO datasets. *For R-CNN[22], which is an algorithm designed for *object detection*, we use only the most confident bounding box proposal per class and per image for evaluation.

see objects at different scales, increasing the overall number of examples. Scanning at multiple scales at test time provides an additional 3% increase in classification mAP.

Does adding object-level supervision help classification? Here we investigate whether adding object-level supervision to our weakly supervised setup improves classification performance. In order to test this, we remove the global max-pooling layer in our model and introduce a "masked pooling" layer that indicates the location of individual objects during training. In detail, the masked pooling layer uses ground truth maps of the same size as the output of the network, signaling the presence or absence of an object class to perform the global max-pooling, but now restricted to the relevant area of the output. This provides learning guidance to the network as the max-scoring object hypothesis has to lie within the ground truth object location in the image. We have also explored a variant of this method, that minimized the object score outside of the masked area to avoid learning from the context of the object, but obtained consistently worse results. Classification results for the masked-pooling method (I. MASKED POOL) on both the Pascal VOC and COCO datasets are provided in Table 2 and show that adding this form of object-level supervision does not bring significant benefits over the weakly-supervised learning.

6. Location prediction experiments

The proposed weakly supervised architecture outputs score maps for different objects. In the previous section we have shown that max-pooling on these maps provides excellent classification performance. However, we have also observed that these scores maps are consistent with the locations of objects in the input images. In this section we investigate whether the output score maps can be used to localize the objects.

Location prediction metric. In order to provide quantitative evaluation of the localization power of our CNN architecture, we introduce a simple metric based on precision-recall using the per-class response maps. We first rescale the maps to the original image size[2]. If the maximal response across scales falls within the ground truth bounding box of an object of the same class within 18 pixels tolerance (which corresponds to the pooling ratio of the network), we label the predicted location as correct. If not, then we count the response as a false positive (it hit the background), and we also increment the false negative count (no object was found). Finally, we use the confidence values of the responses to generate precision-recall curves. Each p-r curve is summarized by Average Precision (AP). The perfect performance (AP=1) means that the network has indicated the presence / absence of the object correctly in all images and for each image containing the object the predicted object location fell inside one of the ground truth bounding boxes of that object (if multiple object instances were present). This metric differs from the standard object detection bounding box overlap metric as it does not take into account whether the extent of the object is predicted correctly and it only measures localization performance for one object instance per image. Note however, that even this type of location prediction is very hard for complex cluttered scenes considered in this work.

Location prediction results. The summary of the location prediction results for both the Pascal VOC and Microsoft COCO datasets is given in Table 2. The per-class results for the Pascal VOC and Microsoft COCO datasets, are shown in Table 3 (J.WEAK SUP) and Figure 5 (green bars), respectively.

Center prediction baseline. We compare the location prediction performance to the following baseline. We use the max-pooled image-level per-class scores of our weakly supervised setup (J.WEAK SUP), but predict the center of the image as the location of the object. As shown in Table 2,

[2] We do simple interpolation in our experiments.

	plane	bike	bird	boat	btl	bus	car	cat	chair	cow	table	dog	horse	moto	pers	plant	sheep	sofa	train	tv	**mAP**
I.MASKED POOL	89.0	76.9	**83.2**	68.3	39.8	**88.1**	62.2	90.2	47.1	83.5	40.2	88.5	**93.7**	83.9	84.6	44.2	80.6	**51.9**	**86.8**	64.1	72.3
J.WEAK SUP	90.3	77.4	81.4	**79.2**	41.1	87.8	66.4	**91.0**	47.3	**83.7**	55.1	**88.8**	93.6	85.2	87.4	43.5	**86.2**	50.8	**86.8**	66.5	74.5
K.CENTER PRED.	78.9	55.0	61.1	38.9	14.5	78.2	30.7	82.6	17.8	65.4	17.2	70.3	80.1	65.9	58.9	18.9	63.8	28.5	71.8	22.4	51.0
L.RCNN*	**92.0**	**80.8**	80.8	73.0	**49.9**	86.8	**77.7**	87.6	**50.4**	72.1	**57.6**	82.9	79.1	**89.8**	**88.1**	**56.1**	83.5	50.1	81.5	**76.6**	**74.8**

Table 3: Location prediction scores on the VOC12 validation set. Maximal responses are labeled as correct when they fall within a bounding box of the same class, and count as false negatives if the class was present but its location was not predicted. We then use the confidence values of the responses to generate precision-recall values.

