WELDON: Weakly Supervised Learning of Deep Convolutional Neural Networks - Supplementary

Thibaut Durand, Nicolas Thome, Matthieu Cord
Sorbonne Universités, UPMC Univ Paris 06, CNRS, LIP6 UMR 7606, 4 place Jussieu, 75005 Paris
{thibaut.durand, nicolas.thome, matthieu.cord}@lip6.fr

1. WELDON Model

Image size I	Region scale α (%)	**L5** size
224×224	100	7×7
249×249	90	8×8
280×280	80	9×9
320×320	70	10×10
374×374	60	12×12
448×448	50	14×14
560×560	40	18×18
747×747	30	24×24

Table 1. Proposed multi-scale CNN feature extraction networks. Input images are rescale to IxI images, with I in the range $[224; 747]$. At each scale, regions span 224×224 areas, so that the region scale is $\alpha = 224/I$.

2. WELDON Ranking

2.1. Notations

We detail the formulation of the WELDON ranking instantiation given in Section 4 of the submitted paper, with the assumption $k = m$. We use a latent structured output ranking formulation, following [8]: our input is a set of N training images $\mathbf{x} = \{x_i\}$, $i \in \{1; N\}$, with their binary labels y_i, and our goal is to predict a ranking matrix $\mathbf{c} \in \mathcal{C}$ of size $N \times N$ providing an ordering of the training examples. The structured output feature map of our ranking instantiation, *i.e.* the computation of **L7** from **L6** (Figure 2 of the submitted paper), is:

$$\mathbf{L7}(\mathbf{x}, \mathbf{c}, \mathbf{h}) = \sum_{p \in \mathcal{P}} \sum_{n \in \mathcal{N}} c_{pn} \mathbf{W_7} \left[\mathbf{l}^{\mathbf{6}p}_{\mathbf{h}^{pn}} - \mathbf{l}^{\mathbf{6}n}_{\mathbf{h}^{np}} \right] \tag{1}$$

where $\mathcal{P}$ (resp. $\mathcal{N}$) is the set of positive (resp. negative) examples, n'_i is the number of regions for image i, and

$$\mathbf{h} = \{(\mathbf{h}^{pn}, \mathbf{h}^{np}) \in \{0,1\}^{n'_p} \times \{0,1\}^{n'_n}, \tag{2}$$
$$\sum_{z=1}^{n'_p} h^{pn}_z = k, \sum_{z=1}^{n'_n} h^{np}_z = k, (p,n) \in \mathcal{P} \times \mathcal{N}\}$$

$\mathbf{h}^{pn}$ is a vector which represent the selected region for image p when we consider the couple of image (p,n). $\mathbf{l}^{\mathbf{6}p}_{\mathbf{h}^{pn}} = \sum_{z=1}^{n'_p} h^{pn}_z \mathbf{l}^{\mathbf{6}p}_z$ is the feature map of image p with the selected regions $\mathbf{h}^{pn}$, where h^{pn}_z is the z-th value of vector $\mathbf{h}^{pn}$ and $\mathbf{l}^{\mathbf{6}p}_z$ is the feature map of region z.

2.2. Proof of Proposition 1 of the submitted paper

In this section, we detail the proof of Proposition 1 of the submitted paper, which generalizes [1] to top instances [5]. Inference consists in computing model prediction, *i.e.* computing $\hat{\mathbf{c}}$:

$$\hat{\mathbf{c}} = \arg\max_{\mathbf{c} \in \mathcal{C}} \mathbf{L8}(\mathbf{x}, \mathbf{c}) \tag{3}$$

where $\mathbf{L8}(\mathbf{x}, \mathbf{c})$ is given in Eq (3) of the submitted paper: $\mathbf{L8}(\mathbf{x}, \mathbf{c}) = s_{top}(\mathbf{L7}(\mathbf{x}, \mathbf{c})) + s_{low}(\mathbf{L7}(\mathbf{x}, \mathbf{c}))$. We prove that the inference is equivalent to a supervised inference, where each image x_i is represented by $s_{top}(\mathbf{W_7}\mathbf{L6}^i) + s_{low}(\mathbf{W_7}\mathbf{L6}^i)$. We show that the selected regions can be predicted independently to ranking $\mathbf{c}$:

$$\mathbf{L8}(\mathbf{x}, \mathbf{c}) = s_{top}(\mathbf{L7}(\mathbf{x}, \mathbf{c})) + s_{low}(\mathbf{L7}(\mathbf{x}, \mathbf{c})) \tag{4}$$

