

# Seed, Expand and Constrain: Three Principles for Weakly-Supervised Image Segmentation

Alexander Kolesnikov and Christoph H. Lampert

IST Austria

{akolesnikov,ch1}@ist.ac.at

**Abstract.** We introduce a new loss function for the weakly-supervised training of semantic image segmentation models based on three guiding principles: to *seed* with weak localization cues, to *expand* objects based on the information about which classes can occur in an image, and to *constrain* the segmentations to coincide with object boundaries. We show experimentally that training a deep convolutional neural network using the proposed loss function leads to substantially better segmentations than previous state-of-the-art methods on the challenging PASCAL VOC 2012 dataset. We furthermore give insight into the working mechanism of our method by a detailed experimental study that illustrates how the segmentation quality is affected by each term of the proposed loss function as well as their combinations.

**Keywords:** Weakly-supervised image segmentation, deep learning

## 1 Introduction

Computer vision research has recently made tremendous progress. Many challenging vision tasks can now be solved with high accuracy, assuming that sufficiently much annotated data is available for training. Unfortunately, collecting large labeled datasets is time consuming and typically requires substantial financial investments. Therefore, the creation of training data has become a bottleneck for the further development of computer vision methods. Unlabeled visual data, however, can be collected in large amounts in a relatively fast and cheap manner. Therefore, a promising direction in the computer vision research is to develop methods that can learn from unlabeled or partially labeled data.

In this paper we focus on the task of semantic image segmentation. Image segmentation is a prominent example of an important vision task, for which creating annotations is especially costly: as reported in [29], manually producing segmentation masks requires several worker-minutes per image. Therefore, a large body of previous research studies how to train segmentation models from weaker forms of annotation.

A particularly appealing setting is to learn image segmentation models using training sets with only per-image labels, as this form of weak supervision can be collected very efficiently. However, there is currently still a large performance gap between models trained from per-image labels and models trained from

full segmentations masks. In this paper we demonstrate that this gap can be substantially reduced compared to the previous state-of-the-art techniques.

We propose a new composite loss function for training convolutional neural networks for the task of weakly-supervised image segmentation. Our approach relies on the following three insights:

- Image classification neural networks, such as AlexNet [19] or VGG [33], can be used to generate reliable object localization cues (**seeds**), but fail to predict the exact spatial extent of the objects. We incorporate this aspect by using a **seeding loss** that encourages a segmentation network to match localization cues but that is agnostic about the rest of the image.
- To train a segmentation network from per-image annotation, a global pooling layer can be used that aggregates segmentation masks into image-level label scores. The choice of this layer has large impact on the quality of segmentations. For example, max-pooling tends to underestimate the size of objects while average-pooling tends to overestimate it [26]. We propose a **global weighted rank pooling** that is leveraged by **expansion loss** to expand the object seeds to regions of a reasonable size. It generalizes max-pooling and average pooling and outperforms them in our empirical study.
- Networks trained from image-level labels rarely capture the precise boundaries of objects in an image. Postprocessing by fully-connected conditional random fields (CRF) at test time is often insufficient to overcome this effect, because once the networks have been trained they tend to be confident even about misclassified regions. We propose a new **constrain-to-boundary loss** that alleviates the problem of imprecise boundaries already at training time. It strives to **constrain** predicted segmentation masks to respect low-level image information, in particular object boundaries.

We name our approach **SEC**, as it is based on three principles: **S**eed, **E**xpand and **C**onstrain. We formally define and discuss the individual components of the SEC loss function in Section 3. In Section 4 we experimentally evaluate it on the PASCAL VOC 2012 image segmentation benchmark, showing that it substantially outperforms the previous state-of-the-art techniques under the same experimental settings. We also provide further insight by discussing and evaluating the effect of each of our contributions separately through additional experiments.

## 2 Related work

Semantic image segmentation, i.e. assigning a semantic class label to each pixel of an image, is a topic of relatively recent interest in computer vision research, as it required the availability of modern machine learning techniques, such as discriminative classifiers [31, 5] or probabilistic graphical models [28, 21]. As the creation of fully annotated training data poses a major bottleneck to the further improvement of these systems, weakly supervised training methods were soon proposed in order to save annotation effort. In particular, competitive methods

were developed that only require partial segmentations [37,11] or object bounding boxes [20,52,8] as training data.

A remaining challenge is, however, to learn segmentation models from just image-level labels [35,36]. Existing approaches fall into three broad categories. *Graph-based models* infer labels for segments or superpixels based on their similarity within or between images [47,48,46,43,27]. Variants of *multiple instance learning* [1] train with a per-image loss function, while internally maintaining a spatial representation of the image that can be used to produce segmentation masks [38,39,40]. Methods in the tradition of *self-training* [30] train a fully-supervised model but create the necessary pixel-level annotation using the model itself in an EM-like procedure [44,45,49]. Our SEC approach contains aspects of the latter two approaches, as it makes use of a per-image loss as well as per-pixel loss terms.

In terms of segmentation quality, currently only methods based on deep convolutional networks [19,33] are strong enough to tackle segmentation datasets of difficulty similar to what fully-supervised methods can handle, such as the PASCAL VOC 2012 [9], which we make use of in this work. In particular, *MIL-FCN* [25], *MIL-ILP* [26] and the approaches of [4,18] leverage deep networks in a multiple instance learning setting, differing mainly in their pooling strategies, i.e. how they convert their internal spatial representation to per-image labels. *EM-Adapt* [23] and *CCNN* [24] rely on the self-training framework and differ in how they enforce the consistency between the per-image annotation and the predicted segmentation masks. *SN-B* [41] adds additional steps for creating and combining multiple object proposals. As far as possible, we provide an experimental comparison to these methods in Section 4.