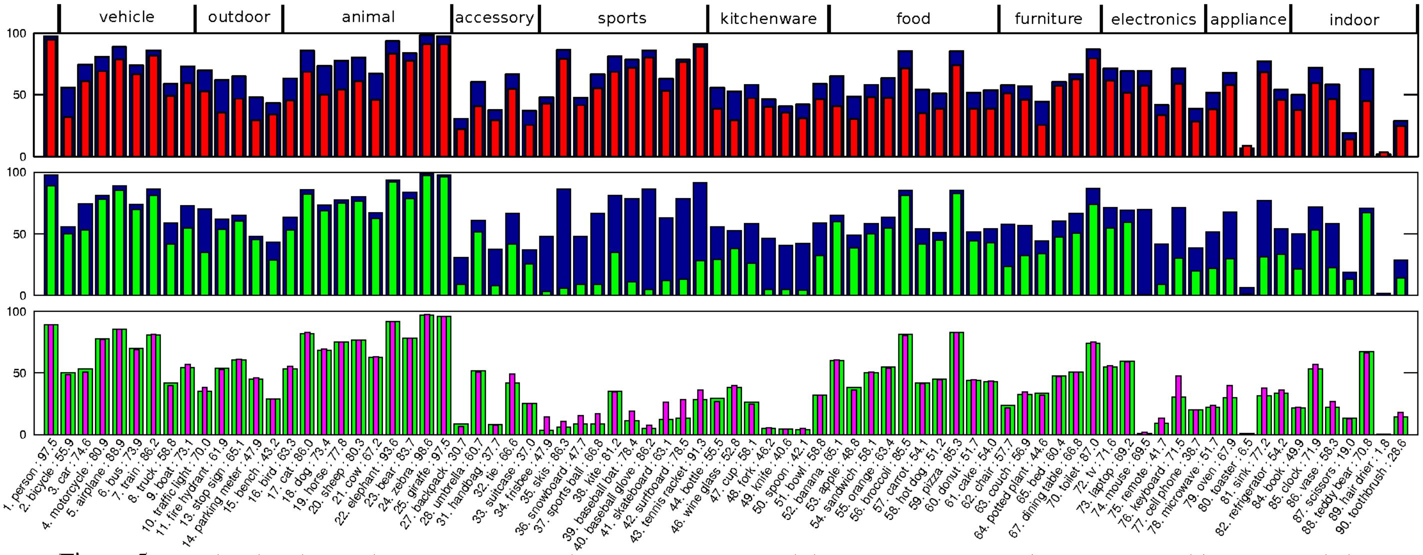

Figure 5: Per-class barplots of the output scores on the Microsoft COCO validation set. From top to bottom : (a) weakly-supervised classification AP (blue) vs. full-image classification AP (red). (b) weakly-supervised classification AP (blue) vs. weakly-supervised location prediction AP (green). (c) weakly-supervised location prediction AP (green) vs. masked-pooling location prediction AP (magenta). At the bottom of the figure, we provide the object names and weakly-supervised classification AP values.

using the center prediction baseline (K.CENTER PRED.) results in a >50% performance drop on COCO, and >30% drop on Pascal VOC, compared to our weakly supervised method (J.WEAK SUP) indicating the difficulty of the location prediction task on this data.

Comparison with R-CNN baseline. In order to provide a baseline for the location prediction task, we used the bounding box proposals and confidence values obtained with the state-of-the-art object detection R-CNN [22] algorithm on the Pascal VOC 2012 validation set. Note that this algorithm was not designed for classification, and its goal is to find all the objects in an image, while our algorithm looks only for a single instance of a given object class. To make the comparison as fair as possible, we process the R-CNN results to be compatible with our metric, keeping for each class and image only the best-scoring bounding box proposal and using the center of the bounding box for evaluation. Results are summarized in Table 2 and the detailed per-class results are shown in Table 3. Interestingly, our weakly supervised method (J.WEAK SUP) achieves comparable location prediction performance to the strong R-CNN baseline, which uses object bounding boxes at training time.

Does adding object-level supervision help location prediction? Here we investigate whether adding the object-level supervision (with masked pooling) helps to better predict the locations of objects in the image. The results on the Pascal VOC dataset are shown in Table 3 and show a very similar overall performance for our weakly supervised (J.WEAK SUP) method compared to the object-level supervised (I.MASKED POOL) setup. This is interesting as it indicates that our weakly supervised method learns to predict object locations and adding object-level supervision does not significantly increase the overall location prediction performance. Results on the COCO dataset are shown in Figure 5 (bottom) and indicate that for some classes with poor location prediction performance in the weakly supervised setup (green) adding object-level supervision (masked pooling, magenta) helps. Examples are small sports objects such as frisbee, tennis racket, baseball bat, snowboard, sports ball, or skis. While for classification the likely presence of these objects can be inferred from the scene context, object-level supervision can help to understand better the underlying concept and predict the object location in the image. We examine the importance of the object context next.