$$= \max_{\mathbf{h}} \sum_{p \in \mathcal{P}} \sum_{n \in \mathcal{N}} c_{pn} \mathbf{W_7} \left[\sum_{z=1}^{n'_p} h^{pn}_z \mathbf{l}^{\mathbf{6}p}_z - \sum_{z'=1}^{n'_n} h^{np}_{z'} \mathbf{l}^{\mathbf{6}n}_{z'} \right]$$
$$+ \min_{\mathbf{h}'} \sum_{p \in \mathcal{P}} \sum_{n \in \mathcal{N}} c_{pn} \mathbf{W_7} \left[\sum_{z=1}^{n'_p} {h'}^{pn}_z \mathbf{l}^{\mathbf{6}p}_z - \sum_{z'=1}^{n'_n} {h'}^{np}_{z'} \mathbf{l}^{\mathbf{6}n}_{z'} \right] \tag{5}$$

$$= \sum_{p \in \mathcal{P}} \sum_{n \in \mathcal{N}} \Bigg(\max_{(\mathbf{h}^p, \mathbf{h}^n)} c_{pn} \left[\sum_{z=1}^{n'_p} h^p_z \mathbf{W_7} \mathbf{l}^{\mathbf{6}p}_z - \sum_{z'=1}^{n'_n} h^n_{z'} \mathbf{W_7} \mathbf{l}^{\mathbf{6}n}_{z'} \right]$$
$$+ \min_{(\mathbf{h'}^p, \mathbf{h'}^n)} c_{pn} \left[\sum_{z=1}^{n'_p} {h'}^p_z \mathbf{W_7} \mathbf{l}^{\mathbf{6}p}_z - \sum_{z'=1}^{n'_n} {h'}^n_{z'} \mathbf{W_7} \mathbf{l}^{\mathbf{6}n}_{z'} \right] \Bigg) \tag{6}$$

The maximization (resp. minimization) can be decomposed for each term of the sum, so maximizing (resp. minimizing) the sum is equivalent to maximize (resp. minimize) each term of the sum. Now, we analyze the predicted regions with respect to c_{pn} value.

If $c_{pn} = 1$

$$\begin{aligned}
&\max_{(\mathbf{h}^p,\mathbf{h}^n)} \left[\sum_{z=1}^{n'_p} h^p_z \mathbf{W_7} \mathbf{l}^{\mathbf{6}p}_z - \sum_{z'=1}^{n'_n} h^n_{z'} \mathbf{W_7} \mathbf{l}^{\mathbf{6}n}_{z'} \right] \\
&+ \min_{(\mathbf{h'}^p,\mathbf{h'}^n)} \left[\sum_{z=1}^{n'_p} h'^p_z \mathbf{W_7} \mathbf{l}^{\mathbf{6}p}_z - \sum_{z'=1}^{n'_n} h'^n_{z'} \mathbf{W_7} \mathbf{l}^{\mathbf{6}n}_{z'} \right] \\
=& \left(\max_{\mathbf{h}^p} \sum_{z=1}^{n'_p} h^p_z \mathbf{W_7} \mathbf{l}^{\mathbf{6}p}_z + \min_{\mathbf{h'}^p} \sum_{z'=1}^{n'_p} h'^p_{z'} \mathbf{W_7} \mathbf{l}^{\mathbf{6}p}_{z'} \right) \qquad (7) \\
&- \left(\max_{\mathbf{h'}^n} \sum_{z=1}^{n'_n} h'^n_z \mathbf{W_7} \mathbf{l}^{\mathbf{6}n}_z + \min_{\mathbf{h}^n} \sum_{z'=1}^{n'_n} h^n_{z'} \mathbf{W_7} \mathbf{l}^{\mathbf{6}n}_{z'} \right) \\
=& s_{top}(\mathbf{W_7 L6}^p) + s_{low}(\mathbf{W_7 L6}^p) \qquad (8) \\
&- (s_{top}(\mathbf{W_7 L6}^n) + s_{low}(\mathbf{W_7 L6}^n))
\end{aligned}$$

