Microsoft COCO: Common Objects in Context

Tsung-Yi Lin[1], Michael Maire[2], Serge Belongie[1], James Hays[3], Pietro Perona[2], Deva Ramanan[4], Piotr Dollár[5], and C. Lawrence Zitnick[5]

[1] Cornell
[2] Caltech
[3] Brown
[4] UC Irvine
[5] Microsoft Research

Abstract. We present a new dataset with the goal of advancing the state-of-the-art in object recognition by placing the question of object recognition in the context of the broader question of scene understanding. This is achieved by gathering images of complex everyday scenes containing common objects in their natural context. Objects are labeled using per-instance segmentations to aid in precise object localization. Our dataset contains photos of 91 objects types that would be easily recognizable by a 4 year old. With a total of 2.5 million labeled instances in 328k images, the creation of our dataset drew upon extensive crowd worker involvement via novel user interfaces for category detection, instance spotting and instance segmentation. We present a detailed statistical analysis of the dataset in comparison to PASCAL, ImageNet, and SUN. Finally, we provide baseline performance analysis for bounding box and segmentation detection results using a Deformable Parts Model.

1 Introduction

One of the primary goals of computer vision is the understanding of visual scenes. Scene understanding involves numerous tasks including recognizing what objects are present, localizing the objects in 2D and 3D, determining the objects' and scene's attributes, characterizing relationships between objects and providing a semantic description of the scene. The current object classification and detection datasets [1,2,3,4] help us explore the first challenges related to scene understanding. For instance the ImageNet dataset [1], which contains an unprecedented number of images, has recently enabled breakthroughs in both object classification and detection research [5,6,7]. The community has also created datasets containing object attributes [8], scene attributes [9], keypoints [10], and 3D scene information [11]. This leads us to the obvious question: what datasets will best continue our advance towards our ultimate goal of scene understanding?

We introduce a new large-scale dataset that addresses three core research problems in scene understanding: detecting non-iconic views (or non-canonical perspectives [12]) of objects, contextual reasoning between objects and the precise 2D localization of objects. For many categories of objects, there exists an iconic view. For example, when performing a web-based image search for the

D. Fleet et al. (Eds.): ECCV 2014, Part V, LNCS 8693, pp. 740–755, 2014.

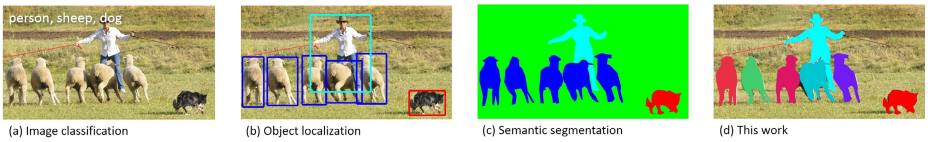


Fig. 1. While previous object recognition datasets have focused on (a) image classification, (b) object bounding box localization or (c) semantic pixel-level segmentation, we focus on (d) segmenting individual object instances. We introduce a large, richly-annotated dataset comprised of images depicting complex everyday scenes of common objects in their natural context

object category "bike," the top-ranked retrieved examples appear in profile, unobstructed near the center of a neatly composed photo. We posit that current recognition systems perform fairly well on iconic views, but struggle to recognize objects otherwise – in the background, partially occluded, amid clutter [13] – reflecting the composition of actual everyday scenes. We verify this experimentally; when evaluated on everyday scenes, models trained on our data perform better than those trained with prior datasets. A challenge is finding natural images that contain multiple objects. The identity of many objects can only be resolved using context, due to small size or ambiguous appearance in the image. To push research in contextual reasoning, images depicting scenes [3] rather than objects in isolation are necessary. Finally, we argue that detailed spatial understanding of object layout will be a core component of scene analysis. An object's spatial location can be defined coarsely using a bounding box [2] or with a precise pixel-level segmentation [14,15,16]. As we demonstrate, to measure either kind of localization performance it is essential for the dataset to have every instance of every object category labeled and fully segmented. Our dataset is unique in its annotation of instance-level segmentation masks, Fig. 1.

To create a large-scale dataset that accomplishes these three goals we employed a novel pipeline for gathering data with extensive use of Amazon Mechanical Turk. First and most importantly, we harvested a large set of images containing contextual relationships and non-iconic object views. We accomplished this using a surprisingly simple yet effective technique that queries for pairs of objects in conjunction with images retrieved via scene-based queries [17,3]. Next, each image was labeled as containing particular object categories using a hierarchical labeling approach [18]. For each category found, the individual instances were labeled, verified, and finally segmented. Given the inherent ambiguity of labeling, each of these stages has numerous tradeoffs that we explored in detail.

The Microsoft Common Objects in COntext (MS COCO) dataset contains 91 common object categories with 82 of them having more than 5,000 labeled instances, Fig. 6. In total the dataset has 2,500,000 labeled instances in 328,000 images. In contrast to the popular ImageNet dataset [1], COCO has fewer categories but more instances per category. This can aid in learning detailed object models capable of precise 2D localization. The dataset is also significantly larger in number of instances per category than the PASCAL VOC [2] and SUN [3] datasets. Additionally, a critical distinction between our dataset and others is

Fig. 2. Example of (a) iconic object images, (b) iconic scene images, and (c) non-iconic images. In this work we focus on challenging non-iconic images.

the number of labeled instances per image which may aid in learning contextual information, Fig. 5. MS COCO contains considerably more object instances per image (7.7) as compared to ImageNet (3.0) and PASCAL (2.3). In contrast, the SUN dataset, which contains significant contextual information, has over 17 objects and "stuff" per image but considerably fewer object instances overall.

