Improving Weakly-Supervised Object Localization By Micro-Annotation

Alexander Kolesnikov
akolesnikov@ist.ac.at
Christoph H. Lampert
chl@ist.ac.at

IST Austria
Am Campus 1
3400 Klosterneuburg
Austria

Abstract

Weakly-supervised object localization methods tend to fail for object classes that consistently co-occur with the same background elements, *e.g. trains* on *tracks*. We propose a method to overcome these failures by adding a very small amount of model-specific additional annotation. The main idea is to cluster a deep network's mid-level representations and assign *object* or *distractor* labels to each cluster. Experiments show substantially improved localization results on the challenging ILSVRC 2014 dataset for bounding box detection and the PASCAL VOC 2012 dataset for semantic segmentation.

1 Introduction

A crucial step for automatic systems for image understanding is the ability to localize objects in a scene in the form of bounding boxes or segmentation masks. Over the last decade, computer vision research has made great progress on this task by employing machine learning techniques, in particular deep learning. However, learning-based methods require large amounts of annotated training data, and manually annotating images with object locations is a tedious and expensive process. It is therefore an important, and largely unsolved, problem to develop object localization methods that can be trained from weak supervision, by which in the context of this work we mean per-image class labels or tags. Analyzing the existing methods in this field, it becomes apparent that certain object classes, for example *trains* or *boats*, are consistently harder to localize from weak supervision than others, in the sense that there is a big gap between the prediction quality of the models trained from just per-image class labels and the quality achieved by models trained with full supervision.

Figure 1: Failure cases of weakly-supervised object localization: based only on per-image class labels one cannot distinguish between objects (*train, snowmobile*) and consistently co-occurring distractors (*tracks, snow*).

One major reason for these failure cases is illustrated in Figure 1. Weakly-supervised object localization tends to fail when object classes systematically co-occur with *distractors*

(background or certain other classes), for example *trains* with *tracks*. Per-image class annotation simply does not contain necessary information to reliably learn the difference between objects and distrators. In this work, we argue that the best way to overcome this problem is to collect a tiny amount of additional annotation, which we call *micro-annotation*.

The approach relies on the hypothesis that even though it might be impossible for a classifier to learn from weak per-image annotation which parts of an image are the object of interest and which are distractors, it will still be possible to distinguish both groups from each other by clustering their appearance in a suitable representation. Then, all we need in order to improve the quality of a weakly-supervised object localization systems is a way to find out which clusters belong to distractors – which is an easy task for a human annotator – and suppress them.

This micro-annotation approach can be used in combination with many existing localization methods. In this work we combine it with the current state-of-the-art methods for weakly-supervised bounding box prediction and for weakly-supervised semantic segmentation, showing improved results on the challenging ILSVR2014 and PASCAL VOC2012 datasets.

Apart from its practical usefulness, an interesting aspect of this approach is that it asks for user annotation after an initial model has already been trained. This allows the requested information to depend on the original model's output, hopefully with the effect that the new information has a maximally beneficial effect on the prediction quality. This setup resembles *active learning*, but with an even better relation between the amount of annotation and the model improvement. In active learning, the annotation provides information through the labeling of individual images, and each provided label typically influences the model parameters by an amount inversely proportional to the total size of the training set. Consequently, active learning is most beneficial for models trained on small datasets. In our approach, a single user interaction can have a large effect on the model parameters and thereby the prediction quality, namely when it establishes that all detected patterns of a certain type are distractors and should be suppressed. The size of the training set plays no role for this effect, and indeed we observe a significant improvement even for models trained on very large datasets.

2 Related work

Many methods for object localization have been proposed that can be trained in a weakly-supervised way from per-image class label annotation. The majority of these methods predict either object bounding boxes [1, 2, 3, 5, 7, 28, 29, 38, 43] or per-pixel segmentation masks [14, 18, 19, 20, 21, 30, 31, 32, 33, 34, 39, 40, 41]. The currenlty most successful approaches obtain localization hints from convolutional neural networks that are trained for the of image classification, *e.g.* [14, 43]. The micro-annotation method can in principle be used on top of any of such method, as long as that has the ability to produce per-location score maps.