If $c_{pn} = -1$

$$\begin{aligned}
&\max_{(\mathbf{h}^p,\mathbf{h}^n)} - \left[\sum_{z=1}^{n'_p} h^p_z \mathbf{W_7} \mathbf{l}^{\mathbf{6}p}_z - \sum_{z'=1}^{n'_n} h^n_{z'} \mathbf{W_7} \mathbf{l}^{\mathbf{6}n}_{z'} \right] \\
&+ \min_{(\mathbf{h'}^p,\mathbf{h'}^n)} - \left[\sum_{z=1}^{n'_p} h'^p_z \mathbf{W_7} \mathbf{l}^{\mathbf{6}p}_z - \sum_{z'=1}^{n'_n} h'^n_{z'} \mathbf{W_7} \mathbf{l}^{\mathbf{6}n}_{z'} \right] \\
=& \left(\max_{\mathbf{h}^p} \sum_{z=1}^{n'_p} h^p_z \mathbf{W_7} \mathbf{l}^{\mathbf{6}p}_z + \min_{\mathbf{h'}^p} \sum_{z'=1}^{n'_p} h'^p_{z'} \mathbf{W_7} \mathbf{l}^{\mathbf{6}p}_{z'} \right) \qquad (9) \\
&- \left(\max_{\mathbf{h'}^n} \sum_{z=1}^{n'_n} h'^n_z \mathbf{W_7} \mathbf{l}^{\mathbf{6}n}_z + \min_{\mathbf{h}^n} \sum_{z'=1}^{n'_n} h^n_{z'} \mathbf{W_7} \mathbf{l}^{\mathbf{6}n}_{z'} \right) \\
=& s_{top}(\mathbf{W_7 L6}^p) + s_{low}(\mathbf{W_7 L6}^p) \qquad (10) \\
&- (s_{top}(\mathbf{W_7 L6}^n) + s_{low}(\mathbf{W_7 L6}^n))
\end{aligned}$$

We notice that the predicted regions are the same in the two cases: the predicted regions can be fixed independently to the value of c_{pn}. The inference can be written as a supervised inference, where the region are fixed independently to the ranking matrix $\mathbf{c}$, and each image x_i is represented by $s_{top}(\mathbf{W_7 L6}^i) + s_{low}(\mathbf{W_7 L6}^i)$.

3. Optimization - Gradient Computation

In this section, we detail some gradient computations: soft-max, logistic regression, and ranking AP.

Soft-max The equation of the soft-max is:

$$\mathbf{L9}(c) = \frac{e^{\mathbf{L8}(c)}}{\sum_{c'} e^{\mathbf{L8}(c')}} \tag{11}$$

The gradient is:

$$\frac{\partial \mathbf{L9}(c)}{\partial \mathbf{L8}(c'')} = \frac{e^{\mathbf{L8}(c)}}{\left(\sum_{c'} e^{\mathbf{L8}(c')}\right)^2} \delta_{c=c''} - \frac{e^{\mathbf{L8}(c)+\mathbf{L8}(c'')}}{\left(\sum_{c'} e^{\mathbf{L8}(c')}\right)^2} \tag{12}$$

where

$$\delta_p = \begin{cases} 1 & \text{if } p \text{ is true} \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

Logistic regression The equation of the logistic regression is:

$$\mathbf{L9}(c) = \left(1 + e^{-y\mathbf{L8}(c)}\right)^{-1} \tag{14}$$

where $y = +1$ if the object is present, and $y = -1$ otherwise. The gradient is:

$$\frac{\partial \mathbf{L9}(c)}{\partial \mathbf{L8}(c)} = \frac{-ye^{-y\mathbf{L8}(c)}}{1 + e^{-y\mathbf{L8}(c)}} \tag{15}$$

Ranking AP We detail the gradient of $\mathbf{L8}(\mathbf{c})$ (Section 2 in supplementary) with respect to $\mathbf{W_7}$:

$$\begin{aligned}
\frac{\partial \mathbf{L8}(\mathbf{c})}{\partial \mathbf{W_7}} = \sum_{p\in\mathcal{P}} \sum_{n\in\mathcal{N}} c_{pn} \Bigg[& \left(\sum_{z=1}^{n'_p} (h^{p+}_z + h^{p-}_z) \mathbf{l}^{\mathbf{6}p}_z \right) \\
& + \left(\sum_{z'=1}^{n'_n} (h^{n+}_{z'} + h^{n-}_{z'}) \mathbf{l}^{\mathbf{6}n}_{z'} \right) \Bigg]
\end{aligned} \tag{16}$$

where

$$\mathbf{h}^{p+} = \max_{\mathbf{h}} \sum_{z=1}^{n'_p} h_z \mathbf{W_7} \mathbf{l}^{\mathbf{6}p}_z, \quad \text{s.t.} \sum_{z=1}^{n'_p} h_z = k \tag{17}$$