An extended version of this work with additional details is available [19].

2 Related Work

Throughout the history of computer vision research datasets have played a critical role. They not only provide a means to train and evaluate algorithms, they drive research in new and more challenging directions. The creation of ground truth stereo and optical flow datasets [20,21] helped stimulate a flood of interest in these areas. The early evolution of object recognition datasets [22,23,24] facilitated the direct comparison of hundreds of image recognition algorithms while simultaneously pushing the field towards more complex problems. Recently, the ImageNet dataset [1] containing millions of images has enabled breakthroughs in both object classification and detection research using a new class of deep learning algorithms [5,6,7].

Datasets related to object recognition can be roughly split into three groups: those that primarily address object classification, object detection and semantic scene labeling. We address each in turn.

Image Classification. The task of object classification requires binary labels indicating whether objects are present in an image; see Fig. 1(a). Early datasets of this type comprised images containing a single object with blank backgrounds, such as the MNIST handwritten digits [25] or COIL household objects [26]. Caltech 101 [22] and Caltech 256 [23] marked the transition to more realistic object images retrieved from the internet while also increasing the number of object categories to 101 and 256, respectively. Popular datasets in the machine learning community due to the larger number of training examples, CIFAR-10 and CIFAR-100 [27] offered 10 and 100 categories from a dataset of tiny 32×32 images [28]. While these datasets contained up to 60,000 images and hundreds of categories, they still only captured a small fraction of our visual world.

Recently, ImageNet [1] made a striking departure from the incremental increase in dataset sizes. They proposed the creation of a dataset containing 22k categories with 500-1000 images each. Unlike previous datasets containing entry-level categories [29], such as "dog" or "chair," like [28], ImageNet used the WordNet Hierarchy [30] to obtain both entry-level and fine-grained [31] categories. Currently, the ImageNet dataset contains over 14 million labeled images and has enabled significant advances in image classification [5,6,7].

Object Detection. Detecting an object entails both stating that an object belonging to a specified class is present, and localizing it in the image. The location of an object is typically represented by a bounding box, Fig. 1(b). Early algorithms focused on face detection [32] using various ad hoc datasets. Later, more realistic and challenging face detection datasets were created [33]. Another popular challenge is the detection of pedestrians for which several datasets have been created [24,4]. The Caltech Pedestrian Dataset [4] contains 350,000 labeled instances with bounding boxes.

For the detection of basic object categories, a multi-year effort from 2005 to 2012 was devoted to the creation and maintenance of a series of benchmark datasets that were widely adopted. The PASCAL VOC [2] datasets contained 20 object categories spread over 11,000 images. Over 27,000 object instance bounding boxes were labeled, of which almost 7,000 had detailed segmentations. Recently, a detection challenge has been created from 200 object categories using a subset of 400,000 images from ImageNet [34]. An impressive 350,000 objects have been labeled using bounding boxes.

Since the detection of many objects such as sunglasses, cellphones or chairs is highly dependent on contextual information, it is important that detection datasets contain objects in their natural environments. In our dataset we strive to collect images rich in contextual information. The use of bounding boxes also limits the accuracy for which detection algorithms may be evaluated. We propose the use of fully segmented instances to enable more accurate detector evaluation.

Semantic Scene Labeling. The task of labeling semantic objects in a scene requires that each pixel of an image be labeled as belonging to a category, such as sky, chair, floor, street, etc. In contrast to the detection task, individual instances of objects do not need to be segmented, Fig. 1(c). This enables the labeling of objects for which individual instances are hard to define, such as grass, streets, or walls. Datasets exist for both indoor [11] and outdoor [35,14] scenes. Some datasets also include depth information [11]. Similar to semantic scene labeling, our goal is to measure the pixel-wise accuracy of object labels. However, we also aim to distinguish between individual instances of an object, which requires a solid understanding of each object's extent.

A novel dataset that combines many of the properties of both object detection and semantic scene labeling datasets is the SUN dataset [3] for scene understanding. SUN contains 908 scene categories from the WordNet dictionary [30] with segmented objects. The 3,819 object categories span those common to object detection datasets (person, chair, car) and to semantic scene labeling

(wall, sky, floor). Since the dataset was collected by finding images depicting various scene types, the number of instances per object category exhibits the long tail phenomenon. That is, a few categories have a large number of instances (wall: 20,213, window: 16,080, chair: 7,971) while most have a relatively modest number of instances (boat: 349, airplane: 179, floor lamp: 276). In our dataset, we ensure that each object category has a significant number of instances, Fig. 5.

Other Vision Datasets. Datasets have spurred the advancement of numerous fields in computer vision. Some notable datasets include the Middlebury datasets for stereo vision [20], multi-view stereo [36] and optical flow [21]. The Berkeley Segmentation Data Set (BSDS500) [37] has been used extensively to evaluate both segmentation and edge detection algorithms. Datasets have also been created to recognize both scene [9] and object attributes [8,38]. Indeed, numerous areas of vision have benefited from challenging datasets that helped catalyze progress.

3 Image Collection

We next describe how the object categories and candidate images are selected.

Common Object Categories. The selection of object categories is a non-trivial exercise. The categories must form a representative set of all categories, be relevant to practical applications and occur with high enough frequency to enable the collection of a large dataset. Other important decisions are whether to include both "thing" and "stuff" categories [39] and whether fine-grained [31,1] and object-part categories should be included. "Thing" categories include objects for which individual instances may be easily labeled (person, chair, car) where "stuff" categories include materials and objects with no clear boundaries (sky, street, grass). Since we are primarily interested in precise localization of object instances, we decided to only include "thing" categories and not "stuff." However, since "stuff" categories can provide significant contextual information, we believe the future labeling of "stuff" categories would be beneficial.