Much fewer works have studied how the process of data annotation can be improved by the power of strong computer vision systems. Two related research directions are *learning with humans in the loop* [4] and *active learning* [25]. In the human in the loop concept, an automatic system and a human user work together during the prediction stage. For example, in a fine-grained classification task [4, 6, 36, 37], the machine would output a selection of possible labels and the human user would pick the most appropriate one. Typically, this process does not improve the model parameters, though, so feedback from a human is always

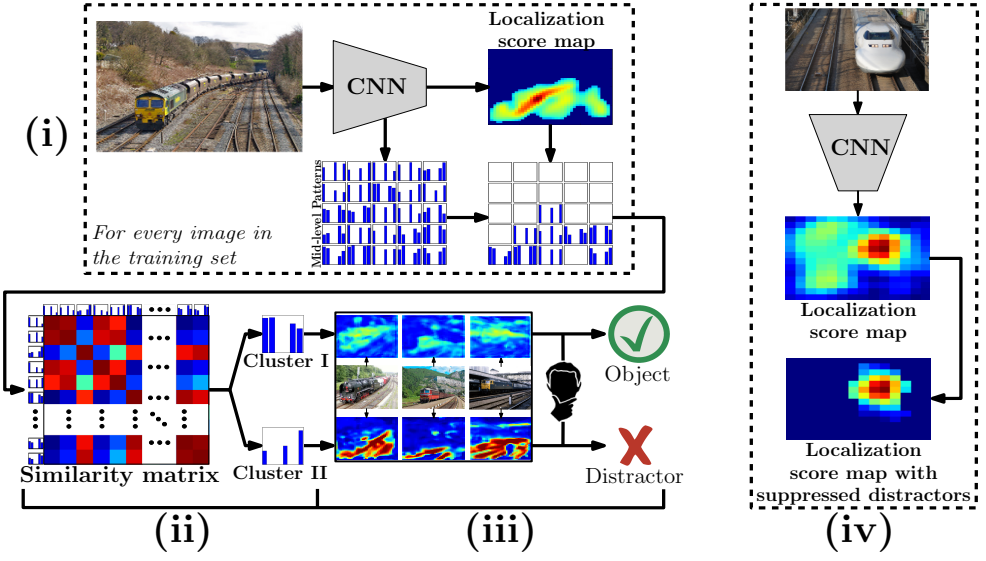

Figure 2: Schematic illustration of the proposed micro-annotation approach: (i) for every image region across all training images that is predicted to show the object of interest (here: *train*), compute its mid-level feature representation (pattern) and form a pool of patterns, (ii) find characteristic clusters in this pool by spectral clustering, (iii) visualize the clusters by heatmaps and ask a user to annotate them as representing either the object or a distractor and (iv) at test time modify the produced localization maps by discarding regions that correspond to the distractors.

required to make high quality predictions. Active learning also has the goal of harvesting human expertise, but it does so during the training stage: an automatic system has access to a large number of unlabeled images and can ask a human annotator to specifically annotate a subset of them [9, 12, 13, 22]. This procedure can reduce the amount of necessary annotation, *e.g.* when the system only ask for annotation of images that it is not already certain about anyway. In the context of object localization, a more efficient weakly-supervised variant of active learning has been proposed in which a human user only has to annotate if a predicted bounding box is correct or not, instead of having to draw it manually [17]. Our approach differs from these setting in particular in the fact that we do not ask a user to provide feedback about individual images, but about clusters in the learned data representation, which reflects information extracted from the whole training set. Thereby, we require much less interaction with the annotator, namely in the order of the number of classes instead of in the order of the number of training images.

On the technical level, our method is related to recent approaches for discovering object detectors in deep convolutional neural networks [16, 26, 42]. However, these rely on the assumption that part-detectors correspond to individual convolution filter outputs, whereas our clustering approach finds co-occurring patterns in the distributed representation learned by the network. The fact that our method identifies image structures by clustering visual representations across many images resembles image co-segmentation [23]. It differs, however, from these earlier works in how it organizes the clustering process and how it uses the structure that are found.

Algorithm 1: Spectral clustering algorithm

input : patterns A^y, eigenvalue threshold ρ (default: 0.7), lower bound m (default: 2) and upper bound M (default 4) for the number of clusters

Compute the similarity matrix: $W^y \in \mathbb{R}^{|A^y|\times|A^y|}$ with $W^y_{a,b} = \langle a,b\rangle \geq 0$ for all $a,b \in A^y$.

Compute the diagonal matrix: $D^y \in \mathbb{R}^{|A^y|\times|A^y|}$ with $D^y_{a,a} = \sum_{b\in A^y} W^y_{a,b}$ for all $a \in A^y$.

Compute the Laplacian matrix: $L^y = D^y - W^y$.

Compute the M smallest eigenvalues $\{\lambda_1,\dots\lambda_M\}$ and eigenvectors $\{v_1,\dots v_M\}$ of $(D^y)^{-1}L^y$ by solving the generalized eigenvalue problem $L^y v = \lambda D^y v$.

Set the number of clusters, k, as the number of eigenvalues below ρ or the lower bound.

Construct matrix $U = [v_1|\cdots|v_k] \in \mathbb{R}^{|A^y|\times k}$.

