Weakly- and Semi-Supervised Learning of a DCNN for Semantic Image Segmentation
(2015)

George Papandreou, Liang-Chieh Chen, Kevin Murphy, Alan L. Yuille

Notes

1 Introduction

The authors in this paper study the problem of learning a DCNN for either (1) weakly annotated training datasuch as bounding boxes or image-level labels or (2) a com-bination of few strongly labeled and many weakly labeledimages, sourced from one or multiple datasets.

The authors present an EM algorithms for training with image-level or bounding box annotation, applicable to both weakly-supervised and semi-supervised settings, this method achieves excellent per-formance when combining a small number of pixel-level annotated images with a large number of image-level or bounding box annotated images, nearly matching the results achieved when all training images have pixel-level annotations.

2 Method

The model is based on deeplab, with a DCNN for predicting image segmentation folowed by a fully connected CRF for refinnement, the results is $y_m \in \{0, \ldots, L\}$ which a the label for a pixel at position $m \in \{1, \ldots, M\}$ in the image.

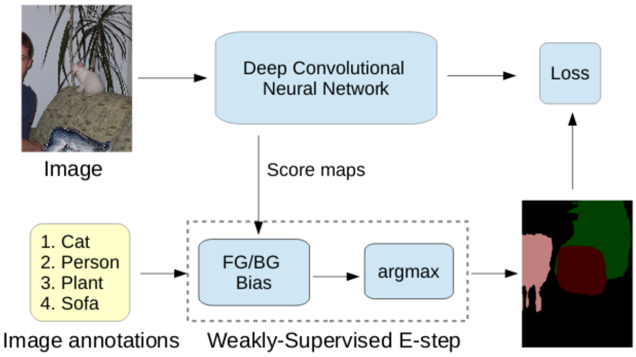


Figure 2. DeepLab model training using image-level labels.

Pixel-level annotations

In a fully supervised case, the training objective is a CE between the predicted labels nad the ground truth:

$$J(\boldsymbol{\theta}) = \log P(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\theta}) = \sum_{m=1}^{M} \log P\left(y_m|\boldsymbol{x};\boldsymbol{\theta}\right)$$

Image-level annotations

In this case, we can only observe the image x and the classes z but not pixel-wise labels, which are considered as latent variables y, we can then construct a graphical model:

$$P(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}; \boldsymbol{\theta}) = P(\boldsymbol{x}) \left(\prod_{m=1}^{M} P\left(y_m | \boldsymbol{x}; \boldsymbol{\theta}\right) \right) P(\boldsymbol{z}|\boldsymbol{y})$$

We can now train our model by applying E step and M step:

E-step : for the E step we want to maximize the segmentation mask predicted using the model and the input image $P\left(\boldsymbol{y}|\boldsymbol{x}; \boldsymbol{\theta}'\right)$, so that the image class predicted from the mask will be correct $P(\boldsymbol{z}|\boldsymbol{y})$:

$$\begin{aligned} \hat{\boldsymbol{y}} &= \underset{\boldsymbol{y}}{\operatorname{argmax}} P\left(\boldsymbol{y}|\boldsymbol{x}; \boldsymbol{\theta}'\right) P(\boldsymbol{z}|\boldsymbol{y}) \\ &= \underset{\boldsymbol{y}}{\operatorname{argmax}} \log P\left(\boldsymbol{y}|\boldsymbol{x}; \boldsymbol{\theta}'\right) + \log P(\boldsymbol{z}|\boldsymbol{y}) \\ &= \underset{\boldsymbol{y}}{\operatorname{argmax}} \left(\sum_{m=1}^{M} f_m\left(y_m|\boldsymbol{x}; \boldsymbol{\theta}'\right) + \log P(\boldsymbol{z}|\boldsymbol{y}) \right) \end{aligned}$$