The specificity of object categories can vary significantly. For instance, a dog could be a member of the "mammal", "dog", or "German shepherd" categories. To enable the practical collection of a significant number of instances per category, we chose to limit our dataset to entry-level categories, i.e. category labels that are commonly used by humans when describing objects (dog, chair, person). It is also possible that some object categories may be parts of other object categories. For instance, a face may be part of a person. We anticipate the inclusion of object-part categories (face, hands, wheels) would be beneficial for many real-world applications.

We used several sources to collect entry-level object categories of "things." We first compiled a list of categories by combining categories from PASCAL VOC [2] and a subset of the 1200 most frequently used words that denote visually identifiable objects [40]. To further augment our set of candidate categories, several children ranging in ages from 4 to 8 were asked to name every object

they see in indoor and outdoor environments. The final 271 candidates may be found in [19]. Finally, the co-authors voted on a 1 to 5 scale for each category taking into account how commonly they occur, their usefulness for practical applications, and their diversity relative to other categories. The final selection of categories attempts to pick categories with high votes, while keeping the number of categories per super-category (animals, vehicles, furniture, etc.) balanced. Categories for which obtaining a large number of instances (greater than 5,000) was difficult were also removed. To ensure backwards compatibility all categories from PASCAL VOC [2] are also included. Our final list of 91 proposed categories is in Fig. 5(a).

Non-iconic Image Collection. Given the list of object categories, our next goal was to collect a set of candidate images. We may roughly group images into three types, Fig. 2: iconic-object images [41], iconic-scene images [3] and non-iconic images. Typical iconic-object images have a single large object in a canonical perspective centered in the image, Fig. 2(a). Iconic-scene images are shot from canonical viewpoints and commonly lack people, Fig. 2(b). Iconic images have the benefit that they may be easily found by directly searching for specific categories using Google or Bing image search. While iconic images generally provide high quality object instances, they can lack important contextual information and non-canonical viewpoints.

Our goal was to collect a dataset such that a majority of images are non-iconic, Fig. 2(c). It has been shown that datasets containing more non-iconic images are better at generalizing [42]. We collected non-iconic images using two strategies. First as popularized by PASCAL VOC [2], we collected images from Flickr which tends to have fewer iconic images. Flickr contains photos uploaded by amateur photographers with searchable metadata and keywords. Second, we did not search for object categories in isolation. A search for "dog" will tend to return iconic images of large, centered dogs. However, if we searched for pairwise combinations of object categories, such as "dog + car" we found many more non-iconic images. Surprisingly, these images typically do not just contain the two categories specified in the search, but numerous other categories as well. To further supplement our dataset we also searched for scene/object category pairs, see [19]. We downloaded at most 5 photos taken by a single photographer within a short time window. In the rare cases in which enough images could not be found, we searched for single categories and performed an explicit filtering stage to remove iconic images. The result is a collection of 328,000 images with rich contextual relationships between objects as shown in Figs. 2(c) and 6.

4 Image Annotation

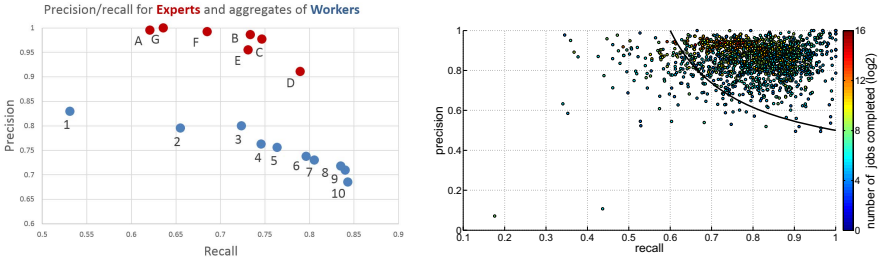
We next describe how we annotated our image collection. Due to our desire to label over 2.5 million category instances, the design of a cost efficient yet high quality annotation pipeline was critical. The annotation pipeline is outlined in Fig. 3. For all crowdsourcing tasks we used workers on Amazon's Mechanical Turk (AMT). Examples of our user interfaces can be found in [19].

Fig. 3. Our image annotation pipeline is split into 3 primary worker tasks: (a) Labeling the categories present in the image, (b) locating and marking all instances of the labeled categories, and (c) segmenting each object instance.

Category Labeling. The first task in annotating our dataset is determining which object categories are present in each image, Fig. 3(a). Since we have 91 potential categories and a large number of images, asking workers to answer 91 binary classification questions per image would be prohibitively expensive. Instead, we used a hierarchical approach [18]. Individual object categories are grouped into 11 super-categories (see [19]). For a given image, a worker was presented with each group of categories in turn and asked to indicate whether any instances exist for that super-category. This greatly reduces the time needed to classify the various categories. For instance, a worker may easily determine whether any animals are present in the image without having to specifically look for cats, dogs, etc. If a worker determines an instance in the super-category is present (animal), they indicate the instance's specific category (dog, cat, etc.) by dragging the category's icon onto the image over one instance of the category. The placement of these icons is critical for the following stage. To ensure high recall, five workers were asked to label each image; a detailed analysis of performance is presented shortly. This stage took 17,751 worker hours to complete.