Let u_a be the row of U that corresponds to a pattern a.

Use *k-means* to cluster the matrix rows $\{u_a\}_{a\in A^y}$ into clusters, $\{U_1,\dots,U_k\}$.

output: clustering $P^y = \{C_1,C_2,\dots,C_k\}$, where $C_i = \{a|u_a \in U_i\}$ for $i = 1,\dots,k$

3 Improving Object Localization By Micro-Annotation

In this section we formally introduce the proposed procedure for collecting micro-annotation (illustrated in Figure 2) and improving object localization (illustrated in Figure 4). The main steps for obtaining the additional annotation for each class are: (i) represent all predicted foreground regions of all images by feature vectors, (ii) cluster the feature vectors (iii) visualize the clusters and let an annotator select which ones actually corresponds to the object class of interest. The information about clusters and their annotation can then be used to better localize objects: (iv) for any (new) image, predict a foreground map using only the image regions that match clusters labeled as 'object'.

In the rest of the section we explain these steps in detail. For this, we denote the set of training images by $\mathcal{D}$ and assume a fixed set of semantic categories $\mathcal{Y}$ that we want to localize. The subsequent construction can be performed independently for each object class. By $y \in \mathcal{Y}$ we always denote the current class of interest.

We assume that we are given a pretrained deep convolutional neural network, f, that predicts the presence of semantic categories for an input image X, but that can also be leveraged to predict the spatial location of semantic objects in input images. Formally, we assume that each image is regularly split into a set of non-overlaping rectangular regions, $\mathcal{U}$, and that f gives rise to a *scoring function* that assigns a localization score, $S^y_u(X)$, to each image region $u \in \mathcal{U}$. Furthermore, we assume the availability of a *thresholding procedure* that converts *score maps*, $S^y(X) \in \mathbb{R}^{|\mathcal{U}|}$, to a set of image regions, $D^y(X) \subset \mathcal{U}$, that represent the predicted localization of the class y. We discuss particular choices for the functions S and D, in Section 4.

(i) Region representations. At any fixed layer of the deep network we can form a feature vector, $\phi_u(X) \in \mathbb{R}^k$, (called a *pattern*) for any region, $u \in \mathcal{U}$, by concatenating the real-valued convolutional filter activations from the fixed layer. For simpler use in the clustering step we assume that $\phi_u(X)$ has nonnegative entries, *e.g.* after a ReLU operation, and that it is L^2-normalized. These are not principled restrictions, though. Arbitrary features could be used in combination with a different clustering algorithm. We form a *pattern set* $A^y = \{\phi_u(X)|\forall u \in D^y(X), \forall X \in \mathcal{D}\}$, *i.e.* the features of all image regions with predicted label y.

(ii) Clustering. We partition the set A^y into a group of clusters, $P^y = \{C_1,\dots,C_k\}$, by *spectral*

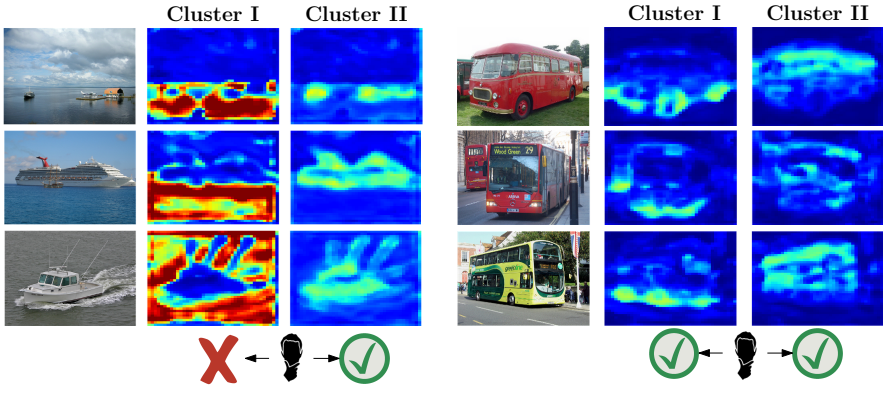

Figure 3: The schematic illustration of the annotation process. For any semantic category, we visualize corresponding mid-level feature clusters by heatmaps. An annotator marks every class as either representing an object of interest or background.

clustering [35]. The number of clusters is determined automatically using the fact that the k-th eigenvalues of the graph Laplacian (computed during the clustering) reflects the quality of creating k clusters (Algorithm 1, line 5).

