

# Leveraging Pretrained Image Classifiers for Language-Based Segmentation

David Golub

Stanford University

golubd@cs.stanford.edu

Roberto Martín-Martín

Stanford University

roberto.martinmartin@stanford.edu

Ahmed El-Kishky

University of Illinois at Urbana-Champaign

elkishk2@illinois.edu

Silvio Savarese

Stanford University

ssavarese@stanford.org

## Abstract

Current semantic segmentation models cannot easily generalize to new object classes unseen during train time: they require additional annotated images and retraining. We propose a novel segmentation model that injects visual priors into semantic segmentation architectures, allowing them to segment out new target labels without retraining. As visual priors, we use the activations of pretrained image classifiers, which provide noisy indications of the spatial location of both the target object and distractor objects in the scene. We leverage language semantics to obtain these activations for a target label unseen by the classifier. Further experiments show that the visual priors obtained via language semantics for both relevant and distracting objects are key to our performance

## 1. Introduction

Semantic segmentation is the task of annotating image pixels with labels that correspond to one or several classes of interest. Recently, end-to-end models based on fully convolutional neural networks (FCNs) [15, 22, 5, 4] have significantly improved accuracy for semantic segmentation, holding the state-of-the-art on multiple benchmarks [6, 11, 45]. The caveat is, though, that training such models for a new class of interest requires large amounts of image annotations either in the form of pixels that correspond to the class in the image [22, 5] or as bounding boxes around the objects of the class [17, 7]. These annotations have to be done manually and are thus costly to obtain. Due to the annotation burden, high-quality segmentation models only exist for a few sufficiently annotated classes.

Previous methods have proposed two approaches to extend the success of FCN-based segmentation to other classes requiring less (or at least easier) annotations.

<https://sites.google.com/stanford.edu/cls-seg>

This work has been supported by an ONR MURI award (W911NF-15-1-0479).

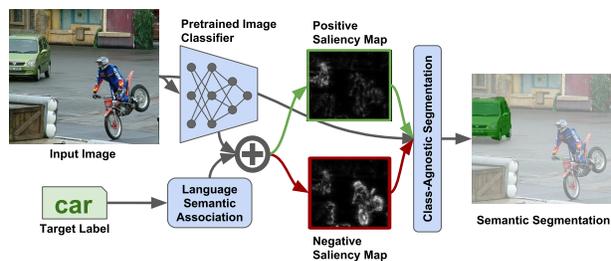


Figure 1: Our approach for semantic segmentation; Given an image and a target label (*car*), we leverage language semantics to extract visual priors from the pretrained classifier and generate a positive (green frame) and a negative (red frame) activation map, indicating regions of the image related and unrelated to the target; We use the activation maps as attention channels for a novel class-agnostic segmentation method that segments the original image

The first of these are *weakly supervised* approaches; these class of methods try to reduce the complexity on the annotation process by leveraging image-level labels, a class label for an entire image [15, 42, 31]. The second family of approaches apply *few-shot* semantic segmentation [34, 30, 43]; these approaches alleviate the annotation burden by learning to segment out a target class with the support of a few additional images. However, both of these approaches fail to leverage the semantic similarity new unseen classes and previously seen classes and require the addition of new manually-labelled images.

Recent work [41] has shown that the representations learned by deep neural networks trained for different visual tasks are strongly correlated, and that transferring information between them is possible. Inspired by this idea, we aim to exploit a model trained for image classification for semantic segmentation of new or semantically-related classes. Prior methods [42, 31] have already shown that this transfer of information from classification to segmentation is possible. They use techniques such as Guided-Backprop [39], GradCAM [37], or SmoothGrad [33] to obtain noisy but interpretable object localizations in the form of saliency maps

Semantic Segmentation Method	New Target Label	Annotation Level	Knowledge Transfer	Pretrained Classifier Role	Semantics Role
Regular (e.g. [22, 20, 4])	✗	Pixel	✗	Initialize Weights	-
Weakly Supervised [1, 42]	✗	Image	✗	Initialize Weights	Training Data
		Image and Pixel			-
Few-Shot [30, 34, 9, 43]	✓ (>1 an.)	Image and Pixel	from Classifier via Weights	Initialize Weights	-
Ours	✓ (0 an.)	Image and Pixel	from Classifier via Language Semantics	Initialize Weights and Generate Saliency Maps	Additional Input

Table 1: Comparison of families of semantic segmentation methods; Only our approach can segment semantically images for a new label without any additional annotation. All methods use pretrained classifiers to initialize weights, but ours use them also to generate saliency maps. Some weakly supervised segmentation models with image-level labels [42, 1] use saliency maps as training data while we use it as additional input

from image-level labels [1] (see Fig. 1). However, these methods train a new image classifier on the same dataset used for segmentation so that the labels for segmentation can also be back-propagated into the classifier. This procedure excludes them from the our setting since they cannot segment previously unseen, semantically related labels at test time, requiring retraining for each new label.

In contrast, we propose to achieve semantic segmentation using off-the-shelf classifiers already trained on large image-level annotated datasets like ImageNet [8]. These pretrained classifiers<sup>1</sup> are widely and easily accessible<sup>2</sup>. We use the activations of the neurons of a classifier to create saliency maps, noisy spatial indications of relevant areas in the image.

