Learning Transferrable Knowledge for Semantic Segmentation with Deep Convolutional Neural Network

Seunghoon Hong[†,‡] Junhyuk Oh[‡] Honglak Lee[‡] Bohyung Han[†]
[†]Dept. of Computer Science and Engineering
POSTECH, Pohang, Korea
{maga33,bhhan}@postech.ac.kr
[‡]Dept. of Electrical Engineering and Computer Science
University of Michigan, Ann Arbor, MI, USA
{junhyuk,honglak}@umich.edu

Abstract

We propose a novel weakly-supervised semantic segmentation algorithm based on Deep Convolutional Neural Network (DCNN). Contrary to existing weakly-supervised approaches, our algorithm exploits auxiliary segmentation annotations available for different categories to guide segmentations on images with only image-level class labels. To make segmentation knowledge transferrable across categories, we design a decoupled encoder-decoder architecture with attention model. In this architecture, the model generates spatial highlights of each category presented in images using an attention model, and subsequently performs binary segmentation for each highlighted region using decoder. Combining attention model, the decoder trained with segmentation annotations in different categories boosts accuracy of weakly-supervised semantic segmentation. The proposed algorithm demonstrates substantially improved performance compared to the state-of-the-art weakly-supervised techniques in PASCAL VOC 2012 dataset when our model is trained with the annotations in 60 exclusive categories in Microsoft COCO dataset.

1. Introduction

Semantic segmentation refers to the task assigning dense class labels to pixels in an image. Although pixel-wise labels provide richer descriptions of images than bounding box labels or image-level tags, inferring such labels is a much more challenging task as it involves a highly complicated structured prediction problem.

Recent breakthrough in semantic segmentation has been mainly accelerated by the approaches based on Convolutional Neural Networks (CNNs) [4, 21, 11, 10, 25]. Given a classification network pre-trained on a large image collection, they learn a network for segmentation based on strong supervision—pixel-wise class labels. Although the approaches substantially improve the performance over the prior arts, training CNN requires a large number of fine-quality segmentation annotations, which are difficult to collect due to extensive labeling cost. For this reason, scaling up the semantic segmentation task to a large number of categories is very challenging in practice.

Weakly-supervised learning [5, 27, 29, 31] is an alternative approach to alleviate annotation efforts. They infer segmentation labels from training images given weak labels such as bounding boxes [5] or image-level class labels [31, 27, 29]. Since such annotations are easy to collect and even already available in existing datasets [6], it is straightforward to apply those approaches to large-scale problems with many categories. However, the segmentation quality by the weakly-supervised techniques is typically much worse than the one by supervised methods since there is no direct supervision for segmentation such as object shapes and locations during training.

The objective of this paper is to reduce the gap between semantic segmentation algorithms based on strong supervisions (*e.g.*, semi- and fully-supervised approaches) and weak supervisions (*e.g.*, weakly-supervised approaches). Our key idea is to employ segmentation annotations available for different categories to compensate for missing supervisions in weakly annotated images. No additional cost is required to collect such data since there are already several datasets publicly available with pixel-wise annotations, *e.g.*, BSD [22], Microsoft COCO [20], and LabelMe [32]. These datasets have not been actively explored yet for semantic segmentation due to the mismatches in semantic categories with the popular benchmark datasets, *e.g.*, PASCAL VOC [7]. The critical challenge in this problem is to learn common prior knowledge for segmentation transferrable across categories. It is not a trivial task with existing architectures, since they simply pose the semantic segmentation as pixel-wise classification and it is difficult to exploit examples from the unseen classes.

We propose a novel encoder-decoder architecture with an attention model, which is conceptually appropriate to transfer segmentation knowledge from one category to an-

other. In this architecture, the attention model generates *category-specific* saliency on each location of an image, while the decoder performs foreground segmentation using the saliency map based on *category-independent* segmentation knowledge. Our model trained on one dataset is transferable to another by adapting the attention model to focus on unseen categories. Since the attention model is trainable with only image-level class labels, our algorithm is applicable to semantic segmentation on weakly-annotated images through transfer learning. The contributions of this paper are summarized below.

- We propose a new paradigm for weakly-supervised semantic segmentation, which exploits segmentation annotations from different categories to guide segmentations with weak annotations. To our knowledge, this is the first attempt to tackle the weakly-supervised semantic segmentation problem by transfer learning.
- We propose a novel encoder-decoder architecture with attention model, which is appropriate to transfer the segmentation knowledge across categories.
- The proposed algorithm achieves substantial performance improvement over existing weakly-supervised approaches by exploiting segmentation annotations in exclusive categories.

The rest of the paper is organized as follows. We briefly review related work and introduce our algorithm in Section 2 and 3, respectively. The detailed configuration of the proposed network is described in Section 4. Training and inference procedures are presented in Section 5. Section 6 illustrates experimental results on a benchmark dataset.

2. Related Work

Recent success in CNN has brought significant progress on semantic segmentation in the past few years [4, 11, 10, 21, 25]. By posing the semantic segmentation as region-based classification problem, they train the network to produce pixel-wise class labels using segmentation annotations as training data [10, 11, 21, 25]. Based on this framework, some approaches improve segmentation performance by learning deconvolution network to capture accurate object boundaries [26] or adopting fully connected CRF as post-processing [4, 38]. However, the performance of the supervised approaches depends heavily on the size and quality of training data, which limits the scalability of the algorithms.

To reduce the efforts for annotations, weakly-supervised approaches attempt to learn the model for semantic segmentation only with weak annotations [5, 27, 29, 31]. To infer latent segmentation labels, they often rely on the techniques such as Multiple Instance Learning (MIL) [29, 31] or Expectation-Maximization (EM) [27]. Unfortunately, they are not sufficient to make up missing supervision and lead to significant performance degradation compared to fully-supervised approaches. In the middle, semi-supervised approaches [13, 27] exploit a limited number of strong annotations to reduce performance gap between fully- and weakly-supervised approaches. Notably, [13] proposed a decoupled encoder-decoder architecture for segmentation, where it divides semantic segmentation into two separate problems—classification and segmentation—and learns a decoder to perform binary segmentation for each class identified in the encoder. Although this semi-supervised approach improves performance by sharing the decoder for all classes, it still needs strong annotations in the classes of interest for segmentation. We remove this requirement by using segmentation annotations available for other categories.