General spectral clustering does not scale well to large datasets, because when used with a generic similarity measure it requires memory quadratic and runtime cubic in the number of patterns. This is not the case for us: we use a linear (inner product) similarity measure between patterns, which allows us to avoid storing the quadratically sized similarity matrix *(line 1)* explicitly. The necessary eigenvalue problem *(line 4)* we solve efficiently by the Lanczos method [15], which requires only low-rank matrix-vector multiplications. The resulting algorithm scales linearly in the number of patterns to be clustered and can thereby be applied even to datasets with millions of patterns.

(iii) Cluster visualization and annotation. Our main assumption is that any cluster, $C \in P^y$, will correspond either to (part of) the object of interest, or to a distractor. To identify which of these possibilities it is, we introduce an efficient annotation step.

For each class, we randomly sample a small number, *e.g.* 12, of images from the training set. For each sampled image X we produce heatmaps, $H^y(X|C) \in \mathbb{R}^{|\mathcal{U}|}$, for each cluster, C, that depict for each region u the average similarity of the region pattern $\phi^y_u(X)$ to the patterns in the corresponding cluster, *i.e.*

$$H^y_u(X|C) = \frac{1}{|C|} \sum_{a \in C} \langle \phi^y_u(X), a \rangle. \tag{1}$$

Note that in practice we can compute this value without always summing over all patterns: we pre-compute the average cluster pattern, $a_C = \frac{1}{|C|} \sum_{a \in C} a$, and use $H^y_u(X|C) = \langle \phi^y_u(X), a_C \rangle$. We display the heatmaps and ask a human annotator to mark which of the clusters correspond to the object class of interest in the images, see Figure 3 for an illustration of the process. Overall, the annotation requires just one user interaction (a few mouse clicks) per class. Our experience shows that each interaction takes in the order of a few seconds, so a few hundred classes can be annotated within an hour.

(iv) Improved object class localization. The per-class annotations allow us to obtain better location predictions without having to retrain the network. Let X be a new image with

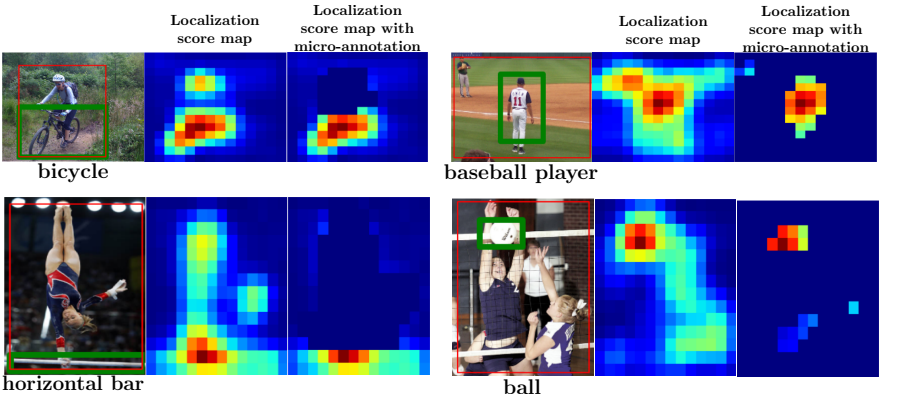

Figure 4: The effect of discarding localizations that correspond to a mid-level representation (pattern) that is assigned to a cluster annotated as distractor.

predicted localization score map $S^y_u(X)$. For each location $u \in \mathcal{U}$ we determine the cluster that results in the highest heatmap score, $C^* = \min_{C \in P^y} H^y_u(X|C)$. If C^* was not annotated as an 'object' cluster, we set $S^y_u(X)$ to $-\infty$, in order to prevent that the class y will be predicted at the location u.

4 Experiments

In this section we evaluate the effectiveness of our approach. We apply it to state-of-the-art methods for predicting object bounding boxes and semantic segmentations from image-level supervision and report experimental result on two challenging computer vision benchmarks: ILSVRC 2014 and PASCAL VOC 2012.

Predicting object bounding boxes. We follow the protocol of the ILSVRC 2014 *classification-with-localization* challenge: the goal is to predict which of 1000 object classes is present in an image and localize it by predicting a bounding box [24]. By the challenge protocol, up to five classes and their bounding boxes can be predicted, and the output is judged as correct if a bounding box of the correct class is predicted with an intersection-over-union score of at least 50% with a ground-truth box.