$$\mathbf{h}^{p-} = \min_{\mathbf{h}} \sum_{z=1}^{n'_p} h_z \mathbf{W_7} \mathbf{l}^{\mathbf{6}p}_z, \quad \text{s.t.} \sum_{z=1}^{n'_p} h_z = k \tag{18}$$

$$\mathbf{h}^{n+} = \max_{\mathbf{h}} \sum_{z=1}^{n'_n} h_z \mathbf{W_7} \mathbf{l}^{\mathbf{6}n}_z, \quad \text{s.t.} \sum_{z=1}^{n'_n} h_z = k \tag{19}$$

$$\mathbf{h}^{n-} = \min_{\mathbf{h}} \sum_{z=1}^{n'_n} h_z \mathbf{W_7} \mathbf{l}^{\mathbf{6}n}_z, \quad \text{s.t.} \sum_{z=1}^{n'_n} h_z = k \tag{20}$$

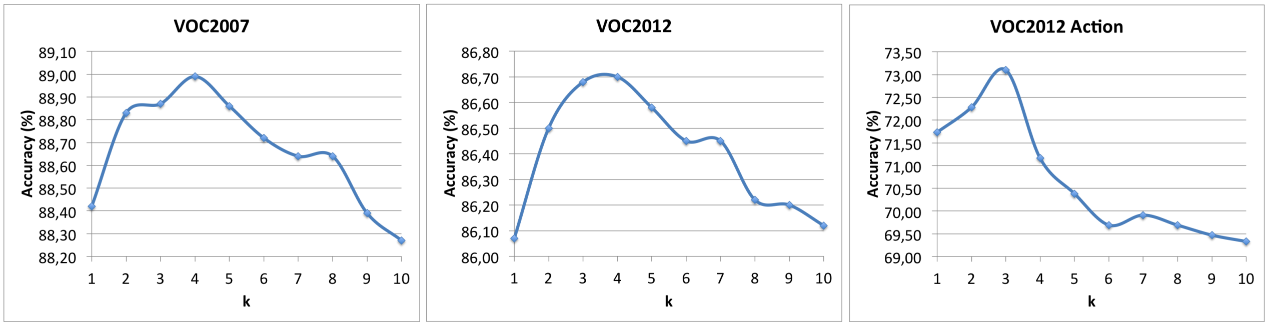

Figure 1. MAP with respect to the number of top/low instances at scale $\alpha = 30\%$

4. Experiments

VOC 2007 [2] VOC 2007 is a multi-label dataset with 20 object classes. The models are learned on the *trainval* set (∼5.000 images) and performances are evaluated on *test* set (∼5.000 images). Each binary classification problem is evaluated with AP, and the final performance is the average of all binary performances.

VOC 2012 [3] VOC 2012 is a multi-label dataset with 20 object classes. The models are learned on the *train* set (∼5.700 images) and performances are evaluated on *val* set (∼5.800 images). The performance evaluation is the same that VOC 2007.

VOC 2012 Action [3] VOC 2012 Action is a multi-label dataset with 10 action classes. The models are learned on the *train* set (∼2.000 images) and performances are evaluated on *val* set (∼2.000 images). Bounding boxes are not used during training or testing. The performance evaluation is the same that VOC 2007.

COCO [6] COCO is a multi-label dataset with 80 object classes. The models are learned on the *train2014* set (∼80.000 images) and performances are evaluated on *val2014* set (∼40.000 images).

MIT67 [7] The dataset has 67 classes of cluttered indoor scenes. We use the standard train/test split with 5360 (resp. 1340) training (resp. testing) images. The performances are evaluated with multi-class accuracy.

15 Scene [4] The dataset has 15 classes of scenes. We use 5 random train/test splits with 1500 (resp. 2985) training (resp. testing) images. The performances are evaluated with multi-class accuracy.

References

[1] T. Durand, N. Thome, and M. Cord. MANTRA: Minimum Maximum Latent Structural SVM for Image Classification and Ranking. In *ICCV*, 2015. 1

[2] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html. 3

[3] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html. 3

[4] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, 2006. 3

[5] W. Li and N. Vasconcelos. Multiple instance learning for soft bags via top instances. June 2015. 1

[6] 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 *European Conference on Computer Vision (ECCV)*, Zürich, September 2014. 3

[7] A. Quattoni and A. Torralba. Recognizing indoor scenes. In *CVPR*, 2009. 3

[8] Y. Yue, T. Finley, F. Radlinski, and T. Joachims. A support vector method for optimizing average precision. In *SIGIR*, 2007. 1