### 3 Weakly supervised segmentation from image-level labels

In this section we present a technical description of our approach. We denote the space of images by  $\mathcal{X}$ . For any image  $X \in \mathcal{X}$ , a segmentation mask  $Y$  is a collection,  $(y_1, \dots, y_n)$ , of semantic labels at  $n$  spatial locations. The semantic labels belong to a set  $\mathcal{C} = \mathcal{C}' \cup \{c^{\text{bg}}\}$  of size  $k$ , where  $\mathcal{C}'$  is a set of all foreground labels and  $c^{\text{bg}}$  is a background label. We assume that the training data,  $\mathcal{D} = \{(X_i, T_i)\}_{i=1}^N$ , consists of  $N$  images,  $X_i \in \mathcal{X}$ , where each image is weakly annotated by a set,  $T_i \subset \mathcal{C}'$ , of foreground labels that occur in the image. Our goal is to train a deep convolutional neural network  $f(X; \theta)$ , parameterized by  $\theta$ , that models the conditional probability of observing any label  $c \in \mathcal{C}$  at any location  $u \in \{1, 2, \dots, n\}$ , i.e.  $f_{u,c}(X; \theta) = p(y_u = c|X)$ . For brevity we will often omit the parameters  $\theta$  in our notation and write  $f(X; \theta)$  simply as  $f(X)$ .

#### 3.1 The SEC loss for weakly supervised image segmentation

Our approach for learning the parameters,  $\theta$ , of the segmentation neural network relies on minimizing a loss function that has three terms. The first term,  $L_{\text{seed}}$ ,

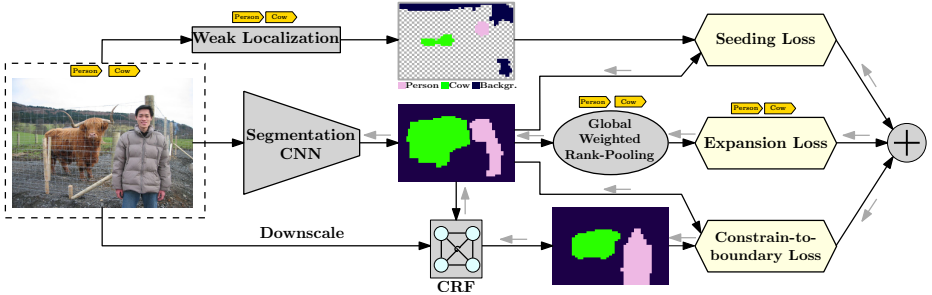


Fig. 1: A schematic illustration of SEC that is based on minimizing a composite loss function consisting of three terms: *seeding loss*, *expansion loss* and *constrain-to-boundary loss*. See Section 3 for details.

provides localization hints to the network, the second term,  $L_{\text{expand}}$ , penalizes the network for predicting segmentation masks with too small or wrong objects, and the third term,  $L_{\text{constrain}}$ , encourages segmentations that respect the spatial and color structure of the images. Overall, we propose to solve the following optimization problem for parameter learning:

$$\min_{\theta} \sum_{(X,T) \in \mathcal{D}} [L_{\text{seed}}(f(X;\theta), T) + L_{\text{expand}}(f(X;\theta), T) + L_{\text{constrain}}(X, f(X;\theta))]. \quad (1)$$

In the rest of this section we explain each loss term in detail. A schematic overview of the setup can be found in Figure 1.

**Seeding loss with localization cues.** Image-level labels do not explicitly provide any information about the position of semantic objects in an image. Nevertheless, as was noted in many recent research papers [22,50,32,3], deep image classification networks that were trained just from image-level labels, may be successfully employed to retrieve cues on object localization. We call this procedure *weak localization* and illustrate it in Figure 2.

Unfortunately, localization cues typically are not precise enough to be used as full and accurate segmentation masks. However, these cues can be very useful to guide the weakly-supervised segmentation network. We propose to use a *seeding loss* to encourage predictions of the neural network to match only “landmarks” given by the weak localization procedure while ignoring the rest of the image. Suppose that  $S_c$  is a set of locations that are labeled with class  $c$  by the weak localization procedure. Then, the *seeding loss*  $L_{\text{seed}}$  has the following form:

$$L_{\text{seed}}(f(X), T, S_c) = -\frac{1}{\sum_{c \in T} |S_c|} \sum_{c \in T} \sum_{u \in S_c} \log f_{u,c}(X). \quad (2)$$

Note that for computing  $L_{\text{seed}}$  one needs the weak localization sets,  $S_c$ , so that many existing techniques from the literature can be used, essentially, as

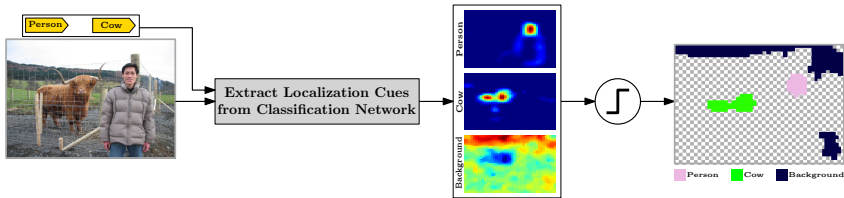


Fig. 2: The schematic illustration of the weak localization procedure.

*black boxes*. In this work, we rely on [50] for weakly localizing foreground classes. However, this method does not provide a direct way to select confident background regions, therefore we use the gradient-based saliency detection method from [32] for this purpose. We provide more details on the weak localization procedure in Section 4.

**Expansion loss with global weighted rank pooling.** To measure if a segmentation mask is consistent with the image-level labels one can aggregate segmentation scores into classification scores and apply the standard loss function for multi-label image classification. In the context of weakly-supervised segmentation/detection various techniques were used by researches to aggregate score maps into a classification scores. The most prominent ones are *global max-pooling* (GMP) [22] that assigns any class  $c$  in any image  $X$  a score of  $\max_{u \in \{1, \dots, n\}} f_{u,c}(X)$  and *global average-pooling* [50] that assigns it a score of  $\frac{1}{n} \sum_{u=1}^n f_{u,c}(X)$ .

Both ways of aggregation have been successfully used in practice. However, they have their own drawbacks. For classes which are present in an image GMP only encourages the response for a single location to be high, while GAP encourages all responses to be high. Therefore, GMP results in a segmentation network that often underestimates the sizes of objects, while network trained using GAP, in contrast, often overestimates them. Our experiments in Section 4 support this claim empirically.

In order to overcome these drawbacks we propose a *global weighted rank-pooling* (GWRP), a new aggregation technique, which can be seen as a generalization of GMP and GAP. GWRP computes a weighted average score for each class, where weights are higher for more promising locations. This way it encourages objects to occupy a certain fraction of an image, but, unlike GAP, is less prone to overestimating object sizes.