Instance Spotting. In the next stage all instances of the object categories in an image were labeled, Fig. 3(b). In the previous stage each worker labeled one instance of a category, but multiple category instances may exist. For each image, a worker was asked to place crosses on top of each instance of a specific category found in the previous stage. To boost recall, the location of the instance found by the worker in the previous stage was shown to the current worker to help them in finding an initial instance. Without this priming, it can be difficult for a worker to quickly find an instance of a category upon first seeing the image. The workers could also use a magnifying glass to find small instances. Each worker was asked to label at most 10 instances of a specific category per image. Each image was completed by 5 workers for a total of 8,417 worker hours.

Instance Segmentation. Our final stage is the laborious task of segmenting each category instance, Fig. 3(c). For this stage we modified the excellent user interface developed by Bell et al. [16] for image segmentation. Our interface asks



(a) (b)

Fig. 4. (a) Precision and recall of experts (red) and the majority vote of AMT workers (blue). Note that the aggregate of 3 workers has better or similar recall to most experts. (b) illustrates the precision and recall of workers, with color indicating how many jobs they completed. For details and definition of ground truth for each plot see text.

the worker to segment a category instance specified by a worker in the previous stage. If other instances have already been segmented in the image, those segmentations are shown to the worker. If the worker does not see an instance of the category in the image (false positive from the previous stage) the worker may click "No <object name> in the image." Similarly if a worker does not find an unsegmented instance in the image they may specify "No unsegmented <object name> in the image."

Segmenting 2,500,000 object instances is an extremely time consuming task requiring over 22 worker hours per 1,000 segmentations. To minimize cost we only had a single worker segment each image. However, we initially found that most workers only produce a coarse outline of the instance resulting in poor segmentations. As a consequence, we required all workers to complete a training task for each object category. After reading the instructions, the training task asked workers to segment an object instance. If the worker's segmentation did not adequately match the ground truth segmentation the worker is repeatedly asked to improve their segmentation until it passes. The use of a training task vastly improves the quality of the workers (only about 1 in 3 workers passed the training stage) and resulting segmentations. Finally, the work of approved workers was periodically verified to ensure segmentation quality remains high. Example segmentations may be viewed in Fig. 6.

In some images many instances of the same category are tightly grouped together and it is hard to distinguish individual instances. For example, it might be difficult to segment an individual person from a crowd. In these cases, the group of instances is marked as one segment and labeled "do not care" for evaluation, e.g., finding people in a crowd will not affect a detector's score.

Annotation Performance Analysis. To ensure the quality of our annotations we analyze the quality of our workers by comparing them to expert workers. In Fig. 4 we show results for the task of category labeling. We compare the precision and recall of seven expert workers (co-authors of the paper) with the

results obtained by taking the union of one to ten AMT workers. For this task precision is of less importance since false positives will be removed at later stages, where adding false negatives is much more difficult. Fig. 4(a) shows that 5 AMT workers, the same number as was used to collect our labels, achieves the same recall as most of the expert workers. Note that the expert labelers achieved between 65% and 80% recall. These low values of recall are due to our liberal definition of a category being present. If only one expert labels an object category as being present, we assume the category is indeed present. However, the presence of many categories is often ambiguous. Upon closer inspection, we find recall values of 70% to 75% are generally sufficient to capture the non-ambiguous categories. Fig. 4(b) shows the precision and recall of our workers on category labeling. Unlike in Fig. 4(a), the ground truth labels were now estimated using a majority vote. The color indicates the number of jobs completed by each worker. Notice that workers who complete more hits have generally higher precision and recall. All jobs from workers below the black line were rejected.

5 Dataset Statistics

Next, we analyze the properties of the Microsoft Common Objects in COntext (MS COCO) dataset in comparison to several other popular datasets. These include the ImageNet [1], PASCAL VOC 2012 [2], and SUN [3] datasets. Each of these datasets varies significantly in size, list of labeled categories and types of images. ImageNet was created to capture a large number of object categories, many of which are fine-grained. SUN focuses on labeling scene types and the objects that commonly occur in them. Finally, PASCAL VOC's primary application is object detection in natural images. MS COCO is designed for the detection and segmentation of objects occurring in their natural context. The number of instances per category for all 91 categories collected so far are shown in Fig. 5(a). The completion of our final segmentation stage is still ongoing. Please see [19] for a complete list of collected segmentations, including over 580,000 people.

A summary of the datasets showing the number of object categories and the number of instances per category is shown in Fig. 5(d). While MS COCO has fewer categories than ImageNet and SUN, it has more instances per category which we hypothesize will be useful for learning complex models capable of precise localization. In comparison to PASCAL VOC, MS COCO has both more categories and instances.

An important property of our dataset is we strive to find non-iconic images containing objects in their natural context. The amount of contextual information present in an image can be estimated by examining the number of object categories and instances per image, Fig. 5(b, c). For ImageNet we plot the object detection validation set, since the training data only has a single object labeled. On average our dataset contains 3.5 categories and 7.7 instances per image. In comparison ImageNet and PASCAL VOC both have less than 2 categories and 3 instances per image on average. Another interesting comparison is only 10% of the images in MS COCO have only one category per image, in comparison to