In our regime, our test-time target labels are not directly visible during training, although semantically similar concepts may have been seen by the off-the-shelf pretrained classifier. Therefore, we propose the use of language semantics to extract visual priors from the pretrained classifier, connecting the new unseen test-time labels to labels known by the pretrained classifier. We use an automatic WordNet-based [25] process that maps the object class of interest (*target label*) to a set of semantically-related labels contained in the classifier’s training set that we can use to generate a saliency map. Saliency maps generated only from semantically related labels often include regions with distracting objects. Therefore, we also map the target label to adversary semantically unrelated labels present in the image (*negative labels*) that generate additional *negative* saliency maps. These two saliency maps contain spatial information of semantically related (positive) and unrelated (negative) objects in the image. Therefore, we attend on these saliency maps in a novel class-agnostic segmentation model (Fig. 1), that produces the final segmentation mask for the image.

Our full approach segments an input image for a given test label without any additional human annotation for labels inside or outside of the classifier’s training set, and thus

<sup>1</sup>We use the term *pretrained classifier* as it is standard in the community for an off-the-shelf available model. However, we emphasize that we do not *post-train* these classifiers with new data; our method uses them as-is.

<sup>2</sup>A simple web search on “pre-trained image classifier” yields 19 variants of models pretrained on ImageNet.

addresses the few-shot segmentation problem without re-training the model for unseen test-time labels. In summary, our main two contributions are as follows:

- We present a novel model for semantic segmentation that extracts visual priors from a pretrained image classifier using language semantics. Our model outperforms previous one-shot and five-shot methods of [43] on the proposed partitions of Pascal VOC [34] by 4% and 3.2% mIOU on Pascal VOC classes without additional human-annotated support sets at test time. Our model also outperforms a foreground-background baseline and a model with a saliency map from a random label (no semantics) by 16.3% mIOU.
- We introduce a novel attention-based class-agnostic segmentation model that attends on two additional channels to the input image: a positive and a negative channel. The positive channel indicates regions possibly containing the target object while the negative channel indicates visually distracting areas, generated in our segmentation model from the activation of a pretrained classifier to related and unrelated semantic labels to the target label. Our model using positive and negative labels outperforms the best model using only semantically related labels by 8.4% mIOU, with a gain of over 20% mIOU on complex classes such as aeroplane, bus, car, dog, and bottle.

## 2. Related Work

Semantic segmentation is a long-standing task in computer vision, with applications in scene understanding [19, 12, 38], autonomous navigation [10, 26, 44, 35], and activity tracking [28, 27]. Despite its longevity, this task remains challenging because object appearances and their visual boundaries vary widely depending on factors such as lighting, pose, scale and noise.

Recently, methods based on convolutional neural networks have shown impressive progress on this task [15, 16, 20, 22]. These networks are trained in a fully-supervised manner on large pixel-level annotated datasets such as Pascal VOC [11], City Scapes [6] and MSCOCO [21]. The dependency on thousands or more fine-grained, pixel-level

annotated images is a limitation to generalize these models to new categories.

Researchers have proposed two general procedures to alleviate the annotation burden to train semantic segmentation methods, *weak-supervision* and *few-shot* learning, that we review in the following.

### 2.1. Weakly Supervised Semantic Segmentation

Weakly supervised (or partially supervised) semantic segmentation relaxes the pixel-level labeling dependency by using bounding boxes [17, 16, 1] or image-level annotations [18, 42, 15, 3] to generate automatic segmentation annotations for a target label. These annotations are then used to fine-tune a segmentation network that is originally trained on a densely annotated dataset.

Khoreva *et al.* [17] leverage bounding boxes to generate candidate segmentations for objects of interest. These segmentations are refined further with low-level vision techniques such as GrabCut [32] and a class-agnostic semantic segmentation model. While the class-agnostic model and use of GrabCut is similar to our model, this technique still requires numerous bounding box annotations, in addition to a high-quality object detector trained on all target labels.

Several methods reduce the granularity of annotations further by training their own classifier with image-level class labels, and use them to generate saliency maps [42, 31, 15]. Two recent approaches [42, 31] use the saliency maps as noisy segmentation labels to finetune existing semantic segmentation models. Hong *et al.* [15] add a segmentation network on top of label-specific activations in the second-to-last layer of their trained classifier. This network is then finetuned on human-annotated segmentations of the target class. While conceptually related to our method, these approaches do not use pretrained classifiers but require a training process that assumes that all possible target labels are known during training. Moreover, once learned, these segmentation models need to be retrained for new labels and therefore cannot be applied to segmenting unseen labels at test-time.

### 2.2. Few-shot Semantic Segmentation

Inspired by the recent success of few-shot and meta-learning approaches for object classification [23, 40] and detection [2, 30], few-shot semantic segmentation methods reduce the annotation burden to several or sometimes a single annotation at pixel or bounding-box level for a new label. First introduced in [34] and developed further in [30, 43], even for multiple labels at once in [9], few-shot segmentation methods are usually based on conditional models that ground their predictions on a small set of pairs of images and segmentation masks.

Zhang *et al.* [30] propose a novel pooling strategy to leverage the similarity between the support set of segmenta-

tion masks and the target image. This method significantly outperforms prior work (5.2% mIOU improvement). However, this method requires at least one human-annotated image at pixel-level with labels for each target, and cannot be applied using a test label only.

## 3. Semantic Segmentation Model

The input to our method is a label of interest,  $l$ , and an RGB image,  $I$ , of width  $w$  and height  $h$ . Our model generates a binary semantic segmentation mask,  $M_l \in \{0, 1\}^{w \times h}$ , indicating for each pixel if it belongs to an object of class label  $l$  or not. We decompose the segmentation problem into two main steps as shown in Fig. 2. In the first step, we obtain a positive and a negative saliency map over the input image from an image classification network pretrained on ImageNet (Sec. 3.1). The positive saliency map corresponding to a label is an image  $S_l^+ \in R^{w \times h}$  providing a noisy indication of the relevant image parts to that label. The negative saliency map is an image  $S_l^- \in R^{w \times h}$  indicating image parts that are not relevant to the label, e.g. distracting objects. Since in our segmentation task, the exact target labels are not part of the classifier’s training set, we propose to exploit language semantics with WordNet to link the target label to related and unrelated labels seen by the pretrained classifier.