Formally, let an index set  $I^c = \{i_1, \dots, i_n\}$  define the descending order of prediction scores for any class  $c \in \mathcal{C}$ , i.e.  $f_{i_1,c}(x) \geq f_{i_2,c}(x) \geq \dots \geq f_{i_n,c}(x)$  and let  $0 < d_c \leq 1$  be a decay parameter for class  $c$ . Then we define the GWRP classification scores,  $G_c(f(X), d_c)$ , for an image  $X$ , as following:

$$G_c(f(X); d_c) = \frac{1}{Z(d_c)} \sum_{j=1}^n (d_c)^{j-1} f_{i_j,c}(X), \text{ where } Z(d_c) = \sum_{j=1}^n (d_c)^{j-1}. \quad (3)$$

Note, that for  $d_c = 0$  GWRP turns into GMP (adopting the convention that  $0^0 = 1$ ), and for  $d_c = 1$  it is identical to GAP. Therefore, GWRP generalizes both approaches and the decay parameter can be used to interpolate between the behavior of both extremes.

In principle, the decay parameter could be set individually for each class and each image. However, this would need prior knowledge about how large objects of each class typically are, which is not available in the weakly supervised setting. Therefore, we only distinguish between three groups: for object classes that occur in an image we use a decay parameter  $d_+$ , for object classes that do not occur we use  $d_-$ , and for background we use  $d_{\text{bg}}$ . We will discuss how to choose their values in Section 4.

In summary, the *expansion loss* term is

$$L_{\text{expand}}(f(X), T) = -\frac{1}{|T|} \sum_{c \in T} \log G_c(f(X); d_+) \quad (4)$$

$$- \frac{1}{|\mathcal{C}' \setminus T|} \sum_{c \in \mathcal{C}' \setminus T} \log(1 - G_c(f(X); d_-)) - \log G_{c_{\text{bg}}}(f(X); d_{\text{bg}}).$$

**Constrain-to-boundary loss.** The high level idea of the *constrain-to-boundary loss* is to penalize the neural network for producing segmentations that are discontinuous with respect to spatial and color information in the input image. Thereby, it encourages the network to learn to produce segmentation masks that match up with object boundaries.

Specifically, we construct a fully-connected CRF,  $Q(X, f(X))$ , as in [17], with unary potentials given by the logarithm of the probability scores predicted by the segmentation network, and pairwise potentials of fixed parametric form that depend only on the image pixels. We downscale the image  $X$ , so that it matches the resolution of the segmentation mask, produced by the network. More details about the choice of the CRF parameters are given in Section 4. We then define the *constrain-to-boundary loss* as the mean KL-divergence between the outputs of the network and the outputs of the CRF, i.e.:

$$L_{\text{constrain}}(X, f(X)) = \frac{1}{n} \sum_{u=1}^n \sum_{c \in \mathcal{C}} Q_{u,c}(X, f(X)) \log \frac{Q_{u,c}(X, f(X))}{f_{u,c}(X)}. \quad (5)$$

This construction achieves the desired effect, since it encourages the network output to coincide with the CRF output, which itself is known to produce segmentation that respect image boundaries. An illustration of this effect can be seen in Figure 1.

### 3.2 Training

The proposed network can be trained in an end-to-end way using back-propagation, provided that the individual gradients of all layers are available. For computing gradients of the fully-connected CRF we employ the procedure from [34], which

was successfully used in the context of semantic image segmentation. Figure 1 illustrates the flow of gradients for the backpropagation procedure with gray arrows.

## 4 Experiments

In this section we validate our proposed loss function experimentally, including a detailed study of the effects of its different terms.

### 4.1 Experimental setup

**Dataset and evaluation metric.** We evaluate our method on the PASCAL VOC 2012 image segmentation benchmark, which has 21 semantic classes, including background [9]. The dataset images are split into three parts: training (*train*, 1464 images), validation (*val*, 1449 images) and testing (*test*, 1456 images). Following the common practice we augment the training part by additional images from [10]. The resulting *trainaug* set has 10,582 weakly annotated images that we use to train our models. We compare our approach with other approaches on both *val* and *test* parts. For the *val* part, ground truth segmentation masks are available, so we can evaluate results of different experiments. We therefore use this data also to provide a detailed study of the influence of the different components in our approach. The ground truth segmentation masks for the *test* part are not publicly available, so we use the official PASCAL VOC evaluation server to obtain quantitative results. As evaluation measure we use the standard PASCAL VOC 2012 segmentation metric: mean intersection-over-union (mIoU).

**Segmentation network.** As a particular choice for the segmentation architecture, in this paper we use *DeepLab-CRF-LargeFOV* from [6], which is a slightly modified version of the 16-layer VGG network [33]. The network has inputs of size 321x321 and produces segmentation masks of size 41x41, see [6] for more details on the architecture. We initialize the weights for the last (prediction) layer randomly from a normal distribution with mean 0 and variance 0.01. All other convolutional layers are initialized from the publicly available VGG model [33]. Note, that in principle, our loss function can be combined with any deep convolutional neural network.

**Localization networks.** The localization networks for the foreground classes and the background class are also derived from the standard VGG architecture. In order to improve the localization performance, we finetune these networks for solving a multilabel classification problem on the *trainaug* data. Due to space limitations we provide exact details on these networks and optimization parameters in the supplementary material.

Note, that in order to reduce the computational effort and memory consumption required for training SEC it is possible to precompute the localization cues. If precomputed cues are available SEC imposes no additional overhead for evaluating and storing the localization networks at training time.

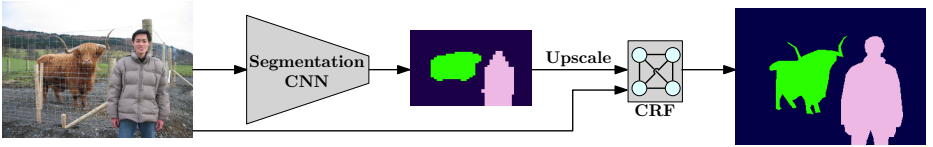


Fig. 3: The schematic illustration of our approach at test time.

**Optimization.** For training the network we use the batched stochastic gradient descent (SGD) with parameters used successfully in [6]. We run SGD for 8000 iterations, the batch size is 15 (reduced from 30 to allow simultaneous training of two networks), the dropout rate is 0.5 and the weight decay parameter is 0.0005. The initial learning rate is 0.001 and it is decreased by a factor of 10 every 2000 iterations. Overall, training on a GeForce TITAN-X GPU takes 7-8 hours, which is comparable to training times of other models, reported, e.g., in [23,24].