In computer vision, the idea of employing external data to improve performance of target task has been explored in context of domain adaptation [33, 15, 9, 8] or transfer learning [19, 36]. However, the approaches in domain adaptation often assume that there are shared categories across domains, and the techniques with transfer learning are often limited to simple classification tasks. We refer [30] for comprehensive surveys on domain adaptation and transfer learning. Hoffman *et al*. [12] proposed a large-scale detection system by transferring knowledge for object detection between categories. Our work shares the motivations with this work, but aims to solve a highly complicated structured prediction problem, semantic segmentation.

There has been a long line of research on learning visual attention [1, 2, 3, 18, 24, 37, 35]. Their objective is to learn the attention mechanism that can adaptively focus on salient part of an image or video for various computer vision tasks, such as object recognition [1, 2, 18], object tracking [3], caption generation [37], image generation [35], etc. Our work is an extension of this idea to semantic segmentation by transfer learning.

3. Algorithm Overview

This paper tackles the weakly-supervised semantic segmentation problem in transfer learning perspective. Suppose that we have two sets of data, $\mathcal{T} = \{1, ..., N_t\}$ and $\mathcal{S} = \{1, ..., N_s\}$, which are composed of N_t and N_s images, respectively. Note that a set of images in *target* domain, denoted by $\mathcal{T}$, only have image-level class labels while the other set of data $\mathcal{S}$, referred to as *source* domain, have pixel-wise segmentation annotations. Our objective is to improve the weakly-supervised semantic segmentation on the target domain using the segmentation annotations available in the source domain. We assume that both target and source domains are composed of exclusive sets of categories. In this setting, there is no direct supervision (*i.e*., ground-truth segmentation labels) for the categories in the target domain, which makes our objective similar to a

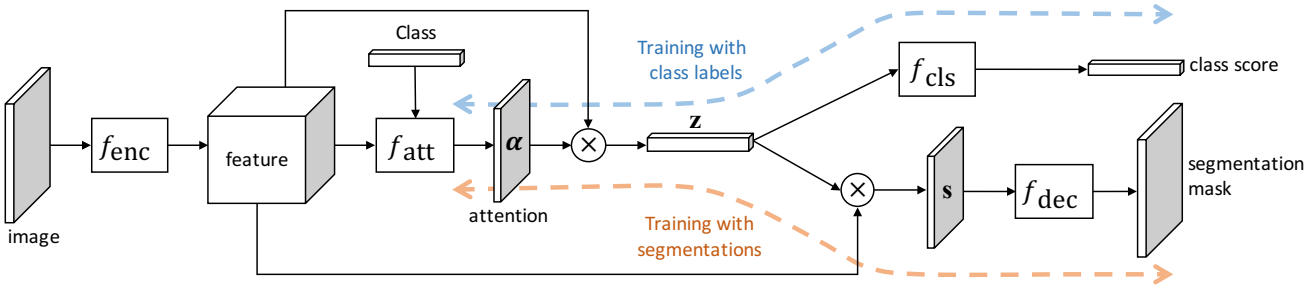


Figure 1. Overall architecture of the proposed algorithm. Given a feature extracted from the encoder, the attention model estimates adaptive spatial saliency of each category associated with input image (Section 4.2). The outputs of attention model are subsequently fed into the decoder, which generates foreground segmentation mask of each focused region (Section 4.3). During training, we fix the encoder by pre-trained weights, and leverage the segmentation annotations from source domain to train both the decoder and the attention model, and image-level class labels in both domains to train the attention model under classification objective. After training, semantic segmentation on the target domain is performed naturally by exploiting the decoder trained with source images and the attention model adapted to target domain (Section 5).

weakly-supervised semantic segmentation setting.

To transfer segmentation knowledge from source to target domain, we propose a novel encoder-decoder architecture with attention model. Figure 1 illustrates the overall architecture of the proposed algorithm. The network is composed of four parts: encoder, attention model, classifier and decoder. In this architecture, the input image is first transformed to a multi-dimensional feature vector by the encoder, and the attention model identifies salient region for each category associated with the image. The output of the attention model reveals location information of each category in a coarse feature map, where the dense and detailed foreground segmentation mask for each category is obtained by the decoder.

Training our network involves different mechanisms for source and target domain examples, since they are associated with heterogeneous annotations with different levels of supervision. We leverage the segmentation annotations from source domain to train both the decoder and the attention model with segmentation objective, while image-level class labels in both target *and* source domains are used to train the attention model under classification objective. The training is performed jointly for both objectives using examples from both domains.

The proposed architecture exhibits several advantages to capture transferrable segmentation knowledge across domains. Employing the decoupled encoder-decoder architecture [13] makes it possible to share the information for shape generation among different categories. The attention model provides not only predictions for localization but also category-specific information that enables us to adapt the decoder trained in source domain to target domain. The combination of two components makes information for segmentation transferable across different categories, and provides useful segmentation prior that is missing in weakly annotated images in target domain.

4. Architecture

This section describes our framework for semantic segmentation through transfer learning.

4.1. Preliminaries

We first describe notations and general configurations of the proposed model. Our network is composed of four parts, f_{enc}, f_{att}, f_{cls} and f_{dec}, which are neural networks corresponding to encoder, attention model, classifier and decoder, respectively. Our goal is to train all components using the examples from both domains except f_{enc}, which exploits a pre-trained network without fine-tuning.