In the second step, we feed the two saliency maps and the original input image into a semantic segmentation network with attention channels (a modified DeepLabv3 [3] architecture) that outputs a class likelihood image  $P_l \in [0, 1]^{w \times h}$ . Each pixel  $p \in P_l$  indicates the likelihood of the corresponding pixel in the original image to belong to the class label  $l$  described by the saliency maps. Lastly, we post-process the likelihood image using GrabCut [32] and obtain the final segmentation image  $M_l$ .

### 3.1. Semantic Saliency Maps Generation

**Mapping target label to related and unrelated training labels:** While pre-trained image classifiers can cover over several thousand object categories (also called *synsets*), in our segmentation regime we assume that the exact target label was not seen during training, including the training of the classifier. We do, however, assume that labels semantically-related to the target label may have been seen by the pretrained classifier. As such, we propose to leverage language semantics to map the unseen target class to classes seen by the classifier during training.

Given a target label  $l$  and an image classification model pretrained on a set of labels  $L_{\text{train}}$ , our goal is to map  $l$  to  $K$  semantically related labels  $L_{\text{sem}}^+ = [l_1^+, \dots, l_K^+]$ , where  $l_k^+ \in L_{\text{train}}$ . We use a language-semantics mapper to obtain these proxy labels. We will study two types of language-semantics mappers in Sec. 3.2—WordNet [25] and Word2Vec [24]. The language-semantics mapper can gen-

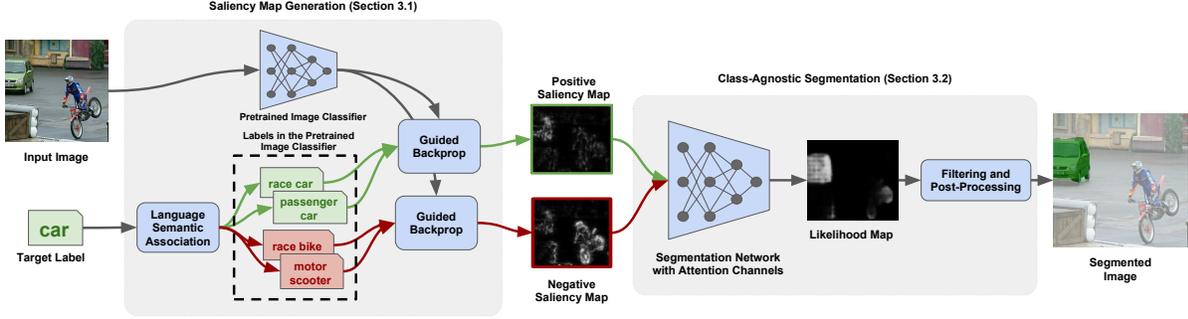


Figure 2: Diagram of our approach for semantic segmentation. At test time, our approach receives a new image and a previously unseen label (*car*). We use a language semantic association to map the target to positive (green) and negative (red) labels in the set of training labels of the pretrained classifier. We then use Guided Backpropagation [39] to extract saliency maps from the pretrained classifier corresponding to the positive and negative labels. The saliency maps are used as attention channels in a segmentation network to generate a likelihood image that is post-processed to generate the final segmentation mask of the unseen label

erate a very large number of proxy labels, larger than  $K$ . This would lead to a very noisy saliency map in the next step. Therefore, we further use the pretrained image classifier to prune the semantic proxy labels. We feed the image to the pretrained classifier and obtain a likelihood score for all proxy labels. We select among the resulting set of labels from the semantic mapper the  $K$  with larger probability of be in the image as given by the image classifier.

Similarly, we would like to map  $l$  to a set  $L_{sem}^-$  of  $K$  semantically unrelated labels that could act as distractors in the image. We assume that distractors are objects detected by the classifier that are not semantically related to  $l$ . Concretely, we use the  $K$  labels of the classifier with highest likelihood score that are not part of  $L_{sem}^+$ .

**Generating positive and negative saliency maps:** Given our proxy positive label set ( $L_{sem}^+$ ) or proxy negative label set ( $L_{sem}^-$ ), we use class activation mappings [39, 37, 33] to extract the corresponding positive or negative semantic saliency map over the image,  $S_l^+$  or  $S_l^-$ . Activation mappings indicate the areas in an image that most actively contribute to a certain classifier decision and are therefore correlated to the active class label. We will use the activation map as noisy location information about the object class.

Since the process is similar for positive and negative labels, we will discuss here for a generic label  $l_k$  and a set  $L_{sem}$ . We first forward propagate the original image,  $I$ , through the pretrained image classifier. Then, we use each proxy label,  $l_k$ , as the target label and backpropagate the error signals back through the network to the input image to get a gradient tensor of size  $\mathbb{R}^{3 \times w \times h}$ . For each pixel location  $i, j$  in the gradient tensor, we perform max pooling over the three channels and normalize the gradients over pixels so they lie in the range  $[0, 1]$ . This process produces the saliency map of a single positive or negative proxy label  $l_k$ :  $S_{l_k} \in \mathbb{R}^{w \times h}$ . Finally, we compute the average sum of the saliency maps from all proxy labels and generate the final saliency map  $S_l \in \mathbb{R}^{w \times h}$ . Repeating this procedure for

$L_{sem}^+$  and  $L_{sem}^-$  lead to the positive and negative saliency maps,  $S_l^+$  and  $S_l^-$ . These saliency maps are passed as attention input to the next step, the class-agnostic semantic segmentation network.