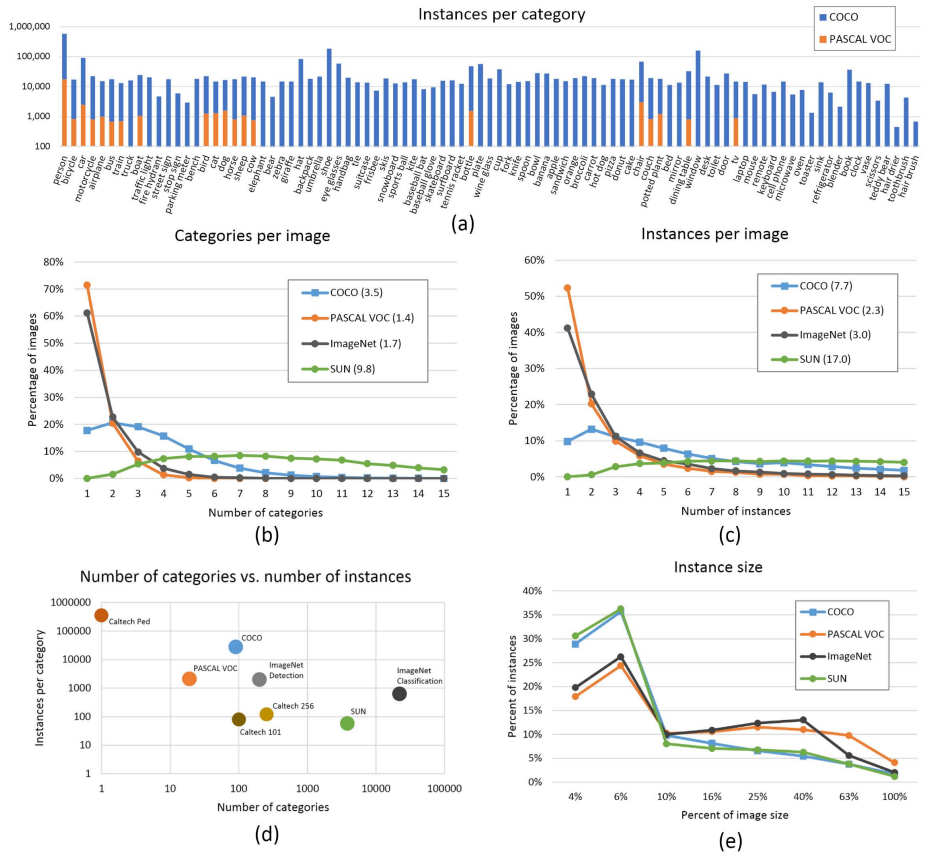


Fig. 5. (a) Number of annotated instances per category for MS COCO and PASCAL VOC. (b,c) Number of annotated categories and annotated instances, respectively, per image for MS COCO, ImageNet Detection, PASCAL VOC and SUN (average number of categories and instances are shown in parentheses). (d) Number of categories vs. the number of instances per category for a number of popular object recognition datasets. (e) The distribution of instance sizes for the MS COCO, ImageNet Detection, PASCAL VOC and SUN datasets.

over 60% of images containing a single object category in ImageNet and PASCAL VOC. As expected, the SUN dataset has the most contextual information since it is scene-based.

Finally, we analyze the average size of objects in the datasets. Generally smaller objects are harder to recognize and require more contextual reasoning to recognize. As shown in Fig. 5(e), the average sizes of objects is smaller for both MS COCO and SUN.

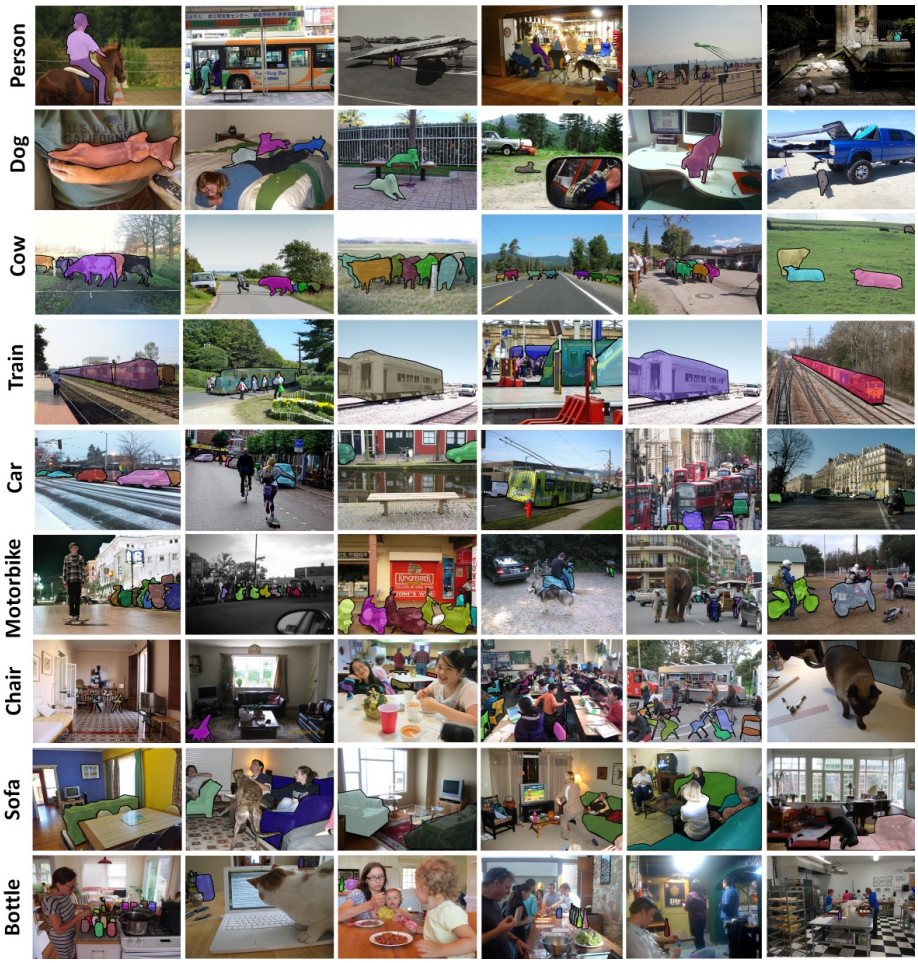


Fig. 6. Samples of annotated images in the MS COCO dataset

6 Algorithmic Analysis

To establish a concrete benchmark, we split our dataset into training, validation, and test data. We have a training set of 164,000 images and a validation and test set of 82,000 images each. We took care to minimize the chance of near-duplicate images existing across splits by explicitly removing duplicates (detected with [43]) and splitting images by date and user. Following now-established protocol, we will release annotations for train and validation images, but not test.