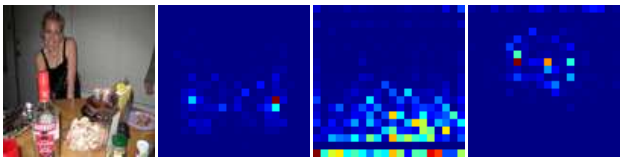
Let $\mathbf{x}$ denote a training image from either source or target domain. We assume that the image is associated with a set of class labels $\mathcal{L}^*$, which is given by either ground-truth (in training) or prediction (in testing). Given an input image $\mathbf{x}$, the network first extracts a feature descriptor as

$$\mathbf{A} = f_{\text{enc}}(\mathbf{x}; \theta_e), \quad \mathbf{A} \in \mathrm{R}^{M \times D} \tag{1}$$

where θ_e is the model parameter for the encoder, and M and D denote the number of hidden units in each channel and the number of channels, respectively. We employ VGG-16 layer net [34] pre-trained on ImageNet [6] as our encoder f_{enc}, and the feature descriptor $\mathbf{A}$ is obtained from the last convolutional layer to retain spatial information in the input image. The extracted feature and associated labels are then used to generate attentions and segment objects, which are discussed in the following subsections.

4.2. Attention model

Given a feature descriptor extracted from the encoder $\mathbf{A} \in \mathrm{R}^{M \times D}$ and its associated class labels $\mathcal{L}^*$, the objective of our attention model is to learn a set of positive weight vectors $\{\boldsymbol{\alpha}^l\}_{\forall l \in \mathcal{L}^*}$ defined over a 2D space, where each element of $\boldsymbol{\alpha}^l \in \mathrm{R}^M$ represents the relevance of each location



(a) Input image (b) $\boldsymbol{\alpha}^{bottle}$ (c) $\boldsymbol{\alpha}^{table}$ (d) $\boldsymbol{\alpha}^{person}$

Figure 2. Examples of learned attentions. (a) Input image, (b), (c) and (d) represent attention weights obtained by Eq. (3). The proposed attention model adaptively focuses on different areas in an image depending on input labels.

to the l^{th} category. Our attention model is formally given by

$$\mathbf{v}^l = f_{\text{att}}(\mathbf{A}, \mathbf{y}^l; \theta_\alpha), \quad \mathbf{v}^l \in \mathrm{R}^M \tag{2}$$

$$\alpha_i^l = \frac{\exp\left(v_i^l\right)}{\sum_i \exp\left(v_i^l\right)}, \quad \boldsymbol{\alpha}^l \in \mathrm{R}^M, \tag{3}$$

where $\mathbf{y}^l$ is a one-hot label vector for the l^{th} category, θ_α denotes parameters of the attention model, and $\mathbf{v}^l$ represents unnormalized attention weights. To encourage the model to pay attention to only a part of the image, we normalize $\mathbf{v}^l$ to $\boldsymbol{\alpha}^l$ using a softmax function as suggested in [37].

To obtain category-specific attention $\boldsymbol{\alpha}^l$ using our attention model f_{att}, we employ multiplicative interactions [23] between feature and label vector. It learns a set of gating parameters represented by a 3-way tensor to model correlation between feature and label vectors. For scalability issue, we reduce the number of parameters by the factorization technique proposed in [23], and our model can be written as

$$\mathbf{v}^l = \mathbf{W}^{att}\left(\mathbf{W}^{feat}\mathbf{A} \odot \mathbf{W}^{label}\mathbf{y}^l\right) + \mathbf{b}, \tag{4}$$

where $\odot$ denotes element-wise multiplication and $\mathbf{b} \in \mathrm{R}^M$ is a bias. Note that the weights are given by $\mathbf{W}^{feat} \in \mathrm{R}^{d \times MD}$, $\mathbf{W}^{label} \in \mathrm{R}^{d \times L}$ and $\mathbf{W}^{att} \in \mathrm{R}^{M \times d}$, where L and d denote the size of label vector and the number of factors, respectively. We observe that using multiplicative interaction generally gives better results than additive ones (*e.g.*, concatenation), because it is capable of capturing high-order dependency between feature and label.

To apply the attention to our transfer-learning scenario, the model f_{att} should be trainable in both target and source domains. Since examples in each domain are associated with different types of annotations, we train attention model based on different objectives on two separate branches. In the following, we first describe the learning objective for attention model with weak annotations, whereas the one with strong annotations is described in the next subsection.

To train the attention model with only image-level class labels, we create f_{cls} composed of two fully-connected layers on top of the attention model, and optimize both f_{att} and f_{cls} under a classification objective. To this end, we extract features based on the category-specific attention by aggregating features over the spatial region as follows:

$$\mathbf{z}^l = \mathbf{A}^{\mathrm{T}}\boldsymbol{\alpha}^l, \quad \mathbf{z}^l \in \mathrm{R}^D. \tag{5}$$

Intuitively, $\mathbf{z}^l$ represents a category-specific feature defined over all the channels in the feature map.

Using the images with weak annotations in both target and source domain, we jointly train attention model and classifier to minimize the classification loss as follows:

$$\min_{\theta_\alpha, \theta_c} \sum_{i \in \mathcal{T} \cup \mathcal{S}} \sum_{l \in \mathcal{L}_i^*} e_c\left(\mathbf{y}_i^l, f_{\text{cls}}(\mathbf{z}_i^l; \theta_c)\right), \tag{6}$$

where θ_c denotes parameters associated with classifier, and e_c denotes the loss between ground-truth $\mathbf{y}_i^l$ and predicted label vector $f_{\text{cls}}(\mathbf{z}_i^l; \theta_c)$. We employ a cross-entropy loss to measure classification error e_c.

The optimization of Eq. (6) is susceptible for overfitting since the ground-truth class label $\mathbf{y}^l$ is given as an input to attention model as well. In practice, we observe that our model avoids this issue by effectively eliminating the direct link from attention to label prediction and constructing intermediate representation $\mathbf{z}$ using the original feature $\mathbf{A}$.