### 3.2. Class-Agnostic Semantic Segmentation with Attention

We propose to use the positive and negative saliency maps as spatial attention to guide a class-agnostic semantic segmentation model. To achieve that we modify a DeepLabv3 [3] architecture to condition the segmentation on the saliency maps,  $S_l^+$  and  $S_l^-$ . The original DeepLabv3 architecture is a segmentation network  $F : \mathbb{R}^{3 \times w \times h} \mapsto \mathbb{R}^{n \times w \times h}$ , where  $n$  is the number of classes. We concatenate the two saliency maps as additional channels to the original image and set  $n = 2$  (foreground or background) to obtain a modified DeepLabv3 network  $F' : \mathbb{R}^{5 \times w \times h} \mapsto \mathbb{R}^{2 \times w \times h}$ . The second channel of the output corresponds to the class likelihood image for the target label,  $P_l$ , while the first channel corresponds to the likelihood image of the background.

The final step is to obtain the binary segmentation mask,  $M_l$ , from the likelihood image,  $P_l$ . A direct way would be to threshold the likelihood. However, finding a single optimal binary threshold is hard because the mean and variance of pixel activations per likelihood image varies significantly across images. We propose instead to use GrabCut [32] to produce our final segmentation mask. GrabCut uses four types of pixel-wise annotations: sure background  $sb$ , probable background  $pb$ , probable foreground  $pf$ , and sure foreground  $sf$ . We can map likelihood values to these four types of pixel annotations. We observe that the highest and smallest values in  $P_l$ ,  $p_{max}$  and  $p_{min}$ , vary per image and per label. Therefore, we design our annotations to be invariant to these shifts. Let  $\delta = p_{max} - p_{min}$ . Given threshold values  $t_{fg}, t_{bg}, t_{unk}$ , we create an annotation image  $A$  where each

Target Label	WordNet Proxy Labels (ours)	Word2Vec Proxy Labels
bottle	beer bottle, <b>bottle-screw</b> , pill bottle, soda bottle, water bottle	beer bottle, <b>bottle-screw</b> , pill bottle, wine bottle, soda bottle
car	<b>car wheel</b> , racer, sports car, streetcar, <b>car mirror</b> , freight car	<b>car wheel</b> , racer, sports car, street car, pickup truck, fire truck
dog	pug, terrier, shepherd, tibetan terrier, <b>monkey dog</b>	terrier, pit bull, toy poodle, wire-haired fox terrier
chair	folding chair, barber chair	folding chair, barber chair, <b>desk, toilet seat, table lamp</b>
cat	tabby cat, tiger cat, siamese cat, <b>cat bear</b>	tabby cat, <b>fox terrier</b> , tiger cat, <b>toy terrier, hamster</b>
train	bullet train	<b>school bus, freight car, streetcar</b> , steam locomotive
sofa	studio couch	<b>folding chair, pillow, desk, bookcase</b> , studio couch

Table 2: Top proxy ImageNet labels via our proposed WordNet-based method vs Word2Vec baseline; number of proxy labels is limited to  $K = 5$ ; words ordered by semantic similarity; **bold** indicates visually distracting proxy labels; Word2Vec links a significantly larger number of distracting labels

pixel  $A(u, v)$  is defined as:

$$\begin{cases} sg & \text{if } P_l(u, v) \leq p_{min} + t_{bg} * \delta \\ pb & \text{if } p_{min} + t_{bg} * \delta \leq P_l(u, v) < p_{min} + t_{unk} * \delta \\ pf & \text{if } p_{min} + t_{unk} * \delta \leq P_l(u, v) < p_{min} + t_{fg} * \delta \\ sf & \text{if } p_{min} + t_{fg} * \delta \leq P_l(u, v) \end{cases}$$

We use GrabCut with  $A$  and obtain the final binary segmentation mask  $M_l$ .

## 4. Experimental Setup

### 4.1. Implementation

**Pretrained Image Classifier:** We use an image classifier pretrained on ImageNet1k as a pretrained model. Pretrained models on this dataset are the most studied and widely available. We use the VGG-19 [36] architecture, as we observe its saliency maps are more concentrated on keypoints in the image as opposed to saliency maps from a ResNet50 architecture. We combine this classifier with Guided-Backpropagation [39] to generate saliency maps.

**Language Semantic Association:** We implement and evaluate two possible semantic mappers: WordNet [25] and Word2Vec [24]. WordNet groups hierarchically words together by their semantic meaning. With WordNet we obtain first, a set of synonym labels,  $[l_1, \dots, l_N]$ , from the target label  $l$ , and then, a set of hypernyms for both the target label and the synonyms. Hypernyms are words that describe more general superclasses (categories) containing the target label and synonyms. The labels used to train image classifiers (e.g. ImageNet1K labels) are generic and often hypernyms of the labels used in image segmentation (e.g. Pascal VOC labels). We extract all labels from the classifier that have a word in common with any of the hypernyms and generate our set of semantic-related proxy labels.

Word2Vec is a family of algorithms to generate word embeddings based on semantics. With Word2Vec we first embed the target labels  $l$  and the labels in the set of possible proxy labels in a shared vector space using 300-dimensional GloVe embeddings [29] trained on the Common Crawl 840B word corpus. For labels that contains multiple words,

we average individual word embeddings into a single vector. Then, we choose the proxy labels with highest cosine similarity to  $l$  as our set of semantic-related proxy labels. For both approaches, we set  $K = 5$  as the maximum number of proxy labels in our experiments.