In this work we are particularly interested in the weakly-supervised setting, when models are trained using only per-image category information. We build on the state-of-the art technique GAP [43], which uses a deep convolutional network with modified VGG [27] architecture. Internally, GAP produces localization score maps, $S^y(X)$, by means of the *CAM (class activation maps)* procedure, see [43] for details. GAP also includes a thresholding function: given a score map for a class y, all locations that have a with a score larger than 20% of the maximum score are selected, *i.e.* $D^y(X) = \{u | S^y_u(X) > 0.2 \max_{u \in \mathcal{U}} S^y_u(X)\}$. At test time, for any input image five bounding boxes are produced, one for each class of the set of top-5 classes predicted by the convolutional neural network. For each predicted class the bounding box is produced, so that it covers the largest connected component of $D^y(X)$[1]

[1]Better results can be achieved by using a more involved method based on multiple image crops. This was used for the results in [43], but is not described in the manuscript, so we use the simpler but reproducible setting.

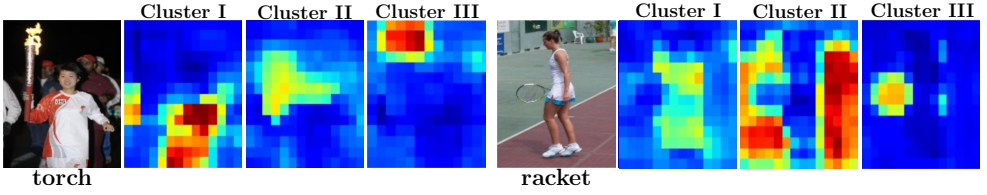

Figure 5: Examples of mid-level pattern clusters: *torch* has clusters that correspond to *person*, *torch* and *fire*, and *racket* has clusters that correspond to *player*, *court* and *racket*.

Trained on the 1.2 million ILSVRC *train* images, this approach achieves an error rate of 49.9% on the ILSVRC *val* set.

Improving bounding box predictions. We use the proposed micro-annotation technique to improve the *CAM* localization score maps. For computing mid-level feature representations (patterns) we use the *conv5_3* layer of the modified VGG network from [43]. This choice is motivated by the closely related papers [26, 42] that studied object/part detectors emerging in convolutional neural networks. We set $\rho = 0.7$ as clustering threshold parameter and predict between 2 and 4 clusters. For the majority of classes (all except 56) we obtain only two clusters.

Annotating all clusters for the 1000 classes requires less then 6 hours of annotator time. In 182 semantic classes at least one cluster was identified as *distractor*, while for the remaining classes different clusters typically correspond to different object parts. See Figure 5 for a visualization of obtained object parts.

After modifying the localization scores according to our method we compute bounding boxes for all *val* images. By construction, only the localization scores for 182 classes are affected compared to the baseline GAP results. On average, the localization performance improved by 4.9% for those classes, with individual improvements up to 38% (*flatworm*). In Figure 6(a) we illustrate the results for the 79 classes for which localization score changed by at least 5% compared to the baseline. We observed that for vast majority of classes our method helps to improve localization performance. As expected, many of these are examples where object and background consistently co-occur, for example *speedboat* (improved from 42% to 80%) or *snowmobile* (improved from 32% to 60%). For a few classes, we observed a decrease of performance. We inspected these visually and observed a few possible reasons: one possibility is that distractors are present in the image, but they actually help to find a better bounding box: for example, drawing a bounding box around a complete person often achieves above 50% intersection-over-union for predicting *bath towels*. A second possibility is that objects consist of multiple visually disconnected parts, *e.g. sandals*. GAP's large-connected-component rule tends to fail for these, but by including distractors into the score map, such as the *foot*, can accidentally overcome this issue. We believe that this insight will be helpful for designing future weakly-supervised localization methods.

We additionally investigate the question whether the eigenvalues, which are produced by spectral clustering, can be leveraged to automatically identify classes with distractors. For this purpose we sort all 1000 classes by the second smallest eigenvalue given by spectral clustering in increasing order and study how the cumulative improvement of localization quality varies when micro-annotation is collected only for parts of the classes. Figure 6(b) depicts the curve, with the fraction of annotated classes on the x-axis and the fraction of localization improvement on the y-axis. One can see that the eigenvalues may be used to ef-

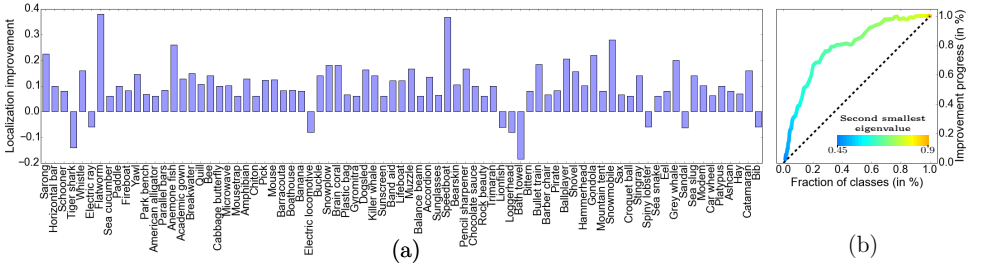

Figure 6: (a) Improvement of the localization scores by applying micro-annotation technique (for 79 classes with biggest changes); (b) Visualization of the trade off between the fraction of annotated classes and the fraction of overall improvement.