**Decay parameters.** The GWRP aggregation requires specifying the decay parameters,  $d_-$ ,  $d_+$  and  $d_{bg}$ , that control the weights for aggregating the scores produced by the network. Inspired by the previous research [23,24] we do so using the following rules-of-thumb that express prior beliefs about natural images:

- for semantic classes that are not present in the image we want to predict as few pixels as possible. Therefore, we set  $d_- = 0$ , which corresponds to GMP.
- for semantic classes that are present in the image we suggest that the top 10% scores represent 50% of the overall aggregated score. For our 41x41 masks this roughly corresponds to  $d_+ = 0.996$ .
- for the background we suggest that the top 30% scores represent 50% of the overall aggregated score, resulting in  $d_{bg} = 0.999$ .

**Fully-connected CRF at training time.** In order to enforce the segmentation network to respect the boundaries of objects already at training time we use a fully-connected CRF [17]. As parameters for the pairwise interactions, we use the default values from the authors’ public implementation, except that we multiply all spatial distance terms by 12 to reflect the fact that we downsampled the original image in order to match the size of the predicted segmentation mask.

**Inference at test time.** Our segmentation neural network is trained to produce probability scores for all classes and locations, but the spatial resolution of a predicted segmentation mask is lower than the original image. Thus, we upscale the predicted segmentation mask to match the size of the input image, and then apply a fully-connected CRF [17] to refine the segmentation. This is a common practice, which was previously employed, e.g., in [23,24,6]. Figure 3 shows a schematic illustration of our inference procedure at test time.

**Reproducibility.** In our experiments we rely on the *caffe* deep learning framework [13] in combination with a *python* implementation of the SEC loss. The code and pretrained models are publicly available<sup>1</sup>.

<sup>1</sup> <https://github.com/kolesman/SEC>



## 4.2 Results

**Numeric Results.** Table 1 compares the performance of our weakly supervised approach with previous approaches that are trained in the same setup, i.e. using only images from PASCAL VOC 2012 and only image-level labels. It shows that SEC substantially outperforms the previous techniques. On the *test* data, where the evaluation is performed by an independent third party, the PASCAL VOC evaluation server, it achieves 13.5% higher mean intersection-over-union score than the state-of-the-art approaches with new best scores on 20 out of 21 semantic classes. On the validation data, for which researchers can compute scores themselves, SEC improves over the state-of-the-art by 14.1%, and achieves new best scores on 19 out of the 21 classes.

Results of other weakly-supervised methods on PASCAL VOC and the fully-supervised variant of DeepLab are summarized in Table 2. We provide these results for reference but emphasize that they should not simply be compared to Table 1, because the underlying methods were trained on different (and larger) training sets or were given additional forms of weak supervision, e.g. user clicks. Some entries need further explanation in this regard: [23] reports results for the EM-Adapt model when trained with weak annotation for multiple image crops. The same model was reimplemented and trained with only per-image supervision in [24], so these are the values we report in Table 1. The results reported for SN\_B [41] and the *seg* variant of the MIL+ILP+SP [26] are incomparable to others because they were obtained with help of *MCG* region proposals [2] that were trained in a fully supervised way on PASCAL VOC data. Similarly, MIL+ILP+SP-*bb* makes use of bounding box proposals generated by the BING method [7] that was trained using PASCAL VOC bounding box annotation.

Note that we do include the *sppxl* variant of MIL+ILP+SP in Table 1. While it is trained on roughly 760.000 images of the ImageNet dataset, we do not consider this an unfair advantage compared to our and other methods, because those implicitly benefit from ImageNet images as well when using pretrained classification networks for initialization.

**Qualitative Results.** Figure 4 illustrates typical successful segmentations. It shows that our method can produce accurate segmentations even for non-trivial images and recover fine details of the boundary. Figure 5 illustrates some failure cases. As is typical for weakly-supervised systems, SEC has problems segmenting objects that occur almost always in front of the same background, e.g. boats on water, or trains on tracks. We addressed this problem recently in follow-up work [15]. A second failure mode is that object regions can be segmented correctly, but assigned wrong class labels. This is actually quite rare for SEC, which we attribute to the fact that the DeepLab network has a large field-of-view and therefore can make use of the full image when assigning labels. Finally, it can also happen that segmentations cover only parts of objects. This is likely due to imperfections of the weak localization cues that tend to reliably detect only the most discriminative parts of an object, e.g. the face of a person. This might not be sufficient to segment the complete object, however,

Table 1: Results on PASCAL VOC 2012 (mIoU in %) for weakly-supervised semantic segmentation with only per-image labels.