Bounding-box Detection. We begin by examining the performance of the well-studied 20 PASCAL object categories on our dataset. We take a subset of 55,000 images from train/val data for the following experiment and obtain tight-fitting bounding boxes from the annotated segmentation masks. We evaluate

Table 1. Top: Detection performance evaluated on **PASCAL VOC 2012**. DPMv5-P is the performance reported by Girshick et al. in VOC release 5. DPMv5-C uses the same implementation, but is trained with MS COCO. **Bottom**: Performance evaluated on **MS COCO** for DPM models trained with PASCAL VOC 2012 (DPMv5-P) and MS COCO (DPMv5-C). For DPMv5-C we used 5000 positive and 10000 negative training examples. While MS COCO is considerably more challenging than PASCAL, use of more training data coupled with more sophisticated approaches [5,6,7] should improve performance substantially.

	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	moto	person	plant	sheep	sofa	train	tv	Avg.
DPMv5-P	**45.6**	49.0	11.0	**11.6**	**27.2**	50.5	**43.1**	**23.6**	**17.2**	23.2	**10.7**	**20.5**	42.5	**44.5**	**41.3**	**8.7**	**29.0**	**18.7**	**40.0**	34.5	**29.6**
DPMv5-C	43.7	**50.1**	**11.8**	2.4	21.4	**60.1**	35.6	16.0	11.4	**24.8**	5.3	9.4	**44.5**	41.0	35.8	6.3	28.3	13.3	38.8	**36.2**	26.8
DPMv5-P	35.1	17.9	3.7	2.3	**7**	45.4	**18.3**	8.6	**6.3**	17	4.8	**5.8**	35.3	25.4	**17.5**	4.1	**14.5**	9.6	31.7	27.9	16.9
DPMv5-C	**36.9**	**20.2**	**5.7**	**3.5**	6.6	**50.3**	16.1	**12.8**	4.5	**19.0**	**9.6**	4.0	**38.2**	**29.9**	15.9	**6.7**	13.8	**10.4**	**39.2**	**37.9**	**19.1**

models tested on both the MS COCO and PASCAL datasets, see Table 1. We evaluate two different models. **DPMv5-P**: the latest implementation of [44] (release 5 [45]) trained on PASCAL VOC 2012. **DPMv5-C**: the same implementation trained on COCO (5000 positive and 10000 negative images). We use the default parameter settings for training COCO models.

If we compare the average performance of DPMv5-P on PASCAL VOC and MS COCO, we find that average performance on MS COCO drops by nearly a *factor of 2*, suggesting that MS COCO does include more difficult (non-iconic) images of objects that are partially occluded, amid clutter, etc. We notice a similar drop in performance for the model trained on MS COCO (DPMv5-C).

The effect on detection performance of training on PASCAL VOC or MS COCO may be analyzed by comparing DPMv5-P and DPMv5-C. They use the same implementation with different sources of training data. Table 1 shows DPMv5-C still outperforms DPMv5-P in 6 out of 20 categories when testing on PASCAL VOC. In some categories (e.g., dog, cat, people), models trained on MS COCO perform worse, while on others (e.g., bus, tv, horse), models trained on our data are better.

Consistent with past observations [46], we find that including difficult (non-iconic) images during training may not always help. Such examples may act as noise and pollute the learned model if the model is not rich enough to capture such appearance variability. Our dataset allows for the exploration of such issues.

Torralba and Efros [42] proposed a metric to measure cross-dataset generalization which computes the 'performance drop' for models that train on one dataset and test on another. The performance difference of the DPMv5-P models across the two datasets is 12.7 AP while the DPMv5-C models only have 7.7 AP difference. Moreover, overall performance is much lower on MS COCO. These observations support two hypotheses: 1) MS COCO is significantly more difficult than PASCAL VOC and 2) models trained on MS COCO can generalize better to easier datasets such as PASCAL VOC given more training data. To gain insight into the differences between the datasets, see [19] for visualizations of person and chair examples from the two datasets.

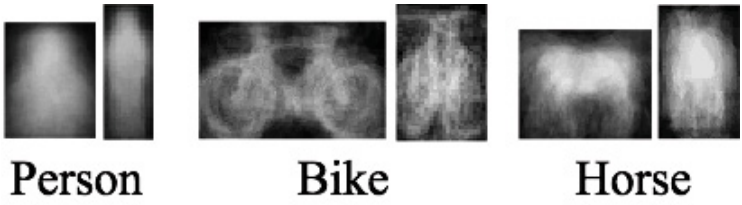


Fig. 7. We visualize our mixture-specific shape masks. We paste thresholded shape masks on each candidate detection to generate candidate segments.

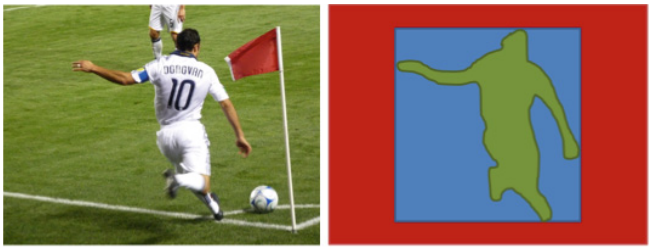


Fig. 8. Evaluating instance detections with segmentation masks versus bounding boxes. Bounding boxes are a particularly crude approximation for articulated objects; in this case, the majority of the pixels in the (**blue**) tight-fitting bounding-box do not lie on the object. Our (**green**) instance-level segmentation masks allows for a more accurate measure of object detection and localization.