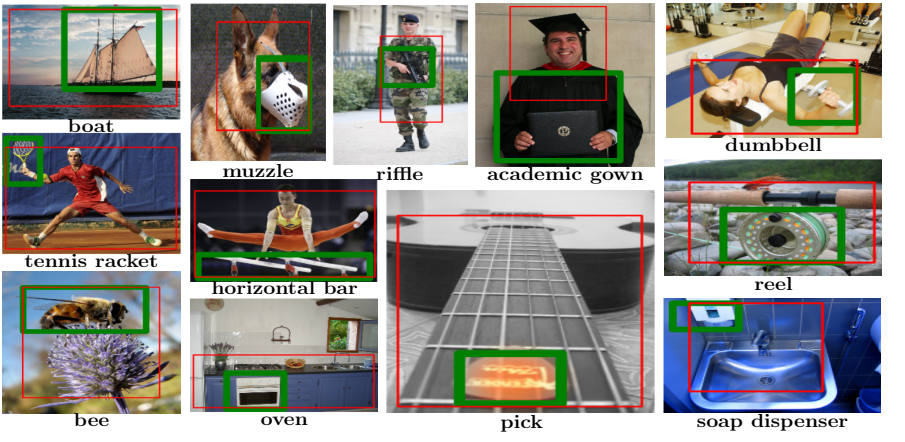

Figure 7: Typical mistakes (red boxes) of the baseline approach on ILSVRC *val* that are corrected by our method (green boxes). In particular, we demonstrate the following cases of foreground/background confusion: *boat/water*, *muzzle/dog*, *riffle/soldier*, *academic gown/academic hat*, *dumbbell/sportsman*, *tennis racket/tennis player*, *horizontal bar/gymnast*, *pick/guitar*, *reel/rod*, *bee/flower*, *oven/kitchen*, *soap dispenser/shell*.

ficiently trade off annotation effort for localization performance. For example, it is sufficient to annotate 15% of all classes to obtain 50% of overall localization improvement, or 50% of the all classes for 85% of improvement.

Semantic image segmentation. Micro-annotation can also improve weakly-supervised semantic image segmentation. We follow the protocol of the *PASCAL VOC* challenges [8]: the goal is to produce segmentation masks by assigning one of 21 labels (20 semantic classes or background) to each pixel of an image. The evaluation metric is the mean intersection-over-union scores across all labels.

We are again interested in the weakly-supervised setting where models are trained only from per-image class annotation. Following the common practice, we use the *PASCAL VOC2012* data with augmentation from [10]. In total, the training set (*train*) has 10,582 images. We report results on the validation part (*val*, 1449 images), and the test part (*test*, 1456 images). Because the ground truth annotations for the *test* set are not public, we rely

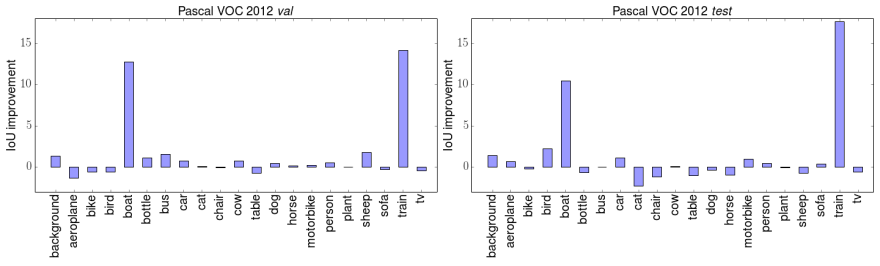

Figure 8: Improvement of the intersection-over-union scores by applying micro-annotation.

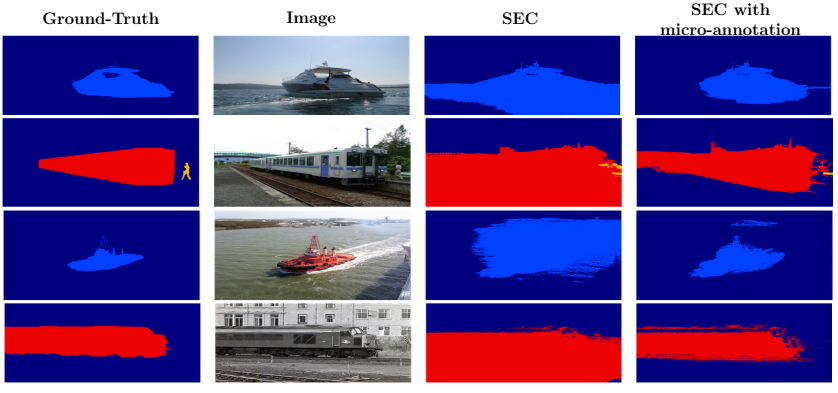

Figure 9: Examples of predicted segmentations masks without and with micro-annotation.

on the independent evaluation server[2] to obtain numerical results for this data.