PASCAL VOC 2012 <i>val</i> set	[4] (Img+Obj)	[14] (stage1)	EM-Adapt (re-impl. of [24])	CCNN [24]	MIL+ILP +SP-sppxl <sup>†</sup> [26]	SEC (proposed)	PASCAL VOC 2012 <i>test</i> set	MIL-FCN [25]	CCNN [24]	MIL+ILP +SP-sppxl <sup>†</sup> [26]	Region score pooling [18]	SEC (proposed)
background		71.7*	67.2	68.5	77.2	<b>82.4</b>	background	≈71 <sup>‡</sup>	74.7	≈74 <sup>‡</sup>	<b>83.5</b>	
aeroplane		30.7*	29.2	25.5	37.3	<b>62.9</b>	aeroplane	24.2	38.8	33.1	<b>56.4</b>	
bike		<b>30.5*</b>	17.6	18.0	18.4	26.4	bike	19.9	19.8	21.7	<b>28.5</b>	
bird		26.3*	28.6	25.4	25.4	<b>61.6</b>	bird	26.3	27.5	27.7	<b>64.1</b>	
boat		20.0*	22.2	20.2	<b>28.2</b>	27.6	boat	18.6	21.7	17.7	<b>23.6</b>	
bottle		24.2*	29.6	36.3	31.9	<b>38.1</b>	bottle	38.1	32.8	38.4	<b>46.5</b>	
bus		39.2*	47.0	46.8	41.6	<b>66.6</b>	bus	51.7	40.0	55.8	<b>70.6</b>	
car		33.7*	44.0	47.1	48.1	<b>62.7</b>	car	42.9	50.1	38.3	<b>58.5</b>	
cat		50.2*	44.2	48.0	50.7	<b>75.2</b>	cat	48.2	47.1	57.9	<b>71.3</b>	
chair		17.1*	14.6	15.8	12.7	<b>22.1</b>	chair	15.6	7.2	13.6	<b>23.2</b>	
cow		29.7*	35.1	37.9	45.7	<b>53.5</b>	cow	37.2	44.8	37.4	<b>54.0</b>	
diningtable		22.5*	24.9	21.0	14.6	<b>28.3</b>	diningtable	18.3	15.8	<b>29.2</b>	28.0	
dog		41.3*	41.0	44.5	50.9	<b>65.8</b>	dog	43.0	49.4	43.9	<b>68.1</b>	
horse		35.7*	34.8	34.5	44.1	<b>57.8</b>	horse	38.2	47.3	39.1	<b>62.1</b>	
motorbike		43.0*	41.6	46.2	39.2	<b>62.3</b>	motorbike	52.2	36.6	52.4	<b>70.0</b>	
person		36.0*	32.1	40.7	37.9	<b>52.5</b>	person	40.0	36.4	44.4	<b>55.0</b>	
plant		29.0*	24.8	30.4	28.3	<b>32.5</b>	plant	33.8	24.3	30.2	<b>38.4</b>	
sheep		34.9*	37.4	36.3	44.0	<b>62.6</b>	sheep	36.0	44.5	48.7	<b>58.0</b>	
sofa		23.1*	24.0	22.2	19.6	<b>32.1</b>	sofa	21.6	21.0	26.4	<b>39.9</b>	
train		33.2*	38.1	38.8	37.6	<b>45.4</b>	train	33.4	31.5	31.8	<b>38.4</b>	
tv/monitor		33.2*	31.6	36.9	35.0	<b>45.3</b>	tv/monitor	38.3	41.3	36.3	<b>48.3</b>	
average	32.2	33.6*	33.8	35.3	36.6	<b>50.7</b>	average	25.7	35.6	35.8	38.0	<b>51.7</b>

(\*results from unpublished/not peer-reviewed manuscripts, <sup>†</sup>trained on ImageNet, <sup>‡</sup>value inferred from average)

Table 2: Summary results (mIoU %) for other methods on PASCAL VOC 2012. Note: the values in this table are not directly comparable to Table 1, as they were obtained under different experimental conditions.

method	<i>val</i>	<i>test</i>	comments
DeepLab [6]	67.6	70.3	fully supervised training
STC [42]	49.8*	51.2*	trained on Flickr
TransferNet [12]	52.1	51.2	trained on MS COCO; additional supervision: from segmentation mask of other classes
[4] (1Point)	42.7	–	additional supervision: 1 click per class
[4] (AllPoints-weighted)	43.4	–	additional supervision: 1 click per instance
[4] (squiggle)	49.1	–	additional supervision: 1 squiggle per class
EM-Adapt [23]	38.2	39.6	uses weak labels of multiple image crops
SN_B [41]	41.9	43.2	uses MCG region proposals (see text)
MIP+ILP+SP-seg [26]	42.0	40.6	trained on ImageNet, MCG proposals (see text)
MIL+ILP+SP-bb [26]	37.8	37.0	trained on ImageNet, BING proposals (see text)

(\*results from manuscripts that are currently unpublished/not peer-reviewed)

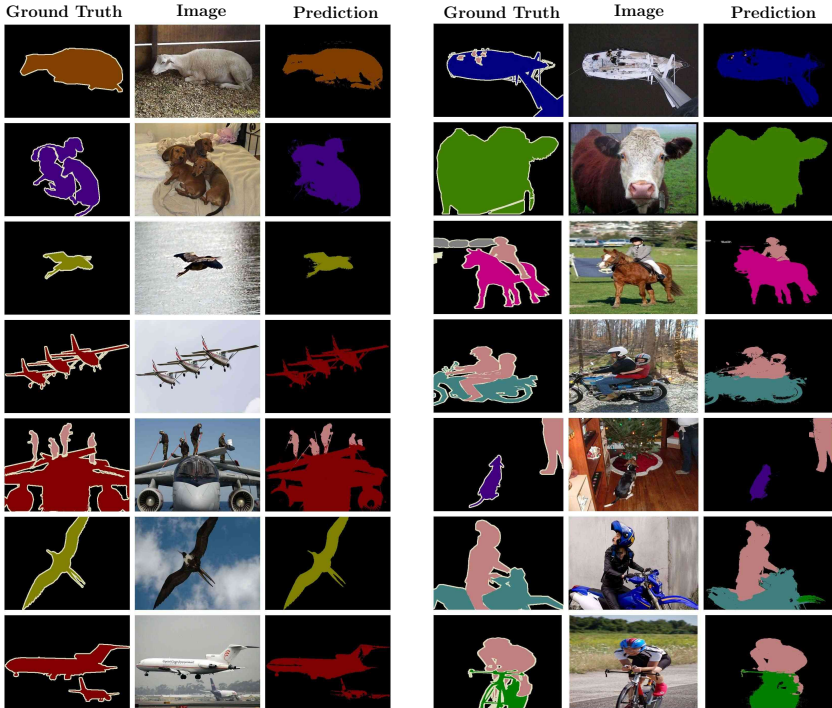


Fig. 4: Examples of predicted segmentations (*val* set, successful cases).

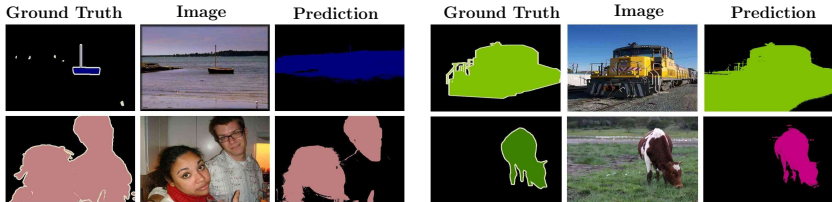


Fig. 5: Examples of predicted segmentations (*val* set, failure cases).

especially when objects overlap each other or consist of multiple components of very different appearance.