The importance of object context. To better assess the importance of object context for the COCO dataset we directly compare the classification (blue) and location prediction (green) scores in Figure 5 (middle). In this setup a high classification score but low location prediction score means

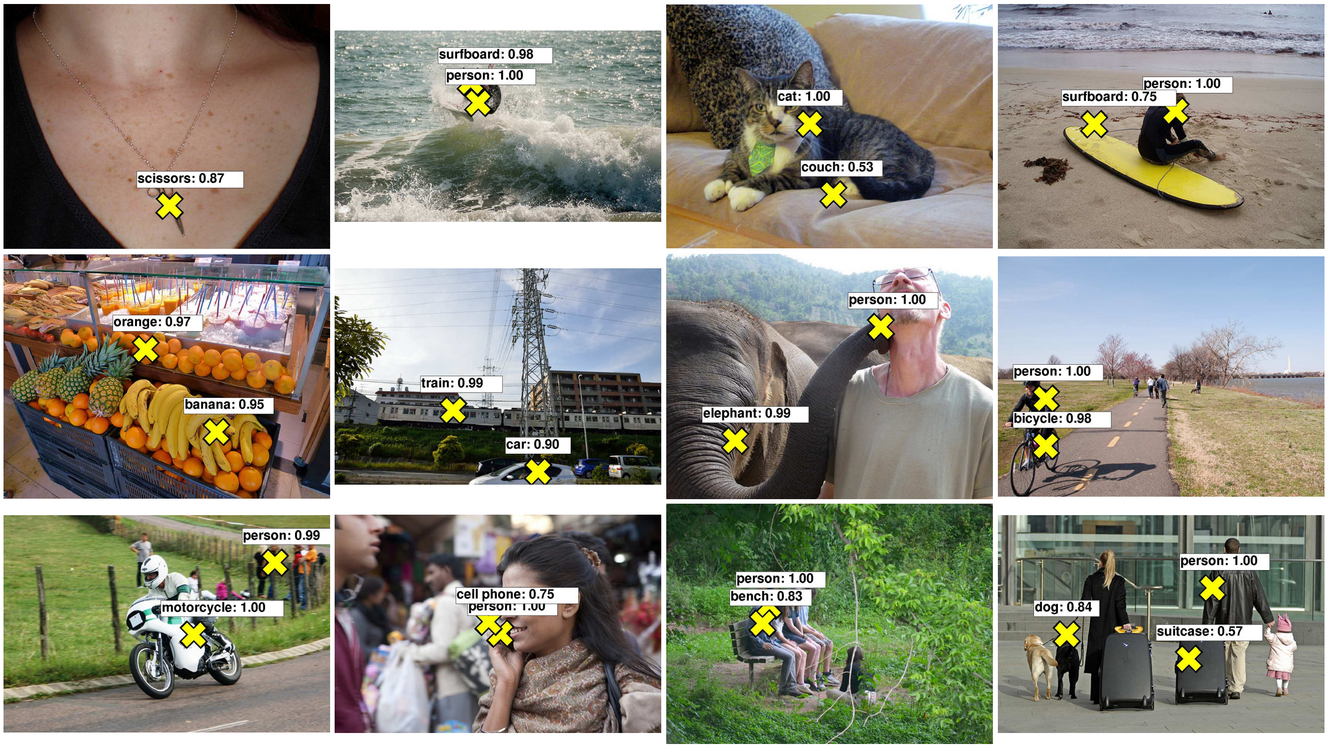

Figure 6: Example location predictions for images from the Microsoft COCO validation set obtained by our weakly-supervised method. Note that our method does not use object locations at training time, yet can predict locations of objects in test images (yellow crosses). The method outputs the most confident location per object per class. **Please see additional results on the project webpage[1].**

that the classification decision was taken primarily based on the object context. Fore example, the presence of a baseball field is a strong indicator for presence of a baseball bat and a baseball glove. However, as discussed above these objects are hard to localize in the image. The kitchenware (forks, knives, spoons) and electronics (laptop, keyboard, mouse) superclasses show a similar behavior. Nevertheless, a good classification result can still be informative and can guide a more precise search for these objects in the image.