Figure 2 illustrates the learned attention weights for each class. We observe that the attention model captures spatial saliency effectively given its input labels.

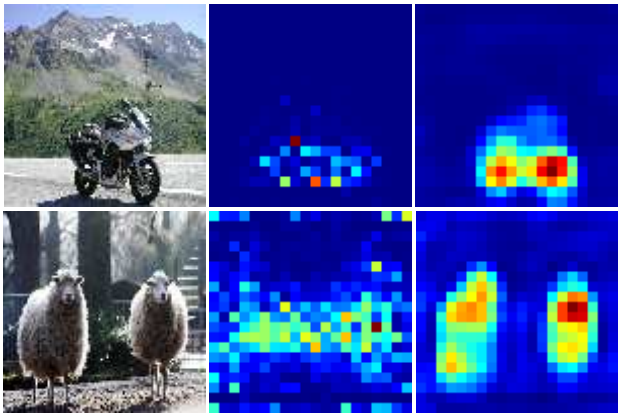
4.3. Decoder

The attention model described in the previous section generates a set of adaptive saliency maps for each category $\{\boldsymbol{\alpha}^l\}_{\forall l \in \mathcal{L}^*}$, which provides useful information for localization. Given these attentions, the next step of our algorithm is to reconstruct dense foreground segmentation mask for each attended category by the decoder. However, the direct application of attention weights to segmentation may be problematic, since the activations tend to be sparse due to the softmax operation in Eq. (3) and may lose information encoded in the feature map useful for shape generation.

To resolve this issue and reconstruct useful information for segmentation, we feed the additional inputs to the decoder using attention $\boldsymbol{\alpha}^l$ and the original feature $\mathbf{A}$. Rather than directly using the attention, we exploit the intermediate representation $\mathbf{z}^l$ obtained from Eq. (5). It represents relevance of each *channel* out of the feature maps with respect to the l^{th} category. Then we aggregate spatial activations in each channel of the feature using $\mathbf{z}^l$ as coefficients, which is given by

$$\mathbf{s}^l = \mathbf{A}\mathbf{z}^l, \quad \mathbf{s} \in \mathrm{R}^M \tag{7}$$

where $\mathbf{s}^l$ represents *densified attention* in the same size with $\boldsymbol{\alpha}^l$ and serves as inputs to the decoder. As shown in Figure 3, densified attention maps preserve more details of the object shape compared to the original attention ($\boldsymbol{\alpha}^l$).



(a) Input image (b) Attention (c) Densified attention

Figure 3. Examples of attention ($\boldsymbol{\alpha}^l$) and densified attention ($\mathbf{s}^l$).

Given densified attention $\mathbf{s}^l$ as input, the attention model and decoder are jointly trained to minimize the segmentation loss by the following objective function

$$\min_{\theta_\alpha,\theta_s} \sum_{i\in\mathcal{S}} \sum_{l\in\mathcal{L}_i^*} e_s\left(\mathbf{d}_i^l, f_{\text{dec}}(\mathbf{s}_i^l;\theta_s)\right), \tag{8}$$

where $\mathbf{d}_i^l$ denotes a *binary* segmentation mask of the i^{th} image for the l^{th} category, and e_s denotes pixel-wise loss function between ground-truth and predicted segmentation masks. Similar to classification, we employ a cross-entropy loss function for e_s. Since training requires ground-truth segmentation annotations, the objective function is only optimized with images in source domain. Note that the above equation involves optimization of attention model. During training, the attention model is learned using data with two different types of annotations under both Eq. (6) and (8).

We employ recently proposed deconvolution network [26] for our decoder architecture f_{dec}. Given an input to the decoder $\mathbf{s}^l$, it generates a segmentation mask in the same size as the input image by multiple successive operations of unpooling, deconvolution and rectification. Pooling switches are shared between pooling and unpooling layers, which is appropriate to recover accurate object boundary. We refer to [26] for more details about this network.

We train the decoder for segmentation given attention $\boldsymbol{\alpha}^l$. By decoupling classification, which is a domain specific task, from decoding [13], we capture category-independent information for shape generation and apply the architecture to any unseen categories. Since all weights in the decoder are shared between different categories, it potentially encourages the decoder to capture common shape information that can be generally applicable to multiple categories.

5. Training and Inference

This section describes the training and inference procedure of the proposed algorithm. Combining Eq. (6) and (8), the overall objective function is given by

$$\begin{aligned}\min_{\theta_\alpha,\theta_c,\theta_s} &\sum_{i\in\mathcal{T}\cup\mathcal{S}} \sum_{l\in\mathcal{L}_i^*} e_c\left(\mathbf{y}_i^l, f_{\text{cls}}(\mathbf{z}_i^l;\theta_c)\right) \\ &+ \lambda \sum_{j\in\mathcal{S}} \sum_{l\in\mathcal{L}_j^*} e_s\left(\mathbf{d}_j^l, f_{\text{dec}}(\mathbf{s}_j^l;\theta_s)\right),\end{aligned} \tag{9}$$

where λ controls balance between classification and segmentation losses. Note that it allows joint optimization of attention model for both classification and segmentation. Although our attention model is generally good even trained with only class labels (see Figure 3), training attention based only on classification objective sometimes leads to noisy predictions due to missing supervision of localization. By jointly training with segmentation objective, we regularize to avoid finding noisy solution for target domain categories. After training, we remove the classification layers f_{cls} since it is required only in training to learn attentions for the data from target domain categories.

For inference of target domain images, we first apply a separate classifier to identify a set of labels $\tilde{\mathcal{L}}^*$ associated with the image. Then, for each identified label $l \in \tilde{\mathcal{L}}^*$, we iteratively construct attention weights $\boldsymbol{\alpha}_i^l$ and obtain foreground segmentation mask $f_{\text{dec}}(\mathbf{s}_i^l)$ from the decoder output. Given foreground probability maps from all labels $\{f_{\text{dec}}(\mathbf{s}_i^l)\}_{\forall l\in\tilde{\mathcal{L}}^*}$, the final segmentation label is obtained by taking the maximum probability across channels.