**Modified DeepLabv3 with Attention Channels** We use an open-source DeepLabv3 implementation and modify it as described in Sec. 3.2. We use Resnet-50 [14] to pre-initialize weights for all of the layers except the first and last layer. In the first layer, the weights for the first three channels are initialized using the pretrained ImageNet classifier. The weights for the fourth and fifth channel are randomly initialized.

### Conditional Fully Convolutional Network (baseline):

We compare our work to a few-shot conditional Fully Convolutional Network (coFCN) [30] architecture that consists of five convolution and pooling layers, followed by one transpose convolution layer to generate the final predictions. The coFCN pools visual features from the annotated support set with outputs of a fully convolutional neural network. Then, these pooled features are run through a transpose convolution to produce final pixel-wise foreground/background probabilities. As in the original paper, we use VGG-16 to pre-initialize weights for all of the layers.

**Post-Processing with GrabCut:** We use the OpenCV implementation of GrabCut and set  $t_{fg} = 0.7$ ,  $t_{unk} = 0.5$ , and  $t_{bg} = 0.15$ . We run GrabCut for five iterations.

### 4.2. Dataset

In our experiments, we use the target labels and images from the Pascal VOC dataset [11]. The Pascal VOC dataset consists of roughly 11,000 images with 20 classes with a broad variety of object types (animals, vehicles, indoor/outdoor items, ...).

As in prior work [34], we use the Semantic Boundaries Dataset [13] and Pascal VOC train set [11] to train our model, and use the Pascal VOC validation set for testing.

$i = 0$	$i = 1$	$i = 2$	$i = 3$			
aeroplane, bicycle, bird, boat, bottle	bus, car, cat, chair, cow	diningtable, dog, horse, motorbike, person	potted plant, sheep, sofa, train, tv/monitor			
		PASCAL-5 $_{i=0}$	PASCAL-5 $_{i=1}$	PASCAL-5 $_{i=2}$	PASCAL-5 $_{i=3}$	Mean
OSLSM [34]: 1-shot		33.6	55.3	40.9	33.5	40.8
coFCN [30]: 1-shot		36.7	50.6	44.9	32.3	41.1
SG [43]: 1-shot		40.2	58.4	48.4	38.4	46.3
OSLSM [34]: 5-shot		35.9	58.1	42.7	39.1	43.9
5-shot-coFCN [30]		37.5	50.0	44.1	33.9	41.4
5-shot-SG [43]		41.9	58.6	48.6	39.4	47.1
SEM-0-C-NONE: No saliency map		39.6	40.3	37.4	31.6	37.2
SEM-1-C-RAND: One saliency map from random label		31.2	31.8	41.8	31.2	34.0
SEM-1-C-GT: One saliency map from target label		37.3	42.8	45.4	43.3	42.2
SEM-2-C-RAND: Saliency maps from target label and random label		40.8	<b>57.9</b>	47.7	38.5	46.2
SEM-2-C-MEAN: Saliency maps from target label and top 5 labels		43.1	56.2	47.12	<b>47.0</b>	48.4
<b>SEM-2-C-NEG</b> : Saliency maps from target label and negative labels		<b>48.7</b>	57.6	<b>48.9</b>	46.0	<b>50.3</b>

Table 3: Mean intersection over union (mIOU) on the partitions described in [34]. We compare against the previously proposed one-shot and few shot models (OSLSM [34], (FCN [30]), (SG [43])). We then compare our proposed approach to a variant that only uses a single saliency map (SEM-1-channel-\*), ablations without not using the generated saliency maps (SEM-0-C-NONE) and random labels (SEM-2-C-RAND, SEM-1-C-RAND)

We partition Pascal VOC into disjoint train/test label splits. Hence, in contrast to classical train/test partitions, our test examples share no labels in common with the train set.

We use the four label partitions from Shaban *et al.* [34] (Table 3, top). In this setup, each partition consists of five to six consecutive labels, sorted in increasing alphabetic order.

### 4.3. Training

The only part to train in our model is the class-agnostic semantic segmentation network with attention (Sec. 3.2). For this model, we use the default hyperparameters of the original DeepLabv3 for training. We use the model trained for 30,000 steps for testing.

## 5. Experiments

### 5.1. WordNet vs. Word2Vec Labels

We conduct five sets of experiments. In the first set, we compare the quality of proxy labels using a WordNet mapping to a Word2Vec embedding. We identified visually distracting labels as labels of objects that indicate a different class that can co-occur with the target label, and therefore, hinders the segmentation results. Table 2 depicts some of the results of the comparison. The top three rows indicate cases when Word2Vec and WordNet extract similar proxy labels. The bottom four rows indicate cases when Word2Vec extracts visually distracting labels. WordNet may generate less than the maximum number of proxy labels,  $K = 5$ , if the existing ImageNet1k labels are not related to the target label.

A limitation of the Word2Net mapping is that it may miss semantically relevant words that are not in the WordNet ontology, e.g. the ImageNet label *steam locomotive* from the label *train*. Nevertheless, we find that Word2Vec consistently produces more visually distracting proxy labels than

the WordNet-based approach. Only 64% of the of the words extracted by Word2Vec are visually distracting as opposed to 18% of the words extracted by WordNet. Based on this results, we use WordNet as the preferred semantic mapping in the rest of our experiments.

### 5.2. Role of GrabCut

In this experiment, we replace GrabCut postprocessing with a static threshold of 0.5 for the *SEM-2-C-NEG* model. Without GrabCut, the performance drops to 48.1 mIOU, a 1.8% improvement over the previous 1-shot state of art.