But we also need to specify the observation model $P(\boldsymbol{z}|\boldsymbol{y})$, which is assumed that it is factorized over the pixel positions:

$$\log P(\boldsymbol{z}|\boldsymbol{y}) = \sum_{m=1}^{M} \phi\left(y_m, \boldsymbol{z}\right) + (\text{const})$$

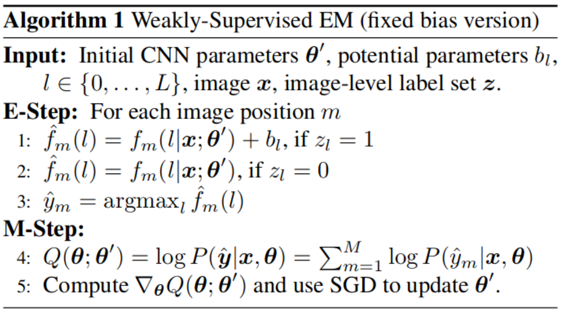
Given that:

$$\phi\left(y_m = l, \boldsymbol{z}\right) = \begin{cases} b_l & \text{if } z_l = 1 \\ 0 & \text{if } z_l = 0 \end{cases}$$

M-step : Now in the M-step, we want to optimize the model so that its output is close to the prediction we obtained in the E step:

$$\sum_{y} P\left(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{z}; \boldsymbol{\theta}'\right) \log P(\boldsymbol{y}|\boldsymbol{x}; \boldsymbol{\theta}) \approx \log P(\hat{\boldsymbol{y}}|\boldsymbol{x}; \boldsymbol{\theta})$$

And this is summerized as follows:



Algorithm 1 Weakly-Supervised EM (fixed bias version)

```
Input: Initial CNN parameters $\boldsymbol{\theta}'$, potential parameters $b_l$,
    $l \in \{0, \ldots, L\}$, image $\boldsymbol{x}$, image-level label set $\boldsymbol{z}$.
E-Step: For each image position $m$
  $\hat{f}_m(l) = f_m(l|\boldsymbol{x}; \boldsymbol{\theta}') + b_l$, if $z_l = 1$
  $\hat{f}_m(l) = f_m(l|\boldsymbol{x}; \boldsymbol{\theta}')$, if $z_l = 0$
  $\hat{y}_m = \operatorname{argmax}_l \hat{f}_m(l)$
M-Step:
  $Q(\boldsymbol{\theta}; \boldsymbol{\theta}') = \log P(\hat{\boldsymbol{y}}|\boldsymbol{x}, \boldsymbol{\theta}) = \sum_{m=1}^{M} \log P(\hat{y}_m|\boldsymbol{x}, \boldsymbol{\theta})$
  Compute $\nabla_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}; \boldsymbol{\theta}')$ and use SGD to update $\boldsymbol{\theta}'$.
```

Bounding Box Annotations

To construct masks using the bounding boxes, the authors tru to use a simple method by considering all the pixels inside the box and positive and outside as negative, and in case of intersection of two bounding boxes, the pixel is assigned to the smallest one, but this gives a lot of false negative, so to refine these masks, they contrain the center pixel to the foreground and the pixels outside the box to the background and use a CRF with the appropriate setting the unary terms to refine the masks per class and then take the argmax for the mask of all classes.

And here is an illustration of the process:

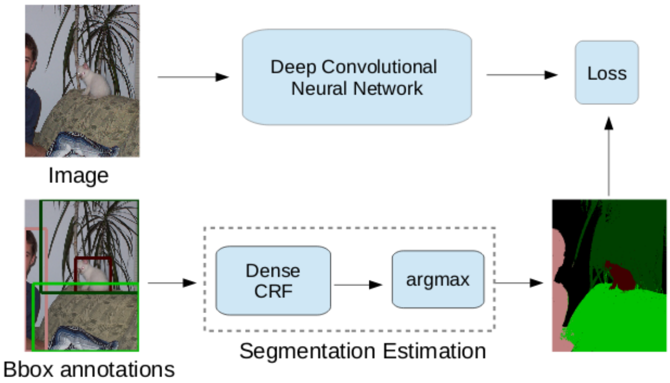


Figure 3. DeepLab model training from bounding boxes.