Predicting extent of objects. To evaluate the ability to predict the extent of objects (not just the location) we also evaluate our method using the standard area overlap ratio as used in object detection [17]. We have implemented a simple extension of our method that aggregates CNN scores within selective search [53] object proposals. This procedure obtains on the Pascal VOC 2012 validation set the mAP of 11.74, 27.47, 43.54% for area overlap thresholds 0.5, 0.3, 0.1, respectively. The relatively low performance could be attributed to (a) the focus of the network on discriminative object parts (e.g. aeroplane propeller, as in Figure 4) rather than the entire extent of an object and (b) no max-pooling over scales in our current training procedure. Similar behavior on discriminative parts was recently observed in scene classification [62].

7. Conclusion

So, is object localization with convolutional neural networks for free? We have shown that our weakly supervised CNN architecture learns to predict the location of objects in images despite being trained from cluttered scenes with only weak image-level labels. We believe this is possible because of (i) the hierarchical convolutional structure of CNNs that appears to have a bias towards spatial localization combined with (ii) the extremely efficient end-to-end training that back-propagates loss gradients from image-level labels to candidate object locations. While the approximate position of objects can be predicted rather reliably, this is not true (at least with the current architecture) for the extent of objects as the network tends to focus on distinctive object parts. However, we believe our results are significant as they open-up the possibility of large-scale reasoning about object relations and extents without the need for detailed object level annotations.

Acknowledgements. This work was supported by the MSR-INRIA laboratory, ERC grant Activia (no. 307574), ERC grant Leap (no. 336845) and the ANR project Semapolis (ANR-13-CORD-0003).

References

[1] http://www.di.ens.fr/willow/research/weakcnn/, 2014. 8

[2] H. Arora, N. Loeff, D. Forsyth, and N. Ahuja. Unsupervised segmentation of objects using efficient learning. In *CVPR*, 2007. 2

[3] A. Bergamo, L. Bazzani, D. Anguelov, and L. Torresani. Self-taught object localization with deep networks. *CoRR*, abs/1409.3964, 2014. 3

[4] M. Blaschko, A. Vedaldi, and A. Zisserman. Simultaneous object detection and ranking with weak supervision. In *NIPS*, 2010. 2

[5] T. Brox, L. Bourdev, S. Maji, and J. Malik. Object segmentation by alignment of poselet activations to image contours. In *CVPR*, 2011. 2

[6] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. *arXiv:1405.3531v2*, 2014. 1, 2, 3, 6

[7] X. Chen, A. Shrivastava, and A. Gupta. Neil: Extracting visual knowledge from web data. In *ICCV*, 2013. 2

[8] O. Chum and A. Zisserman. An exemplar model for learning object classes. In *CVPR*, 2007. 2

[9] R. G. Cinbis, J. Verbeek, and C. Schmid. Weakly Supervised Object Localization with Multi-fold Multiple Instance Learning. Mar 2015. 2

[10] R. Collobert, K. Kavukcuoglu, and C. Farabet. Torch7: A matlab-like environment for machine learning. In *BigLearn, NIPS Workshop*, 2011. 5

[11] D. Crandall and D. Huttenlocher. Weakly supervised learning of part-based spatial models for visual object recognition. In *ECCV*, 2006. 2

[12] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *ECCV Workshop*, 2004. 1

[13] T. Deselaers, B. Alexe, and V. Ferrari. Localizing objects while learning their appearance. In *ECCV*, 2010. 2

[14] S. Divvala, A. Farhadi, and C. Guestrin. Learning everything about anything: Webly-supervised visual concept learning. In *CVPR*, 2014. 2

[15] C. Doersch, S. Singh, A. Gupta, J. Sivic, and A.A. Efros. What makes Paris look like Paris? *ACM TOG*, 31(4):101, 2012. 2

[16] 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. *arXiv:1310.1531*, 2013. 3

[17] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (VOC) challenge. *IJCV*, 88(2):303–338, Jun 2010. 2, 8

[18] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *IEEE PAMI*, 32(9):1627–1645, 2010. 2, 4

[19] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *CVPR*, 2008. 1

[20] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *CVPR*, 2003. 2

[21] J. Foulds and E. Frank. A review of multi-instance learning assumptions. *The Knowledge Engineering Review*, 25(01):1–25, 2010. 2

[22] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 1, 2, 3, 6, 7

[23] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid. Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation. In *CVPR*, 2009. 2

[24] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Simultaneous detection and segmentation. In *ECCV*, 2014. 2

[25] H. Harzallah, F. Jurie, and C. Schmid. Combining efficient object localization and image classification. In *CVPR*, 2009. 1

[26] M. Hejrati and D. Ramanan. Analyzing 3d objects in cluttered images. In *NIPS*, 2012. 2

[27] M. Juneja, A. Vedaldi, C. V. Jawahar, and A. Zisserman. Blocks that shout: Distinctive parts for scene classification. In *CVPR*, 2013. 2