### 4.3 Detailed Discussion

To provide additional insight into the working mechanisms of the SEC loss function, we performed two further sets of experiments on the *val* data. First, we analyze different global pooling strategies, and second, we perform an ablation study that illustrates the effect of each of the three terms in the proposed loss function visually as well as numerically.

**Effect of global pooling strategies.** As discussed before, the quality of segmentations depends on which global pooling strategy is used to convert seg-

mentation mask into per-image classification scores. To quantify this effect, we train three segmentation networks from weak supervision, using either GMP, GAP or GWRP as aggregation methods for classes that are present in the image. For classes that are not present we always use GMP, i.e. we penalize any occurrence of these classes. In Figure 6 we demonstrate visual results for every pooling strategy and report two quantities: the fraction of pixels that are predicted to belong to a foreground (fg) class, and the segmentation performance as measured by mean IoU. We observe that GWRP outperforms the other method in terms of segmentation quality and the fractions of predicted foreground pixels supports our earlier hypothesis: the model trained with GMP tends to underestimate object sizes, while the model trained with GAP on average overestimates them. In contrast, the model trained with GWRP, produces segmentations in which objects are, on average, close to the correct size<sup>2</sup>.

**Effect of the different loss terms.** To investigate the contribution of each term in our composite loss function we train segmentation networks with loss functions in which different terms of the SEC loss were omitted. Figure 7 provides numerical results and illustrates typical segmentation mistakes that occur when certain loss terms are omitted. Best results are achieved when all three loss terms are present. However, the experiments also allow us to draw two interesting additional conclusions about the interaction between the loss terms.

**Semi-supervised loss and large field-of-view.** First, we observe that having  $L_{\text{seed}}$  in the loss function is crucial to achieve competitive performance. Without this loss term our segmentation network fails to reflect the localization of objects in its predictions, even though the network does match the global label statistics rather well. See the third column of Figure 7 for the illustration of this effect.

We believe that this effect can be explained by the large (378x378) field-of-view (FOV) of the segmentation network<sup>3</sup>: if an object is present in an image, then the majority of the predicted scores may be influenced by this object, no matter where object is located. This helps in predicting the right class labels, but can negatively affect the localization ability. Other researchers addressed this problem by explicitly changing the architecture of the network in order to reduce its field-of-view [23]. However, networks with a small field-of-view are less powerful and often fail to recognize which semantic labels are present on an image. We conduct an additional experiment (see the supplementary material for details) that confirm that SEC with a small (211x211) field-of-view network performs clearly worse than with the large (378x378) field-of-view network, see Figure 8 for numeric results and visual examples. Thus, we conclude that the seeding loss provides the necessary localization guidance that enables the large field-of-view network to still reliably localize objects.

<sup>2</sup> Note that these experiments were done after the network architecture and parameters were fixed. In particular, we did not tune the decay parameters for this effect.

<sup>3</sup> We report the theoretical fields-of-view inferred from the network architecture. The empirical field-of-view that is actually used by the network can be smaller [51].

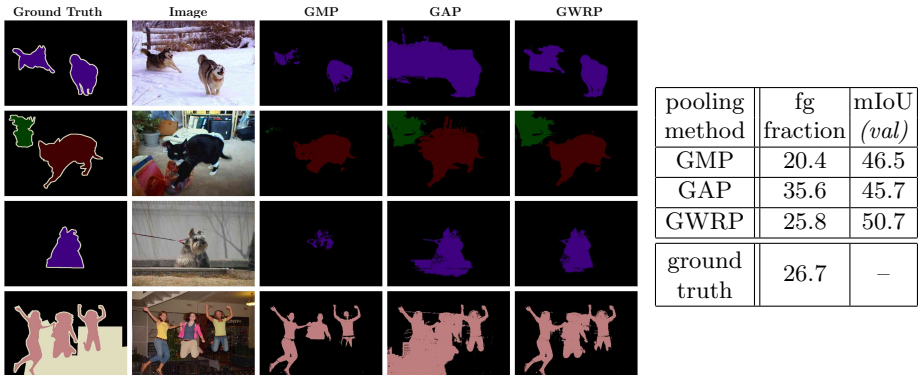


Fig. 6: Results on the *val* set and examples of segmentation masks for models trained with different pooling strategies.

**Effects of the expansion and constrain-to-boundary losses.** By construction, the constrain-to-boundary loss encourages nearby regions of similar color to have the same label. However, this is often not enough to turn the weak localization cues into segmentation masks that cover a whole object, especially if the object consists of visually dissimilar parts, such as people wearing clothes of different colors. See the sixth column of Figure 7 for an illustration of this effect.

The expansion loss, based on GWRP, suppresses the prediction of classes that are not meant to be in the image, and it encourages classes that are in the image to have reasonable sizes. When combined with the seeding loss, the expansion loss actually results in a drop in performance. The fifth column of Figure 7 shows an explanation of this: objects sizes are generally increased, but the additionally predicted regions do not match the image boundaries.

In combination, the seeding loss provides reliable seed locations, the expansion loss acts as a force to enlarge the segmentation masks to a reasonable size, and the constrain-to-boundary loss constrains the segmentation mask to line up with image boundaries, thus integrating low-level image information. The result are substantially improved segmentation masks as illustrated in the last column of Figure 7.

## 5 Conclusion

We propose a new loss function for training deep segmentation networks when only image-level labels are available. We demonstrate that our approach outperforms previous state-of-the-art methods by a large margin when used under the same experimental conditions and provide a detailed ablation study.

We also identify potential directions that may help to further improve weakly-supervised segmentation performance. Our experiments show that knowledge about object sizes can dramatically improve the segmentation performance. SEC

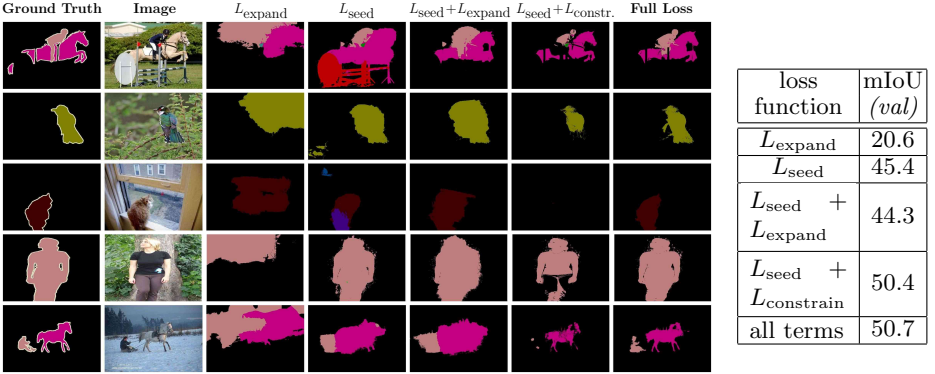


Fig. 7: Results on the *val* set and examples of segmentation masks for models trained with different loss functions.