Generating Segmentations from Detections. We now describe a simple method for generating object bounding boxes and segmentation masks, following prior work that produces segmentations from object detections [47,48,49,50]. We learn aspect-specific pixel-level segmentation masks for different categories. These are readily learned by averaging together segmentation masks from aligned training instances. We learn different masks corresponding to the different mixtures in our DPM detector. Sample masks are visualized in Fig. 7.

Detection Evaluated by Segmentation. Segmentation is a challenging task even assuming an object detector reports correct results as it requires fine localization of object part boundaries. To decouple segmentation evaluation from detection correctness, we benchmark segmentation quality using only correct detections. Specifically, given that the object detector reports a correct bounding box, how well does the predicted segmentation of that object match the groundtruth segmentation? As criterion for correct detection, we impose the standard requirement that intersection over union between predicted and groundtruth boxes is at least 0.5. We then measure the intersection over union of the predicted and groundtruth segmentation masks, see Fig. 8. To establish a baseline for our dataset, we project learned DPM part masks onto the image to create segmentation masks. Fig. 9 shows results of this segmentation baseline for the DPM learned on the 20 PASCAL categories and tested on our dataset.

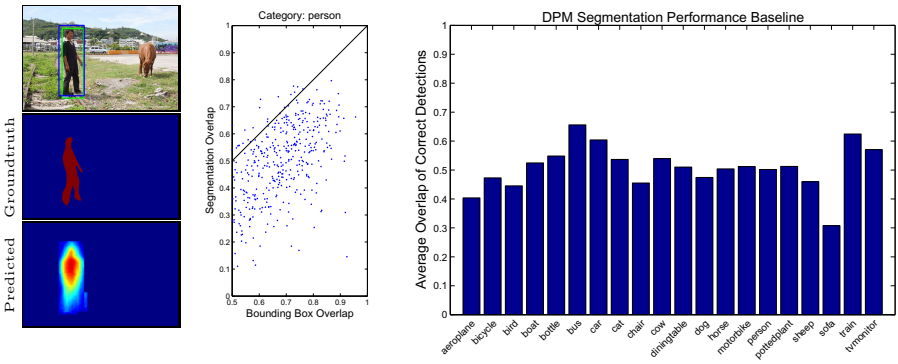


Fig. 9. A predicted segmentation might not recover object detail even though detection and groundtruth bounding boxes overlap well (left). Sampling from the person category illustrates that on a per-instance basis, predicting segmentation from top-down projection of DPM part masks is difficult even for correct detections (center). Averaging over instances for each of the PASCAL VOC categories on our dataset demonstrates that it presents a challenge for object segmentation algorithms (right).

7 Discussion

We described a new dataset for detecting and segmenting objects found in everyday life in their natural environments. Utilizing around 60,000 worker hours, a vast collection of category instances was gathered, annotated and organized to drive the advancement of object detection and segmentation algorithms. Emphasis was placed on finding non-iconic images of objects in natural environments and varied viewpoints. Dataset statistics indicate the images contain rich contextual information with many objects present per image.

There are several promising directions for future annotations on our dataset. We currently only label "things", but labeling "stuff" may also provide significant contextual information that may be useful for detection. Many object detection algorithms benefit from additional annotations, such as the amount an instance is occluded [4] or the location of keypoints on the object [10]. Finally, our dataset could provide a good benchmark for other types of labels, including scene types [3], attributes [9,8] and full sentence written descriptions [51].

To download and learn more about MS COCO please see the project website[1]. Additional details are presented in an extended version of this work [19]. MS COCO will evolve and grow over time; up to date information is available online.

Acknowledgments. Funding for all crowd worker tasks was provided by Microsoft. P.P. and D.R. were supported by ONR MURI Grant N00014-10-1-0933. We would like to thank all members of the community who provided valuable feedback throughout the process of defining and collecting the dataset.