[28] J. D. Keeler, D. E. Rumelhart, and W. K. Leow. Integrated segmentation and recognition of hand-printed numerals. In *NIPS*, 1991. 2

[29] D. Kotzias, M. Denil, P. Blunsom, and N. de Freitas. Deep multi-instance transfer learning. *CoRR*, abs/1411.3128, 2014. 2

[30] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 1, 2, 3, 4

[31] K.J. Lang and G.E. Hinton. A time delay neural network architecture for speech recognition. Technical Report CMU-CS-88-152, CMU, 1988. 1, 2

[32] K.J. Lang, A.H. Waibel, and G.E. Hinton. A time-delay neural network architecture for isolated word recognition. *Neural networks*, 3(1):23–43, 1990. 2

[33] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R.E. Howard, W. Hubbard, and L.D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4):541–551, Winter 1989. 1, 2

[34] Y. J. Lee and K. Grauman. Learning the easy things first: Self-paced visual category discovery. In *CVPR*, 2011. 2

[35] M. Lin, Q. Chen, and S. Yan. Network in network. *arXiv:1312.4400v3*, 2014. 3

[36] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 2

[37] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *CVPR*, 2014. 1, 2, 3, 4, 5, 6

[38] V. Ordonez, G. Kulkarni, and T. Berg. Im2text: Describing images using 1 million captioned photographs. In *NIPS*, 2011. 2

[39] M. Pandey and S. Lazebnik. Scene recognition and weakly supervised object localization with deformable part-based models. In *ICCV*, 2011. 2

[40] G. Papandreou, I. Kokkinos, and P.-A. Savalle. Untangling Local and Global Deformations in Deep Convolutional Networks for Image Classification and Sliding Window Detection. In *CVPR*, 2015. 4

[41] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *ECCV*, 2010. 1

[42] A. Prest, C. Leistner, J. Civera, C. Schmid, and V. Ferrari. Learning object class detectors from weakly annotated video. In *CVPR*, 2012. 2

[43] A. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. CNN features off-the-shelf: an astounding baseline for recognition. *arXiv preprint arXiv:1403.6382*, 2014. 1, 2, 3

[44] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv:1312.6229*, 2013. 1, 2, 3, 5

[45] A. Shrivastava and A. Gupta. Building part-based object detectors via 3d geometry. In *ICCV*, 2013. 2

[46] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *CoRR*, abs/1312.6034, 2013. 3

[47] S. Singh, A. Gupta, and A. A. Efros. Unsupervised discovery of mid-level discriminative patches. In *ECCV*, 2012. 2

[48] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman. Discovering object categories in image collections. In *ICCV*, 2005. 2

[49] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *ICCV*, 2003. 1

[50] H. Song, R. Girshick, S. Jegelka, J. Mairal, Z. Harchaoui, and T. Darrell. On learning to localize objects with minimal supervision. In *ICML*, 2014. 2

[51] Z. Song, Q. Chen, Z. Huang, Y. Hua, and S. Yan. Contextualizing object detection and classification. In *CVPR*, 2011. 6

[52] A. Toshev and C. Szegedy. Deeppose: Human pose estimation via deep neural networks. In *CVPR*, 2014. 1

[53] K. van de Sande, J.R.R. Uijlings, T. Gevers, and A.W.M. Smeulders. Segmentation as Selective Search for Object Recognition. In *ICCV*, 2011. 8

[54] P. Viola, J. Platt, C. Zhang, et al. Multiple instance boosting for object detection. In *NIPS*, 2005. 2

[55] C. Wang, W. Ren, K. Huang, and T. Tan. Weakly supervised object localization with latent category learning. In *ECCV*. 2014. 2

[56] Y. Wei, W. Xia, J. Huang, B. Ni, J. Dong, Y. Zhao, and S. Yan. Cnn: Single-label to multi-label. *arXiv:1406.5726*, 2014. 3, 6

[57] J. Winn and N. Jojic. Locus: Learning object classes with unsupervised segmentation. In *ICCV*, 2005. 2

[58] P. Yadollahpour, D. Batra, and G. Shakhnarovich. Discriminative re-ranking of diverse segmentations. In *CVPR*, 2013. 2

[59] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *CVPR*, 2011. 1

[60] M. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. *arXiv:1311.2901*, 2013. 3, 6

[61] J. Zhang, M. Marszałek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: a comprehensive study. *IJCV*, 73(2):213–238, jun 2007. 1

[62] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Object detectors emerge in deep scene cnns. *CoRR*, abs/1412.6856, 2014. 3, 8