### 5.3. Comparison Against Prior Work

In the second set of experiments, we evaluate and compare the accuracy of our proposed model to the performance of prior few-shot [34, 43] and one shot methods [30] on the proposed group partitions in [34]. Note that in contrast to these methods, our method does not require any human-annotated images at test time; instead, we only use the label. We report mean intersection over union (mIOU) accuracy of our binary segmentation masks on the test set of Pascal-VOC.

The results of the comparison are depicted in Table 3, per-label results in Table 5, and samples in Fig. 3. Even without any additional human annotations as support set, our proposed method (SEM-2-C-NEG) outperforms the work of [43] on three out of four partitions, and achieves an overall mIOU of 50.3%, a 4% improvement over prior one-shot work [43] that achieved an overall mIOU of 46.3%.

On three out of the four partitions introduced in [34] we outperform previous state-of-the-art, without any human-provided images at test time. Our technique enables semantic segmentation models to extract salient visual cues from noisy saliency maps that segmentation networks are able to leverage to accurately segment out new labels at test

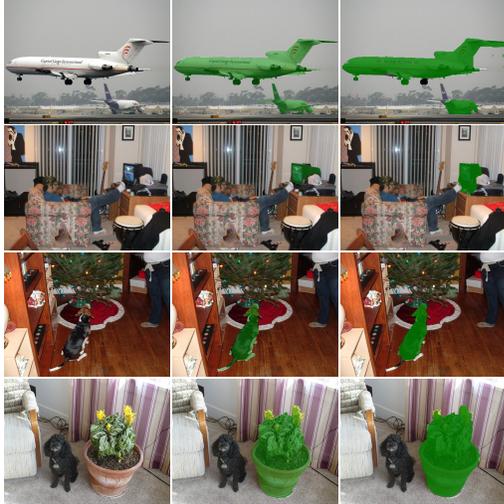


Figure 3: Left: original image; Middle: ground truth segmentation; Right: our resulting segmentation mask; Accurate segmentations for *aeroplane*, *tv-monitor*, *dog* and *potted-plant* (more results in supplementary material)

time. Interestingly, we obtain better results from noisy self-generated saliency maps on the same image than [43] using human-generated ground truth labels on different images.

We observe that performance of the model varies significantly per-partition and per-label. We would like to better understand whether this is caused by the generated saliency maps or the segmentation model that uses them. Therefore, we train an *oracle* segmentation model that uses the ground truth segmentation mask as positive saliency map. The oracle achieves an average mIOU of 95% mIOU, indicating that a large part of the errors in our model would be alleviated by better activation maps. However, the oracle does not reach 100% mIOU which points out that some errors are caused by the class-agnostic segmentation model. Interestingly, the oracle model and our proposed approach mostly suffers on the same target labels: While the median mIOU is over 97%, the oracle model only achieves a 77% mIOU on *bicycle*, 88% mIOU on *pottedplant*, 94% on *chair*, the three worst performing classes of our model. This indicates that these classes are the most difficult to segment for the class-agnostic segmentation model.

We visually inspect failure cases from our model, observing the differences between the likelihood images and the final semantic masks, and find out that the majority of failure cases are due to either 1) GrabCut postprocessing errors (Fig. 6, first row), 2) object detection errors (Fig. 6, second row), or 3) attending on irrelevant but semantically related objects (Fig. 6, third row). We observe that the performance of GrabCut especially degrades when segmenting thin and/or occluded objects that are sparsely connected such as *bicycle*, *pottedplant*, and *chair*, an avenue for improvement in future work.



Figure 4: Qualitative results for our ablation studies. The top row shows results for a 1-channel model with saliency maps from target label *cat* (middle) vs. random label *banana* (left) vs. target label (right); bottom row shows results for a 2-channel model (middle) vs 1-channel model (left) for target label *bottle*.

### 5.4. Role of Saliency Maps and Semantics

In the third set of experiments, we analyze the importance of saliency maps and language semantics in our segmentation model. We use the same train setup as in 5.3, and train three variants of a modified DeepLabv3 network. The first variant is a foreground/background classifier with no saliency maps as input (SEM-0C-NONE), the second uses a saliency map generated from 5 random labels (SEM-1C-RAND), and the last variant uses saliency map generated from a target label (SEM-1C-GT, section 3.1).

**Improvement from saliency maps:** In contrast to prior work [30], our proposed approach significantly outperforms a foreground-background baseline. A model with one saliency map from the target label outperforms the baseline on three out of four partitions, with an overall improvement of 5.6% mIOU, while a model with two saliency maps outperforms the baseline across all partitions, and by 14% mIOU overall. In contrast to a two-saliency-map model, a single-saliency-map model performs worse on partition 0 and only moderately better on partition 1 than the baseline. We hypothesize this is due to the noisy nature of saliency maps; for semantically ambiguous labels such as *car* or *dog*, and small objects such as a *bottle*, a single saliency map may pick up spurious objects in the image and thus misdirect the segmentation model towards an incorrect object.