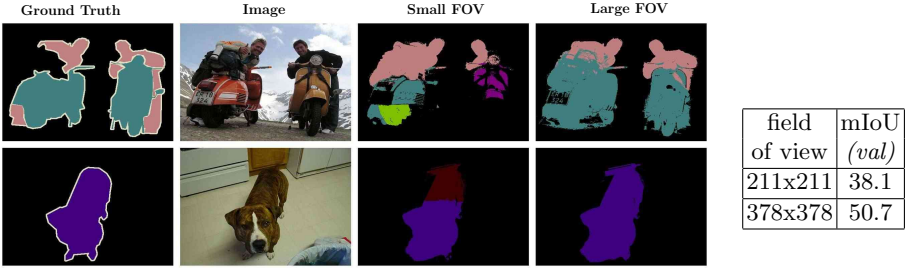


Fig. 8: Results on the *val* set and examples of segmentation masks for models with small or large field-of-views.

readily allows incorporating size information through decay parameters, but a procedure for estimating object sizes automatically would be desirable. A second way to improve the performance would be stronger segmentation priors, for example about shape or materials. This could offer a way to avoid mistakes that are currently typical for weakly-supervised segmentation networks, including ours, for example that boats are confused with the water in their background.

**Acknowledgments.** This work was funded by the European Research Council under the European Unions 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. We also thank Vittorio Ferrari for helpful feedback.

## References

1. Andrews, S., Tsochantaridis, I., Hofmann, T.: Support vector machines for multiple-instance learning. In: NIPS (2002)

2. Arbeláez, P., Pont-Tuset, J., Barron, J., Marques, F., Malik, J.: Multiscale combinatorial grouping. In: CVPR (2014)
3. Bazzani, L., Bergamo, A., Anguelov, D., Torresani, L.: Self-taught object localization with deep networks. In: WACV (2016)
4. Bearman, A., Russakovsky, O., Ferrari, V., Fei-Fei, L.: What's the point: Semantic segmentation with point supervision. ECCV (2016)
5. Carreira, J., Sminchisescu, C.: CPMC: Automatic object segmentation using constrained parametric min-cuts. IEEE T-PAMI 34(7) (2012)
6. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Semantic image segmentation with deep convolutional nets and fully connected CRFs. In: ICLR (2015)
7. Cheng, M.M., Zhang, Z., Lin, W.Y., Torr, P.H.S.: BING: Binarized normed gradients for objectness estimation at 300fps. In: CVPR (2014)
8. Dai, J., He, K., Sun, J.: BoxSup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In: ICCV (2015)
9. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The PASCAL visual object classes (VOC) challenge. IJCV 88(2) (2010)
10. Hariharan, B., Arbelaez, P., Bourdev, L., Maji, S., Malik, J.: Semantic contours from inverse detectors. In: ICCV (2011)
11. He, X., Zemel, R.S.: Learning hybrid models for image annotation with partially labeled data. In: NIPS (2009)
12. Hong, S., Oh, J., Lee, H., Han, B.: Learning transferrable knowledge for semantic segmentation with deep convolutional neural network. CVPR (2016)
13. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. arXiv preprint arXiv:1408.5093v1 (2014)
14. Kim, H., Hwang, S.: Scale-invariant feature learning using deconvolutional neural networks for weakly-supervised semantic segmentation. arXiv preprint arXiv:1602.04984v2 (2016), <http://arxiv.org/abs/1602.04984v2>
15. Kolesnikov, A., Lampert, C.H.: Improving weakly-supervised object localization by micro-annotation. BMVC (2016)
16. Kolesnikov, A., Lampert, C.H.: Seed, expand and constrain: Three principles for weakly-supervised image segmentation. arXiv preprint arXiv:1603.06098 (2016), <http://arxiv.org/abs/1603.06098>
17. Krähenbühl, P., Koltun, V.: Efficient inference in fully connected CRFs with gaussian edge potentials. In: NIPS (2011)
18. Krapac, J., Šegvic, S.: Weakly-supervised semantic segmentation by redistributing region scores to pixels. GCPR (2016)
19. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: NIPS (2012)
20. Liu, S., Yan, S., Zhang, T., Xu, C., Liu, J., Lu, H.: Weakly supervised graph propagation towards collective image parsing. IEEE T-MM 14(2) (2012)
21. Nowozin, S., Gehler, P.V., Lampert, C.H.: On parameter learning in CRF-based approaches to object class image segmentation. In: ECCV (2010)
22. Oquab, M., Bottou, L., Laptev, I., Sivic, J.: Is object localization for free? – weakly-supervised learning with convolutional neural networks. In: CVPR. pp. 685–694 (2015)
23. Papandreou, G., Chen, L.C., Murphy, K.P., Yuille, A.L.: Weakly- and semi-supervised learning of a deep convolutional network for semantic image segmentation. In: ICCV (2015)