2.1 Mixed strong and weak annotations

When we have both strong and weakly annotated examples, we combine the two as follows, using the CE from the labeled examples, and EM for the weakly supervised examples:

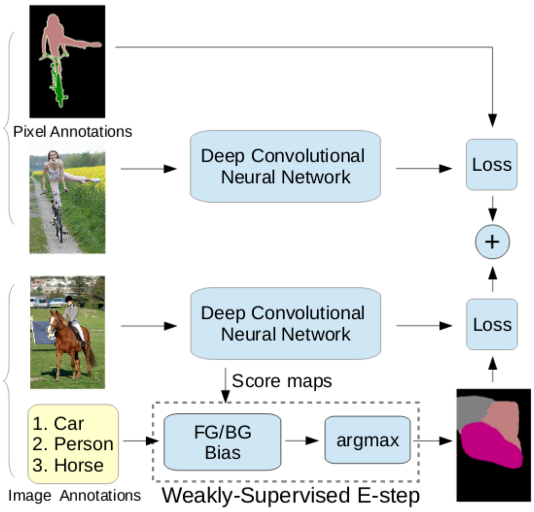


Figure 5. DeepLab model training on a union of full (strong labels) and image-level (weak labels) annotations.

3 Experiments

Method	#Strong	#Weak	val IOU
EM-Fixed (Weak)	-	10,582	20.8
EM-Adapt (Weak)	-	10,582	38.2
EM-Fixed (Semi)	200	10,382	47.6
	500	10,082	56.9
	750	9,832	59.8
	1,000	9,582	62.0
	1,464	5,000	63.2
	1,464	9,118	64.6
Strong	1,464	-	62.5
	10,582	-	67.6

Table 1. VOC 2012 *val* performance for varying number of pixel-level (strong) and image-level (weak) annotations (Sec. 4.3).

Method	#Strong	#Weak	test IOU
MIL-FCN [31]	-	10k	25.7
MIL-sppxl [32]	-	760k	35.8
MIL-obj [32]	BING	760k	37.0
MIL-seg [32]	MCG	760k	40.6
EM-Adapt (Weak)	-	12k	39.6
EM-Fixed (Semi)	1.4k	10k	66.2
	2.9k	9k	68.5
Strong [5]	12k	-	70.3

Table 2. VOC 2012 *test* performance for varying number of pixel-level (strong) and image-level (weak) annotations (Sec. 4.3).

Method	#Strong	#Box	val IOU
Bbox-Rect (Weak)	-	10,582	52.5
Bbox-EM-Fixed (Weak)	-	10,582	54.1
Bbox-Seg (Weak)	-	10,582	60.6
Bbox-Rect (Semi)	1,464	9,118	62.1
Bbox-EM-Fixed (Semi)	1,464	9,118	64.8
Bbox-Seg (Semi)	1,464	9,118	65.1
Strong	1,464	-	62.5
	10,582	-	67.6

Table 3. VOC 2012 *val* performance for varying number of pixel-level (strong) and bounding box (weak) annotations (Sec. 4.4).

Method	#Strong	#Box	test IOU
BoxSup [9]	MCG	10k	64.6
BoxSup [9]	1.4k (+MCG)	9k	66.2
Bbox-Rect (Weak)	-	12k	54.2
Bbox-Seg (Weak)	-	12k	62.2
Bbox-Seg (Semi)	1.4k	10k	66.6
Bbox-EM-Fixed (Semi)	1.4k	10k	66.6
Bbox-Seg (Semi)	2.9k	9k	68.0
Bbox-EM-Fixed (Semi)	2.9k	9k	69.0
Strong [5]	12k	-	70.3

Table 4. VOC 2012 *test* performance for varying number of pixel-level (strong) and bounding box (weak) annotations (Sec. 4.4).