**Contribution from language semantics:** We now analyze the contribution of the language-semantic mapping of the target labels to proxy labels in the pretrained model. We compare the results of that uses the target label (SEM-1C-GT) against a model that uses saliency maps from random labels (SEM-1C-RAND). We find that a model with saliency maps from random labels performs significantly worse across all partitions than a model with the target label. As illustrated in Fig. 4, saliency maps from a random

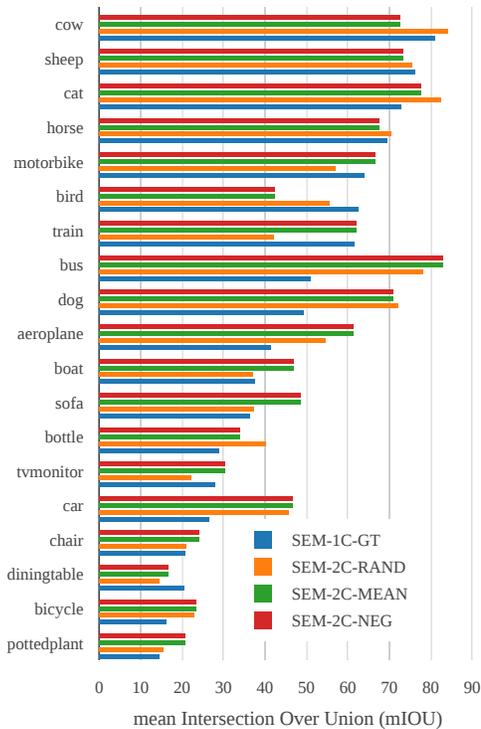


Figure 5: Per-label mIOU of variants of our semantic segmentation models. *1-channel* references to a model that uses one saliency map derived from the target label. *2-channel-random*, *2-channel-mean*, *2-channel-proposed* refer to using a random label, the top 5 labels, and negative labels (3.1) for the second saliency map

label often contain a greater degree of noise and focus on spurious pixels. This leads to less confident outputs from the segmentation network and incorrect segmentations after GrabCut postprocessing. However, as shown in the following section, negative saliency maps from random/distracting labels provide useful semantic information for the model, and performance improves significantly if combined with positive saliency maps from the target label.

### 5.5. Two vs. One Saliency Map

In the fourth set of experiments, we compare variants of our two-saliency-map model to a model that only uses a positive saliency map. In our two-saliency-map model, we experiment generating the negative saliency map with a random label, the mean of the top 5 labels predicted by the classifier for the target image, or the target negative labels (normal model).

The per-partition results of our experiment are reported in Table 3 and per-class results in Table 5. The second saliency map provides complementary semantic information to the first map that our model successfully leverages.

We observe an overall mIOU improvement of 8.3% using two saliency maps over a one-saliency-map model. The largest performance gains are in the classes *car*, *bus*, *dog*, *aeroplane*, *bottle*, and *sofa*, that all have an average mIOU

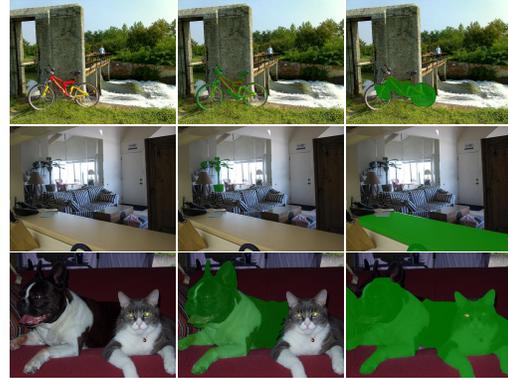


Figure 6: Left: original image; Middle: ground truth segmentation; Right: our resulting segmentation mask; Failure cases for *potted-plant*, *dog* and *bicycle*. The top row shows incorrect object localization, the middle row shows semantically-similar object conflation, and the bottom row shows GrabCut postprocessing errors

improvement of over 20% as seen in Fig. 5 and exemplified in Fig. 4, bottom. These objects often occur in cluttered environments with multiple objects in the scene. Adding a second saliency map improves both *bus* and *car* by a very large margin because 30.2% of images with buses have cars in them. This shows that our method can disambiguate multiple objects in the same scene.

Surprisingly, there is a small degradation for some classes when using two channels. The largest performance degradations are in the classes *bird* and *train*. Images that have these target labels depict typically only the target object in the image. This leads to positive and the negative saliency maps that are very similar to each other, leading to the degradation in performance when using both maps.

Our method is still unable to perform well in complex indoor scenes, as 4 of the 5 worst performing target labels are indoor objects. A large fraction of objects in indoor scenes are semantically related and co-occur with one another: 59.9% of images with a *dining table* have *chairs*, and 21.8% of images with a *chair* have a *sofa*. This poses a problem to disambiguate them from the noisy saliency maps. The segmentation of multiple indoor objects is thus a promising avenue for improvement in future work.

## 6. Conclusion

We presented a novel segmentation method for previously unseen labels. Through language and transfer learning, we can transfer visual priors from image classifiers to semantic segmentation models. The method to transfer information uses saliency maps from pretrained classifiers that indicate possible locations for the object to segment and as well as locations of possible distractors. We thoroughly study the impact of the semantics, saliency maps, and which class labels are most challenging for our proposed method.