24. Pathak, D., Krähenbühl, P., Darrell, T.: Constrained convolutional neural networks for weakly supervised segmentation. In: ICCV (2015)
25. Pathak, D., Shelhamer, E., Long, J., Darrell, T.: Fully convolutional multi-class multiple instance learning. In: ICLR (2015)
26. Pinheiro, P.O., Collobert, R.: From image-level to pixel-level labeling with convolutional networks. In: CVPR (2015)
27. Pourian, N., Karthikeyan, S., Manjunath, B.: Weakly supervised graph based semantic segmentation by learning communities of image-parts. In: CVPR (2015)
28. Rabinovich, A., Vedaldi, A., Galleguillos, C., Wiewiora, E., Belongie, S.: Objects in context. In: ICCV (2007)
29. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet large scale visual recognition challenge. IJCV 115(3) (2015)
30. Scudder, H.J.: Probability of error of some adaptive pattern-recognition machines. IEEE T-IT 11(3) (1965)
31. Shotton, J., Winn, J., Rother, C., Criminisi, A.: Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In: ECCV (2006)
32. Simonyan, K., Vedaldi, A., Zisserman, A.: Deep inside convolutional networks: Visualising image classification models and saliency maps. In: ICLR (2014)
33. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: ICLR (2015)
34. Toyoda, T., Hasegawa, O.: Random field model for integration of local information and global information. IEEE T-PAMI 30(8) (2008)
35. Vasconcelos, M., Vasconcelos, N., Carneiro, G.: Weakly supervised top-down image segmentation. In: CVPR (2006)
36. Verbeek, J., Triggs, B.: Region classification with Markov field aspect models. In: CVPR (2007)
37. Verbeek, J., Triggs, W.: Scene segmentation with CRFs learned from partially labeled images. In: NIPS (2008)
38. Vezhnevets, A., Buhmann, J.M.: Towards weakly supervised semantic segmentation by means of multiple instance and multitask learning. In: CVPR (2010)
39. Vezhnevets, A., Ferrari, V., Buhmann, J.M.: Weakly supervised semantic segmentation with a multi-image model. In: ICCV (2011)
40. Vezhnevets, A., Ferrari, V., Buhmann, J.M.: Weakly supervised structured output learning for semantic segmentation. In: CVPR (2012)
41. Wei, Y., Liang, X., Chen, Y., Jie, Z., Xiao, Y., Zhao, Y., Yan, S.: Learning to segment with image-level annotations. Pattern Recognition (2016)
42. Wei, Y., Liang, X., Chen, Y., Shen, X., Cheng, M., Zhao, Y., Yan, S.: STC: a simple to complex framework for weakly-supervised semantic segmentation. arXiv preprint arXiv:1509.03150v1 (2015), <http://arxiv.org/abs/1509.03150>
43. Xie, W., Peng, Y., Xiao, J.: Weakly-supervised image parsing via constructing semantic graphs and hypergraphs. In: Multimedia (2014)
44. Xu, J., Schwing, A.G., Urtasun, R.: Tell me what you see and I will show you where it is. In: CVPR (2014)
45. Xu, J., Schwing, A.G., Urtasun, R.: Learning to segment under various forms of weak supervision. In: CVPR (2015)
46. Zhang, L., Gao, Y., Xia, Y., Lu, K., Shen, J., Ji, R.: Representative discovery of structure cues for weakly-supervised image segmentation. IEEE T-MM 16(2) (2014)



47. Zhang, L., Song, M., Liu, Z., Liu, X., Bu, J., Chen, C.: Probabilistic graphlet cut: Exploiting spatial structure cue for weakly supervised image segmentation. In: CVPR (2013)
48. Zhang, L., Yang, Y., Gao, Y., Yu, Y., Wang, C., Li, X.: A probabilistic associative model for segmenting weakly supervised images. IEEE T-IP 23(9) (2014)
49. Zhang, W., Zeng, S., Wang, D., Xue, X.: Weakly supervised semantic segmentation for social images. In: CVPR (2015)
50. Zhou, B., Khosla, A., A., L., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: CVPR (2016)
51. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Object detectors emerge in deep scene CNNs. In: ICLR (2015)
52. Zhu, J., Mao, J., Yuille, A.L.: Learning from weakly supervised data by the expectation loss SVM (e-SVM) algorithm. In: NIPS (2014)

## 6 Supplementary material

In this section we provide technical details of the weak localization procedure (section 3.1) and specify how the network with the small field-of-view was derived (section 4.3).

### 6.1 Localization networks

The seeding loss SEC relies on weak localization cues. As was noted in many recent papers [22,50,32,3], localization cues may be produced by leveraging a deep convolutional neural network that is trained for solving an image classification task. We use two different approaches, one for localizing the foreground classes and the other for the background class.

Specifically, for localizing the foreground classes we employ the technique from [50]. As an underlying classification network we use the standard VGG network, initialized from the publicly available model [33]. The VGG architecture is slightly modified in order to make the methodology from [50] applicable. In particular, we implement the following changes into VGG:

- the last two fully-connected layers, *fc6* and *fc7*, are substituted with randomly initialized convolutional layers, which have 1024 output channels and kernels of size 3.
- the output of the last convolutional layer is followed by a global average pooling layer and then by a fully-connected prediction layer with 20 outputs (the number of foreground semantic classes in PASCAL VOC)

Additionally, in order to increase the spatial resolution of the last convolutional layer of the network, we increase the input size to 321x321 and omit the two last max-pooling layers, *pool4* and *pool5*. The resulting network is finetuned with multilabel logistic loss on the *trainaug* part of the PASCAL VOC 2012 dataset (we use the same optimization parameters as in Section 4). Then the network is used to provide class-specific localization heat maps. In order to produce localization cues for every foreground class we threshold the corresponding heat map by 20% of its maximum value, as was suggested in [50]. The resulted localization cues are stacked together into a single weak localization mask, as shown on Figure 2. It may happen that localization cues are in conflict, when different classes are assigned to the same location. We use simple rule to resolve these conflicts: during the stacking process classes that occupy a smaller fraction of an image have priority over classes that occupy a bigger fraction of an image.

For localizing background we rely on the alternative technique from [32]. We also use the VGG as the underlying network. It is modified to have input resolution of size 321x321 and to have a prediction layer with 20 outputs. Analogously to [6], we also change the number of output channels in the fully-connected layers, *fc6* and *fc7*, from 4096 to 1024. We finetune the network using the same procedure as for the network for localizing the foreground classes. Following the gradient-based procedure from [32], we utilize the finetuned network to produce

class-independent saliency maps. The saliency maps can be quite noisy, so we smoothen them by the median filter with a window of size  $3 \times 3$ . Finally, 10% of the least salient locations in each image are selected as background cues.

## 6.2 Small field-of-view

The *Deeplab-Large-FOV* neural network achieves particularly wide field-of-view by utilizing convolutions with “holes”, which were recently suggested in the context of semantic image segmentation in [23]. In order to derive the closest architecture, but with a small field-of-view, we substitute convolutions with “holes” by the standard convolutional layers. This leads to a measurable drop in the size of the field-of-view: from 378x378 to 211x211.