The state-of-the-art technique for weakly-supervised image segmentation from image level labels is *SEC* [14]. Internally, it relies on *CAM* localization score maps, as introduced in the previous section, also based on a modified VGG-16 network [27] (but with different modifications), see [14] for details. In its original form, SEC achieves average intersection-over-union scores of 50.7% (val) and 51.7% (test).

Improving semantic image segmentation. We obtain mid-level pattern clusters for the 20 semantic classes of *PASCAL VOC* following the protocol as in the previous section. After annotating the clusters (which requires just a few minutes) we found two classes that have significant distractors: *boat* and *train*. Thus, we apply our method for these classes and retrain the *SEC* model using the improved localization score maps. The per-class numerical evaluation for the *val* and *test* sets is presented in Figure 8. We observe significant improvement of the intersection-over-union metric for the *boat* class (12.8% on *val*, 10.5% on *test*) and for the *train* class (14.2% on *val*, 17.7% on *test*). The performance for the other classes does not change significantly, only small perturbations occur due to the shared feature representation learned by the deep network. Overall, we achieve 52.2% and 53.0% mean intersection-over-union for the *val* and *test* sets, and improvement of 1.5% and 1.3% percent over the original *SEC* method. For a visual comparison of the segmentations predicted by the baseline and our approach see Figure 9.

[2] http://host.robots.ox.ac.uk:8080/

Reproducibility. We implemented the proposed method in `python` using the *caffe* [] deep learning framework. The trained models and code are publicly available [3].

5 Conclusion

Weakly-supervised localization techniques have the inherent problem of confusing objects of interest with consistently co-occuring distractors. In this paper we present a micro-annotation technique that substantially alleviates this problem. Our key insight is that objects and distractors can be distinguished from each other because they form different clusters in the distributed representation learned by a deep network. We derive an annotation technique that requires only a few mouse clicks of user interaction per class and we propose an algorithm for incorporating the acquired annotation into the localization procedure.

Experiments on the *ILSVRC* 2014 and *PASCAL* 2012 demonstrate that the proposed micro-annotation method improves results further even for the state-of-the-art for weakly-supervised object localization and image segmentation.

Acknowledgments. This work was funded in parts by the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013)/ERC grant agreement no 308036. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the GPUs used for this research.

References

[1] Loris Bazzani, Alessandro Bergamo, Dragomir Anguelov, and Lorenzo Torresani. Self-taught object localization with deep networks. In *WACV*, 2016.

[2] Hakan Bilen, Marco Pedersoli, and Tinne Tuytelaars. Weakly supervised object detection with posterior regularization. In *BMVC*, 2014.

[3] Hakan Bilen, Marco Pedersoli, and Tinne Tuytelaars. Weakly supervised object detection with convex clustering. In *CVPR*, 2015.

[4] Steve Branson, Catherine Wah, Florian Schroff, Boris Babenko, Peter Welinder, Pietro Perona, and Serge Belongie. Visual recognition with humans in the loop. In *ECCV*, 2010.

[5] Ramazan Gokberk Cinbis, Jakob Verbeek, and Cordelia Schmid. Multi-fold MIL training for weakly supervised object localization. In *CVPR*, 2014.

[6] Jia Deng, Jonathan Krause, and Li Fei-Fei. Fine-grained crowdsourcing for fine-grained recognition. In *CVPR*, 2013.

[7] Thomas Deselaers, Bogdan Alexe, and Vittorio Ferrari. Localizing objects while learning their appearance. In *ECCV*, 2010.

[8] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The PASCAL visual object classes (VOC) challenge. *IJCV*, 2010.

[9] Alexander Freytag, Erik Rodner, and Joachim Denzler. Selecting influential examples: Active learning with expected model output changes. In *ECCV*, 2014.

[3] http://pub.ist.ac.at/~akolesnikov/micro

[10] Bharath Hariharan, Pablo Arbelaez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *ICCV*, 2011.

[11] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093v1*, 2014.

[12] Ajay J. Joshi, Fatih Porikli, and Nikolaos Papanikolopoulos. Multi-class active learning for image classification. In *CVPR*, 2009.