6. Experiments

This section describes detailed information in implementation and discusses experimental results.

6.1. Implementation Details

Datasets We employ PASCAL VOC 2012 [7] as target domain and Microsoft COCO (MS-COCO) [20] as source domain, which have 20 and 80 labeled semantic categories, respectively. To simulate the transfer learning scenario, we remove all training images relevant to 20 PASCAL VOC categories from MS-COCO dataset, and use only 17,443 images from 60 categories (excluding the ones in the PASCAL VOC dataset) to construct the source domain data. We train our model using image-level class labels in both datasets and segmentation annotations in MS-COCO dataset, and evaluate the performance on PASCAL VOC 2012 benchmark dataset.

Training We initialize the encoder by fine-tuning the pre-trained CNN from ImageNet [6] to perform multi-class classification on the combined datasets of PASCAL VOC and MS-COCO. The weights in the attention model and classification layers (θ_α and θ_c, respectively) are pre-trained by optimizing Eq. (6). Then we optimize both decoder, attention model and classification layers jointly using the objective function in Eq. (9) with $\lambda = 2$, while the weights in

Table 1. Evaluation results on PASCAL VOC 2012 *validation* set.

Method	bkg	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbk	person	plant	sheep	sofa	train	tv	mean
Weakly-supervised:																						
EM-Adapt [27]	67.2	29.2	17.6	28.6	22.2	29.6	47.0	44.0	44.2	14.6	35.1	24.9	41.0	34.8	41.6	32.1	24.8	37.4	24.0	38.1	31.6	33.8
CCNN [28]	68.5	25.5	18.0	25.4	20.2	36.3	46.8	47.1	48.0	15.8	37.9	21.0	44.5	34.5	46.2	40.7	30.4	36.3	22.2	38.8	36.9	35.3
MIL+seg [31]	79.6	50.2	21.6	40.9	34.9	40.5	45.9	51.5	60.6	12.6	51.2	11.6	56.8	52.9	44.8	42.7	31.2	55.4	21.5	38.8	36.9	42.0
Semi-supervised:																						
DecoupledNet [13]	86.5	69.9	33.6	58.5	42.4	50.4	68.8	63.2	67.5	11.5	61.8	20.0	61.2	66.7	60.1	50.8	30.2	67.9	33.9	59.2	51.0	53.1
EM-Adapt [27]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	47.6
Transfer:																						
TransferNet	85.3	68.5	26.4	69.8	36.7	49.1	68.4	55.8	77.3	6.2	75.2	14.3	69.8	71.5	61.1	31.9	25.5	74.6	33.8	49.6	43.7	52.1
TransferNet-GT	85.2	70.6	25.3	61.7	42.2	38.9	67.5	53.9	73.3	20.6	81.5	26.9	69.6	73.2	66.6	36.7	26.9	82.9	42.2	54.4	39.3	54.3
DecoupledNet†	79.2	13.1	7.7	38.4	14.3	15.0	14.7	46.0	60.5	3.7	28.0	1.7	54.0	37.5	24.0	9.2	4.5	46.2	3.4	18.7	13.0	25.4
BaselineNet	79.1	49.4	15.8	41.5	33.1	38.6	48.4	44.8	57.6	13.1	63.5	3.7	48.4	56.1	50.7	41.4	20.3	61.4	25.4	35.1	24.4	40.6

Table 2. Evaluation results on PASCAL VOC 2012 *test* set.

Method	bkg	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbk	person	plant	sheep	sofa	train	tv	mean
Fully-supervised:																						
FCN-8s [21]	91.2	76.8	34.2	68.9	49.4	60.3	75.3	74.7	77.6	21.4	62.5	46.8	71.8	63.9	76.5	73.9	45.2	72.4	37.4	70.9	55.1	62.2
CRF-RNN [38]	93.1	90.4	55.3	88.7	68.4	69.8	88.3	82.4	85.1	32.6	78.5	64.4	79.6	81.9	86.4	81.8	58.6	82.4	53.5	77.4	70.1	74.7
DeepLab-CRF [4]	93.1	84.4	54.5	81.5	63.6	65.9	85.1	79.1	83.4	30.7	74.1	59.8	79.0	76.1	83.2	80.8	59.7	82.2	50.4	73.1	63.7	71.6
DeconvNet [26]	93.1	89.9	39.3	79.7	63.9	68.2	87.4	81.2	86.1	28.5	77.0	62.0	79.0	80.3	83.6	80.2	58.8	83.4	54.3	80.7	65.0	72.5
Weakly-supervised:																						
EM-Adapt [27]	76.3	37.1	21.9	41.6	26.1	38.5	50.8	44.9	48.9	16.7	40.8	29.4	47.1	45.8	54.8	28.2	30.0	44.0	29.2	34.3	46.0	39.6
CCNN [28]	70.1	24.2	19.9	26.3	18.6	38.1	51.7	42.9	48.2	15.6	37.2	18.3	43.0	38.2	52.2	40.0	33.8	36.0	21.6	33.4	38.3	35.6
MIL+seg [31]	78.7	48.0	21.2	31.1	28.4	35.1	51.4	55.5	52.8	7.8	56.2	19.9	53.8	50.3	40.0	38.6	27.8	51.8	24.7	33.3	46.3	40.6
Transfer:																						
TransferNet	85.7	70.1	27.8	73.7	37.3	44.8	71.4	53.8	73.0	6.7	62.9	12.4	68.4	73.7	65.9	27.9	23.5	72.3	38.9	45.9	39.2	51.2

the decoder (θ_s) are initialized with zero-mean Gaussians. We fix the weights in the encoder (θ_e) during training.

Optimization We implement the proposed algorithm based on Caffe [14] library. We employ Adam optimization [16] to train our network with learning rate 0.0005 and default hyper-parameter values proposed in [16]. The size of mini-batch is set to 64. Training our model takes 4 hours for pre-training attention model including classification layers, and 10 hours for joint training of all other parts, using NVIDIA Titan X GPU.