[1] `http://mscoco.org/`

References

1. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A Large-Scale Hierarchical Image Database. In: CVPR (2009)
2. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL visual object classes (VOC) challenge. IJCV 88(2), 303–338 (2010)
3. Xiao, J., Hays, J., Ehinger, K.A., Oliva, A., Torralba, A.: SUN database: Large-scale scene recognition from abbey to zoo. In: CVPR (2010)
4. Dollár, P., Wojek, C., Schiele, B., Perona, P.: Pedestrian detection: An evaluation of the state of the art. PAMI 34 (2012)
5. Krizhevsky, A., Sutskever, I., Hinton, G.: ImageNet classification with deep convolutional neural networks. In: NIPS (2012)
6. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: CVPR (2014)
7. Sermanet, P., Eigen, D., Zhang, S., Mathieu, M., Fergus, R., LeCun, Y.: OverFeat: Integrated recognition, localization and detection using convolutional networks. In: ICLR (April 2014)
8. Farhadi, A., Endres, I., Hoiem, D., Forsyth, D.: Describing objects by their attributes. In: CVPR (2009)
9. Patterson, G., Hays, J.: SUN attribute database: Discovering, annotating, and recognizing scene attributes. In: CVPR (2012)
10. Bourdev, L., Malik, J.: Poselets: Body part detectors trained using 3D human pose annotations. In: ICCV (2009)
11. Silberman, N., Hoiem, D., Kohli, P., Fergus, R.: Indoor segmentation and support inference from RGBD images. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part V. LNCS, vol. 7576, pp. 746–760. Springer, Heidelberg (2012)
12. Palmer, S., Rosch, E., Chase, P.: Canonical perspective and the perception of objects. Attention and Performance IX 1, 4 (1981)
13. Hoiem, D., Chodpathumwan, Y., Dai, Q.: Diagnosing error in object detectors. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part III. LNCS, vol. 7574, pp. 340–353. Springer, Heidelberg (2012)
14. Brostow, G., Fauqueur, J., Cipolla, R.: Semantic object classes in video: A high-definition ground truth database. PRL 30(2), 88–97 (2009)
15. Russell, B., Torralba, A., Murphy, K., Freeman, W.: LabelMe: a database and web-based tool for image annotation. IJCV 77(1-3), 157–173 (2008)
16. Bell, S., Upchurch, P., Snavely, N., Bala, K.: OpenSurfaces: A richly annotated catalog of surface appearance. SIGGRAPH 32(4) (2013)
17. Ordonez, V., Kulkarni, G., Berg, T.: Im2text: Describing images using 1 million captioned photographs. In: NIPS (2011)
18. Deng, J., Russakovsky, O., Krause, J., Bernstein, M., Berg, A., Fei-Fei, L.: Scalable multi-label annotation. In: CHI (2014)
19. Lin, T., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: Common objects in context. CoRR abs/1405.0312 (2014)
20. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. IJCV 47(1-3), 7–42 (2002)
21. Baker, S., Scharstein, D., Lewis, J., Roth, S., Black, M., Szeliski, R.: A database and evaluation methodology for optical flow. IJCV 92(1), 1–31 (2011)
22. Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In: CVPR Workshop of Generative Model Based Vision, WGMBV (2004)

23. Griffin, G., Holub, A., Perona, P.: Caltech-256 object category dataset. Technical Report 7694, California Institute of Technology (2007)
24. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR (2005)
25. Lecun, Y., Cortes, C.: The MNIST database of handwritten digits (1998)
26. Nene, S.A., Nayar, S.K., Murase, H.: Columbia object image library (coil-20). Technical report, Columbia Universty (1996)
27. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images. Computer Science Department, University of Toronto, Tech. Rep. (2009)
28. Torralba, A., Fergus, R., Freeman, W.T.: 80 million tiny images: A large data set for nonparametric object and scene recognition. PAMI 30(11), 1958–1970 (2008)
29. Ordonez, V., Deng, J., Choi, Y., Berg, A., Berg, T.: From large scale image categorization to entry-level categories. In: ICCV (2013)
30. Fellbaum, C.: WordNet: An electronic lexical database. Blackwell Books (1998)
31. Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S., Perona, P.: Caltech-UCSD Birds 200. Technical Report CNS-TR-201, Caltech. (2010)
32. Hjelmås, E., Low, B.: Face detection: A survey. CVIU 83(3), 236–274 (2001)
33. Huang, G.B., Ramesh, M., Berg, T., Learned-Miller, E.: Labeled faces in the wild. Technical Report 07-49, University of Massachusetts, Amherst (October 2007)
34. Russakovsky, O., Deng, J., Huang, Z., Berg, A., Fei-Fei, L.: Detecting avocados to zucchinis: what have we done, and where are we going? In: ICCV (2013)
35. Shotton, J., Winn, J., Rother, C., Criminisi, A.: TextonBoost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context. IJCV 81(1), 2–23 (2009)
36. Seitz, S.M., Curless, B., Diebel, J., Scharstein, D., Szeliski, R.: A comparison and evaluation of multi-view stereo reconstruction algorithms. In: CVPR (2006)
37. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. PAMI 33(5), 898–916 (2011)
38. Lampert, C., Nickisch, H., Harmeling, S.: Learning to detect unseen object classes by between-class attribute transfer. In: CVPR (2009)
39. Heitz, G., Koller, D.: Learning spatial context: Using stuff to find things. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 30–43. Springer, Heidelberg (2008)
40. Sitton, R.: Spelling Sourcebook. Egger Publishing (1996)
41. Berg, T., Berg, A.: Finding iconic images. In: CVPR (2009)
42. Torralba, A., Efros, A.: Unbiased look at dataset bias. In: CVPR (2011)
43. Douze, M., Jégou, H., Sandhawalia, H., Amsaleg, L., Schmid, C.: Evaluation of gist descriptors for web-scale image search. In: CIVR (2009)
44. Felzenszwalb, P., Girshick, R., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. PAMI 32(9), 1627–1645 (2010)
45. Girshick, R., Felzenszwalb, P., McAllester, D.: Discriminatively trained deformable part models, release 5. PAMI (2012)
46. Zhu, X., Vondrick, C., Ramanan, D., Fowlkes, C.: Do we need more training data or better models for object detection? In: BMVC (2012)
47. Brox, T., Bourdev, L., Maji, S., Malik, J.: Object segmentation by alignment of poselet activations to image contours. In: CVPR (2011)
48. Yang, Y., Hallman, S., Ramanan, D., Fowlkes, C.: Layered object models for image segmentation. PAMI 34(9), 1731–1743 (2012)
49. Ramanan, D.: Using segmentation to verify object hypotheses. In: CVPR (2007)
50. Dai, Q., Hoiem, D.: Learning to localize detected objects. In: CVPR (2012)
51. Rashtchian, C., Young, P., Hodosh, M., Hockenmaier, J.: Collecting image annotations using Amazon's Mechanical Turk. In: NAACL Workshop (2010)