[13] Ashish Kapoor, Kristen Grauman, Raquel Urtasun, and Trevor Darrell. Active learning with gaussian processes for object categorization. In *ICCV*, 2007.

[14] Alexander Kolesnikov and Christoph H. Lampert. Seed, expand and constrain: Three principles for weakly-supervised image segmentation. *ECCV*, 2016.

[15] Cornelius Lanczos. An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. *Journal of Research of the National Bureau of Standards*, 1950.

[16] Quoc V. Le, Marc A. Ranzato, Rajat Monga, Matthieu Devin, Kai Chen, Greg S. Corrado, Jeffrey Dean, and Andrew Ng. Building high-level features using large scale unsupervised learning. In *ICML*, 2012.

[17] Dim P. Papadopoulos, Jasper R.R. Uijlings, Frank Keller, and Vittorio Ferrari. We don't need no bounding-boxes: Training object class detectors using only human verification. *CVPR*, 2016.

[18] George Papandreou, Liang-Chieh Chen, Kevin P. Murphy, and Alan L. Yuille. Weakly- and semi-supervised learning of a deep convolutional network for semantic image segmentation. In *ICCV*, 2015.

[19] Deepak Pathak, Philipp Krähenbühl, and Trevor Darrell. Constrained convolutional neural networks for weakly supervised segmentation. In *ICCV*, 2015.

[20] Deepak Pathak, Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully convolutional multi-class multiple instance learning. In *ICLR*, 2015.

[21] Pedro O. Pinheiro and Ronan Collobert. From image-level to pixel-level labeling with convolutional networks. In *CVPR*, 2015.

[22] Guo-Jun Qi, Xian-Sheng Hua, Yong Rui, Jinhui Tang, and Hong-Jiang Zhang. Two-dimensional active learning for image classification. In *CVPR*, 2008.

[23] Carsten Rother, Tom Minka, Andrew Blake, and Vladimir Kolmogorov. Cosegmentation of image pairs by histogram matching-incorporating a global constraint into MRFs. In *CVPR*, 2006.

[24] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *IJCV*, 2015.

[25] Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.

[26] Marcel Simon and Erik Rodner. Neural activation constellations: Unsupervised part model discovery with convolutional networks. In *ICCV*, 2015.

[27] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[28] Hyun Oh Song, Ross Girshick, Stefanie Jegelka, Julien Mairal, Zaid Harchaoui, and Trevor Darrell. On learning to localize objects with minimal supervision. In *ICML*, 2014.

[29] Hyun Oh Song, Yong Jae Lee, Stefanie Jegelka, and Trevor Darrell. Weakly-supervised discovery of visual pattern configurations. In *NIPS*, 2014.

[30] Manuela Vasconcelos, Nuno Vasconcelos, and Gustavo Carneiro. Weakly supervised top-down image segmentation. In *CVPR*, 2006.

[31] Jakob Verbeek and Bill Triggs. Region classification with Markov field aspect models. In *CVPR*, 2007.

[32] Alexander Vezhnevets and Joachim M. Buhmann. Towards weakly supervised semantic segmentation by means of multiple instance and multitask learning. In *CVPR*, 2010.

[33] Alexander Vezhnevets, Vittorio Ferrari, and Joachim M. Buhmann. Weakly supervised semantic segmentation with a multi-image model. In *ICCV*, 2011.

[34] Alexander Vezhnevets, Vittorio Ferrari, and Joachim M. Buhmann. Weakly supervised structured output learning for semantic segmentation. In *CVPR*, 2012.

[35] Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 2007.

[36] Catherine Wah, Steve Branson, Pietro Perona, and Serge Belongie. Multiclass recognition and part localization with humans in the loop. In *ICCV*, 2011.

[37] Catherine Wah, Grant Horn, Steve Branson, Subhransu Maji, Pietro Perona, and Serge Belongie. Similarity comparisons for interactive fine-grained categorization. In *CVPR*, 2014.

[38] Chong Wang, Kaiqi Huang, Weiqiang Ren, Junge Zhang, and Steve Maybank. Large-scale weakly supervised object localization via latent category learning. *IEEE T-IP*, 2015.

[39] Jia Xu, Alexander G. Schwing, and Raquel Urtasun. Tell me what you see and I will show you where it is. In *CVPR*, 2014.

[40] Jia Xu, Alexander G. Schwing, and Raquel Urtasun. Learning to segment under various forms of weak supervision. In *CVPR*, 2015.

[41] Wei Zhang, Sheng Zeng, Dequan Wang, and Xiangyang Xue. Weakly supervised semantic segmentation for social images. In *CVPR*, 2015.

[42] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Object detectors emerge in deep scene CNNs. In *ICLR*, 2015.

[43] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, 2016.