Inference We adopt VGG 16-layer net [34] as an additional classifier, which is pre-trained on ImageNet [6] and fine-tuned on PASCAL VOC dataset. The predicted class labels are used to generate class-specific attention and segmentation as described in Section 5. Optionally, we employ post processing using fully-connected CRF [17]. In this case, we apply the CRF on foreground/background probability maps for each class label independently, and obtain combined segmentations by taking pixel-wise maximums of foreground probabilities across labels.

6.2. Comparison to Other Methods

This section presents comparative evaluation results of our algorithm PASCAL VOC 2012 benchmark dataset. We follow *comp6* evaluation protocol, and scores are measured by computing Intersection over Union (IoU) between ground truth and predicted segmentation.

Table 1 summarizes the evaluation results on PASCAL VOC 2012 validation dataset. We compared the proposed algorithm with state-of-the-art weakly- and semi-supervised algorithms[1]. Our method is denoted by TransferNet, and TransferNet-GT indicates our method with ground-truth class labels for segmentation inference, which serves as the upper-bound performance of our method since it assumes classification is perfect. The proposed algorithm outperforms all weakly-supervised semantic segmentation techniques with substantial margins, although it does not employ any ground-truth segmentations for categories used in evaluation. The performance of the proposed algorithm is comparable to semi-supervised semantic segmentation methods, which exploits a small number of ground-truth segmentations in addition to weakly-annotated images for training. The results suggest that segmentation annotations from different categories can make up missing supervision in weakly-annotated images; the proposed encoder-decoder architecture based on attention model successfully captures transferable segmentation knowledge from the exclusive segmentation annotations and uses it as prior for segmentation in unseen categories.

Table 2 summarizes our results on PASCAL VOC 2012 test dataset. Our algorithm exhibits superior performance to weakly-supervised approaches, but there are still large per-

[1]Strictly speaking, our method is not directly comparable to both approaches since we use auxiliary examples. Note that we do not use ground-truth segmentation annotations for the categories used in evaluation, since the examples are from different categories.

formance gaps with fully-supervised approaches. It shows that there is domain-specific segmentation knowledge that cannot be made up by annotations form different categories.

The qualitative results of the proposed algorithm are presented in Figure 5. Our algorithm often produces accurate segmentations in the target domain by transferring the decoder trained with source domain examples, although it is not successful in capturing some category-specific fine details in some examples. The missing details can be recovered through post-processing based on CRF. Since the attention model in the target domain may not be perfect due to missing supervisions, our algorithm sometimes produces noisy predictions as illustrated in Figure 5(b).

6.3. Comparison to Baselines

To better understand the benefits from the attention model in our transfer learning scenario, we compare the proposed algorithm with two baseline algorithms, which are denoted by DecoupledNet† and BaselineNet.

DecoupledNet† has identical to the architecture proposed in [13], but has a direct connection between encoder and decoder without attention mechanism. Note that the decoder of DecoupledNet† is trained on MS-COCO dataset while segmentation is tested on PASCAL VOC dataset. The result in Table 1 shows that the model trained on source domain fails to adapt to target domain categories. It is mainly because the decoder cannot interpret the features from unseen categories in target domain. Our model mitigates this issue since the attention model provides coherent representations to decoder across domains.

Although the above baseline shows the benefits of the attention model in our architecture, the advantage of attention estimation from the intermediate layer is still not clear enough. To verity the benefit of attention, we employ another baseline similar to FCN [21] denoted by BaselineNet, which uses class score map as input to the decoder. It can be considered as a special case of our method that the attention is extracted from the final layer of the classification network ($f_{\text{att}} = f_{\text{cls}}$). The performance of BaselineNet is better than DecoupledNet† since the class score map provides category-invariant representations to the decoder. However, the performance is considerably worse than the proposed method as shown in Table 1 and Figure 5. We observe that the class score map is sparse and focused on discriminative regions, while densified attention map in our model contains richer information for segmentation.

The comparisons to the baseline algorithms show that transferring segmentation knowledge across categories is a very challenging task. The naïve extensions of existing architectures have troubles in generalizing the knowledge invariant to categories. In contrast, our model effectively transfers segmentation knowledge by learning general features through attention mechanism.

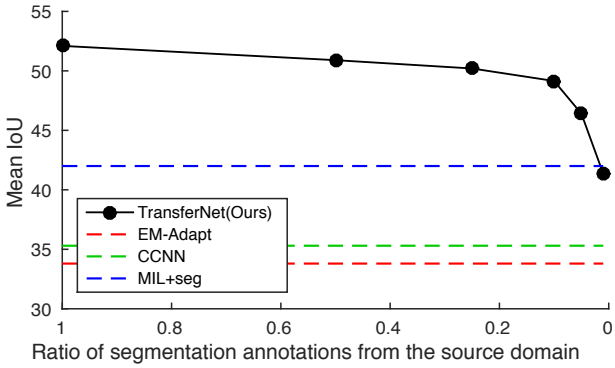


Figure 4. Performance of the proposed algorithm with varying number of annotations in the source domain.