## References

- [1] J. Ahn and S. Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018. 2, 3
- [2] H. Chen, Y. Wang, G. Wang, and Y. Qiao. LSTD: A low-shot transfer detector for object detection. In AAAI, editor, *Proceedings of AAAI Conference on Artificial Intelligence (AAAI)*, 2018. 3
- [3] L. Chen, S. Srivastava, Z. Duan, and C. Xu. Deep cross-modal audio-visual generation. In *Proceedings of the Thematic Workshops of ACM Multimedia 2017*, pages 349–357. ACM, 2017. 3, 4
- [4] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2018. 1, 2
- [5] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 1
- [6] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 3213–3223, 2016. 1, 2
- [7] J. Dai, K. He, and J. Sun. Boxesup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1635–1643, 2015. 1, 2
- [8] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 248–255. Ieee, 2009. 2
- [9] N. Dong and E. P. Xing. Few-shot semantic segmentation with prototype learning. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2018. 2, 3
- [10] A. Ess, T. Müller, H. Grabner, and L. J. Van Gool. Segmentation-based urban traffic scene understanding. In *Proceedings of the British Machine Vision Conference (BMVC)*, volume 1, page 2. Citeseer, 2009. 2
- [11] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010. 1, 2, 5
- [12] S. Gupta, P. Arbeláez, R. Girshick, and J. Malik. Indoor scene understanding with rgb-d images: Bottom-up segmentation, object detection and semantic segmentation. *International Journal of Computer Vision*, 112(2):133–149, 2015. 2
- [13] B. Hariharan, P. Arbelaez, L. Bourdev, S. Maji, and J. Malik. Semantic contours from inverse detectors. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2011. 5
- [14] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5
- [15] S. Hong, H. Noh, and B. Han. Decoupled deep neural network for semi-supervised semantic segmentation. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1495–1503, 2015. 1, 2, 3
- [16] R. Hu, P. Dollár, K. He, T. Darrell, and R. B. Girshick. Learning to segment every thing. In *Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018. 2, 3
- [17] A. Khoreva, R. Benenson, J. Hosang, M. Hein, and B. Schiele. Simple does it: Weakly supervised instance and semantic segmentation. In *Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 1665–1674. IEEE, 2017. 1, 3
- [18] S. Kwak, S. Hong, and B. Han. Weakly supervised semantic segmentation using superpixel pooling network. In *Proceedings of AAAI Conference on Artificial Intelligence (AAAI)*, 2017. 2, 3
- [19] L.-J. Li, R. Socher, and L. Fei-Fei. Towards total scene understanding: Classification, annotation and segmentation in an automatic framework. In *Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 2036–2043. IEEE, 2009. 2
- [20] G. Lin, A. Milan, C. Shen, and I. Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 5168–5177. IEEE, 2017. 2
- [21] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *Proceedings of the European Conference on Computer Vision*, pages 740–755. Springer, 2014. 2
- [22] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440, 2015. 1, 2
- [23] L. Long, W. Wang, J. Wen, M. Zhang, Q. Lin, and B. C. Ooi. Object-level representation learning for few-shot image classification. *arXiv preprint arXiv:1805.10777*, 2018. 3
- [24] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013. 3, 5
- [25] G. A. Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995. 2, 3, 5
- [26] V. Nekrasov, T. Dharmasiri, A. Spek, T. Drummond, C. Shen, and I. Reid. Real-time joint semantic segmentation and depth estimation using asymmetric annotations. *arXiv preprint arXiv:1809.04766*, 2018. 2
- [27] P. Ondruska, J. Dequaire, D. Z. Wang, and I. Posner. End-to-end tracking and semantic segmentation using recurrent neural networks. In *RSS, Workshop on Limits and Potentials of Deep Learning in Robotics*, 2016. 2
- [28] K. E. Papoutsakis and A. A. Argyros. Integrating tracking with fine object segmentation. *Image and Vision Computing*, 31(10):771–785, 2013. 2
- [29] J. Pennington, R. Socher, and C. Manning. Glove: Global vectors for word representation. In *Proceedings of the conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014. 5

- [30] K. Rakelly, E. Shelhamer, T. Darrell, A. Efros, and S. Levine. Conditional networks for few-shot semantic segmentation. In *International conference for learning representations (ICLR)*, 2018. [1](#), [2](#), [3](#), [5](#), [6](#), [7](#)
- [31] C. Redondo-Cabrera and R. J. López-Sastre. Learning to exploit the prior network knowledge for weakly-supervised semantic segmentation. *arXiv preprint arXiv:1804.04882*, 2018. [1](#), [3](#)
- [32] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. In *ACM transactions on graphics (TOG)*, volume 23, pages 309–314. ACM, 2004. [3](#), [4](#)
- [33] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 618–626. IEEE, 2017. [1](#), [4](#)
- [34] A. Shaban, S. Bansal, Z. Liu, I. Essa, and B. Boots. One-shot learning for semantic segmentation. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2017. [1](#), [2](#), [3](#), [5](#), [6](#)
- [35] M. Siam, M. Gamal, M. Abdel-Razek, S. Yogamani, and M. Jagersand. Rtseg: Real-time semantic segmentation comparative study. In *International Conference on Image Processing*, 2018. [2](#)
- [36] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. [5](#)
- [37] D. Smilkov, N. Thorat, B. Kim, F. Viégas, and M. Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017. [1](#), [4](#)
- [38] R. Socher, C. C. Lin, C. Manning, and A. Y. Ng. Parsing natural scenes and natural language with recursive neural networks. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 129–136, 2011. [2](#)
- [39] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014. [1](#), [4](#), [5](#)
- [40] F. S. Y. Yang, L. Zhang, T. Xiang, P. H. Torr, and T. M. Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR)*, 2018. [3](#)
- [41] A. R. Zamir, A. Sax, W. Shen, L. Guibas, J. Malik, and S. Savarese. Taskonomy: Disentangling task transfer learning. In *Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 3712–3722, 2018. [1](#)
- [42] T. Zhang, G. Lin, J. Cai, T. Shen, C. Shen, and A. C. Kot. Decoupled spatial neural attention for weakly supervised semantic segmentation. *CoRR*, abs/1803.02563, 2018. [1](#), [2](#), [3](#)
- [43] X. Zhang, Y. Wei, Y. Yang, and T. Huang. Sg-one: Similarity guidance network for one-shot semantic segmentation. *arXiv preprint arXiv:1810.09091*, 2018. [1](#), [2](#), [3](#), [6](#), [7](#)
- [44] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia. ICnet for real-time semantic segmentation on high-resolution images. In *Proceedings of the European Conference on Computer Vision*, 2018. [2](#)
- [45] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba. Semantic understanding of scenes through the ADE20K dataset. *arXiv preprint arXiv:1608.05442*, 2016. [1](#)