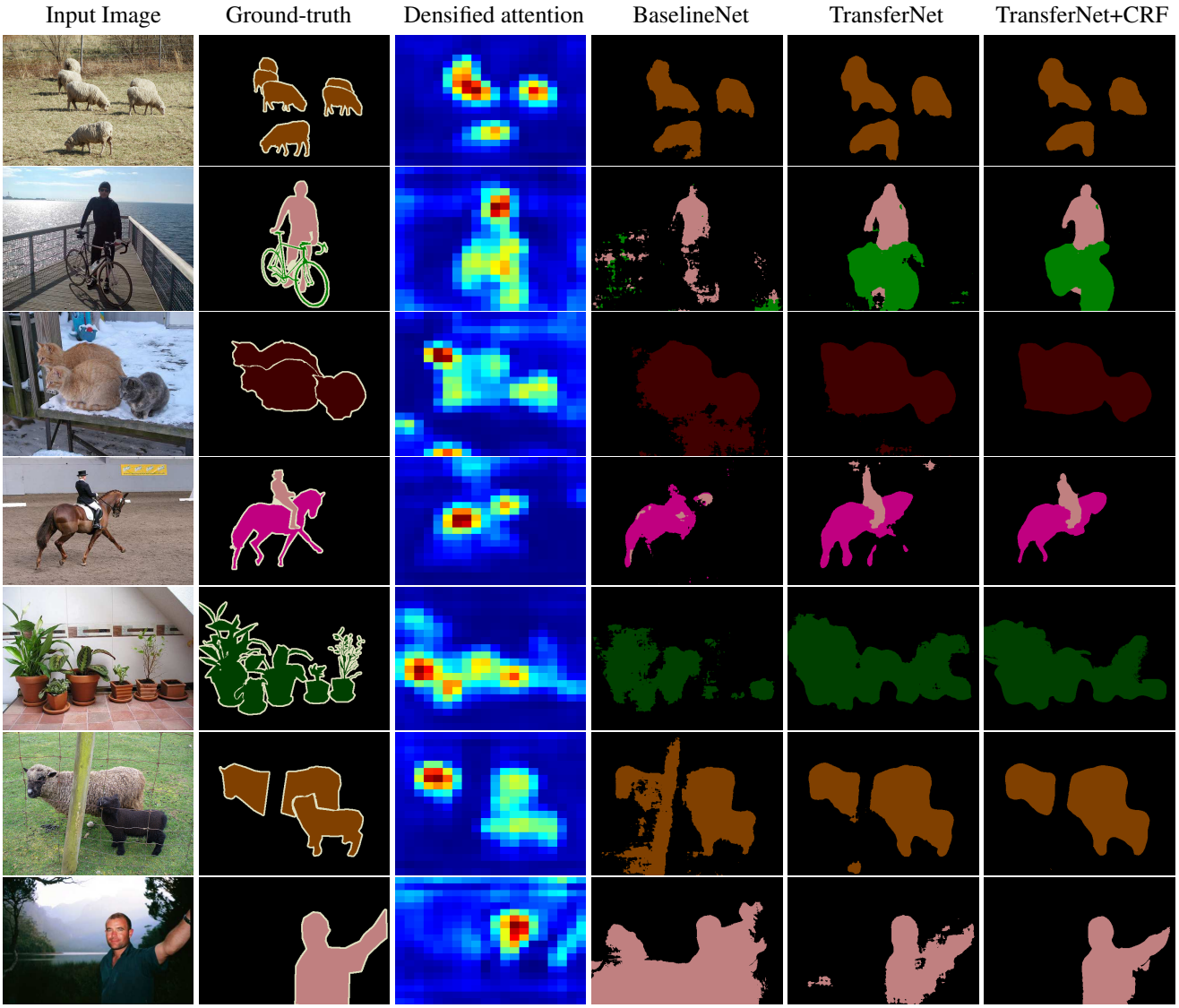
6.4. Impact of Annotation Size in Source Domain

To see the impact of number of annotations in the source domain, we conduct additional experiments by varying the number of annotations in the source domain (MS-COCO). We randomly construct subsets of training data by varying their sizes in ratios (50%, 25%, 10%, 5% and 1%) and average the performance in each size with 3 subsets. The results are illustrated in Figure 4. In general, more annotations in the source domain improve the segmentation quality on the target domain. The performance of the proposed algorithm is still better than other weakly-supervised methods even with a very small fraction of annotations. It suggests that exploiting even small number of segmentations from other categories can effectively reduce the gap between the approaches based on strong and weak supervisions.

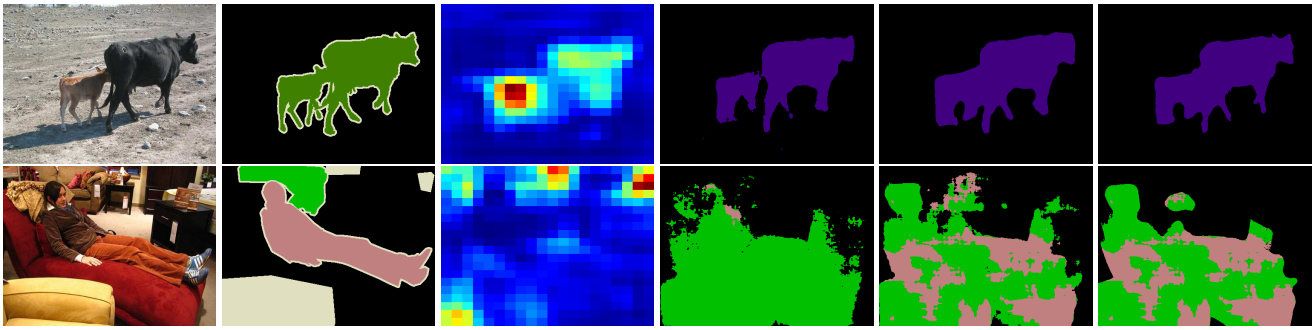
7. Conclusion

We propose a novel approach for weakly-supervised semantic segmentation, which exploits extra segmentation annotations in different categories to improve segmentation performance on the dataset with missing supervisions. The proposed encoder-decoder architecture with attention model is appropriate to capture transferable segmentation knowledge across categories. The results on a challenging benchmark dataset suggest that the gap originated from missing strong supervision can be reduced by transfer learning. We believe that scaling up the proposed algorithm to a large number of categories would be one interesting future research direction, *e.g.*, semantic segmentation on 7.6K categories in ImageNet dataset using segmentation annotations from 20 PASCAL VOC categories.

Acknowledgments This work was partly supported by IITP grant (B0101-16-0307; Machine Learning Center, B0101-16-0552; Deep View), NRF grant (NRF-2011-0031648, Global Frontier R&D Program on Human-Centered Interaction for Coexistence) funded by the Korean government (MSIP), NSF CAREER grant IIS-1453651, and ONR grant N00014-13-1-0762.



(a) Examples that our method produces accurate segmentation.



(b) Examples that our method produces inaccurate segmentation due to misclassification (top) or inaccurate attention (bottom).

Figure 5. Examples of semantic segmentation on PASCAL VOC 2012 validation images. The attentions (the 3rd column) are extracted from the model trained using Eq. (9), and aggregated over all categories for visualization. (a) Our methods based on attention model (TransferNet and TransferNet+CRF) produce accurate segmentation results even without CRF by transferring learned segmentation knowledge from source domain. Our results tend to be denser and more accurate than the results from BaselineNet, which generates segmentation from class score map. (b) Our algorithm sometimes produces inaccurate segmentations when the input labels are wrong due to misclassification (top) or attention output is noisy (bottom).

References

[1] B. Alexe, N. Heess, Y. W. Teh, and V. Ferrari. Searching for objects driven by context. In *NIPS*, 2012. 2

[2] J. Ba, V. Mnih, and K. Kavukcuoglu. Multiple object recognition with visual attention. In *ICLR*, 2015. 2

[3] L. Bazzani, N. de Freitas, H. Larochelle, V. Murino, and J.-A. Ting. Learning attentional policies for object tracking and recognition in video with deep networks. In *ICML*, 2011. 2

[4] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected CRFs. In *ICLR*, 2015. 1, 2, 6

[5] J. Dai, K. He, and J. Sun. BoxSup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *ICCV*, 2015. 1, 2

[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: a large-scale hierarchical image database. In *CVPR*, 2009. 1, 3, 5, 6

[7] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The Pascal Visual Object Classes (VOC) Challenge. *IJCV*, 88(2):303–338, 2010. 1, 5

[8] Y. Ganin and V. Lempitsky. Unsupervised domain adaptation by backpropagation. In *ICML*, 2015. 2

[9] R. Gopalan, R. Li, and R. Chellappa. Domain adaptation for object recognition: An unsupervised approach. In *ICCV*, 2011. 2

[10] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Simultaneous detection and segmentation. In *ECCV*, 2014. 1, 2

[11] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Hypercolumns for object segmentation and fine-grained localization. In *CVPR*, 2015. 1, 2

[12] J. Hoffman, S. Guadarrama, E. S. Tzeng, R. Hu, J. Donahue, R. Girshick, T. Darrell, and K. Saenko. LSDA: large scale detection through adaptation. In *NIPS*, 2014. 2

[13] S. Hong, H. Noh, and B. Han. Decoupled deep neural network for semi-supervised semantic segmentation. In *NIPS*, 2015. 2, 3, 5, 6, 7

[14] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *ACM MM*, 2014. 6

[15] A. Khosla, T. Zhou, T. Malisiewicz, A. Efros, and A. Torralba. Undoing the damage of dataset bias. In *ECCV*, 2012. 2

[16] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 6

[17] P. Krähenbühl and V. Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *NIPS*, 2011. 6

[18] H. Larochelle and G. E. Hinton. Learning to combine foveal glimpses with a third-order boltzmann machine. In *NIPS*, 2010. 2

[19] J. J. Lim, R. Salakhutdinov, and A. Torralba. Transfer learning by borrowing examples for multiclass object detection. In *NIPS*, 2011. 2

[20] 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 *ECCV*, 2014. 1, 5

[21] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 1, 2, 6, 7

[22] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *ICCV*, 2001. 1

[23] R. Memisevic. Learning to relate images. *TPAMI*, 35(8):1829–1846, 2013. 4

[24] V. Mnih, N. Heess, A. Graves, and K. Kavukcuoglu. Recurrent models of visual attention. In *NIPS*, 2014. 2

[25] M. Mostajabi, P. Yadollahpour, and G. Shakhnarovich. Feedforward semantic segmentation with zoom-out features. *CVPR*, 2015. 1, 2

[26] H. Noh, S. Hong, and B. Han. Learning deconvolution network for semantic segmentation. In *ICCV*, 2015. 2, 5, 6

[27] G. Papandreou, L.-C. Chen, K. Murphy, and A. L. Yuille. Weakly- and semi-supervised learning of a DCNN for semantic image segmentation. In *ICCV*, 2015. 1, 2, 6

[28] D. Pathak, P. Krähenbühl, and T. Darrell. Constrained convolutional neural networks for weakly supervised segmentation. In *ICCV*, 2015. 6

[29] D. Pathak, E. Shelhamer, J. Long, and T. Darrell. Fully convolutional multi-class multiple instance learning. In *ICLR*, 2015. 1, 2

[30] N. Patricia and B. Caputo. Learning to learn, from transfer learning to domain adaptation: A unifying perspective. In *CVPR*, 2014. 2

[31] P. O. Pinheiro and R. Collobert. From image-level to pixel-level labeling with convolutional networks. In *CVPR*, 2015. 1, 2, 6

[32] B. Russell, A. Torralba, K. Murphy, and W. T. Freeman. LabelMe: a database and web-based tool for image annotation. *IJCV*, 77(1-3):157–173, 2010. 1

[33] K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In *ECCV*, 2010. 2

[34] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 3, 6

[35] Y. Tang, N. Srivastava, and R. R. Salakhutdinov. Learning generative models with visual attention. In *NIPS*, 2014. 2

[36] T. Tommasi, F. Orabona, and B. Caputo. Learning categories from few examples with multi model knowledge transfer. *TPAMI*, 36(5):928–941, 2014. 2

[37] K. Xu, J. Ba, R. Kiros, K. Cho, A. C. Courville, R. Salakhutdinov, R. S. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, 2015. 2, 4

[38] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. Torr. Conditional random fields as recurrent neural networks. In *ICCV*, 2015. 2, 6