Mass Displacement Networks

Natalia Neverova
nneverova@fb.com
Facebook AI Research
Paris, France
Iasonas Kokkinos
iasonask@fb.com

1 Back-propagation through an MDN module

The input-output mapping defined by Eq. 1 of the main manuscript is differentiable with respect to both input functions, $\mathbf{o}(\mathbf{x})$, $c(\mathbf{x})$, and as such lends itself to end-to-end training with back-propagation. Given a gradient signal $\delta(\cdot)=\frac{\partial L}{\partial m(\cdot)}$ that dictates how the output layer activations should change to decrease the loss L, we obtain the update equations for $c(\cdot)$ and $\mathbf{o}(\cdot)=(o_x(\cdot),o_y(\cdot))$ through the following chain rule:

$$\frac{\partial L}{\partial c(\mathbf{x})}=\sum_{\mathbf{x}_0}\delta_{\mathbf{x}_0}\frac{\partial m(\mathbf{x}_0)}{\partial c(\mathbf{x})},\quad \frac{\partial L}{\partial \{o_x/o_y\}(\mathbf{x})}=\sum_{\mathbf{x}_0}\delta_{\mathbf{x}_0}\frac{\partial m(\mathbf{x}_0)}{\partial \{o_x/o_y\}(\mathbf{x})}, \tag{1}$$

where the summation runs over the top-layer neurons $\mathbf{x}_0$ that send gradients back to neuron $\mathbf{x}$. Turning to the computation of the partial derivatives in equation above, the use of displacement fields means that we no longer have a standard convolutional layer; an input position $\mathbf{x}$ can potentially influence any other output position $\mathbf{x}_0$, as dictated by Eq. 2 of the manuscript. For convenience we rewrite this as follows:

$$m(\mathbf{x}_0)=1-\prod_{\mathbf{x}}\big(1-w(\mathbf{x},\mathbf{x}_0)c(\mathbf{x})\big),\quad \text{where}\ \ w(\mathbf{x},\mathbf{x}_0)=K\big(\mathbf{x}_0-[\mathbf{x}+\mathbf{o}(\mathbf{x})]\big), \tag{2}$$

indicates the amount of influence of $\mathbf{x}$ on $\mathbf{x}_0$. Using the same steps as in [2], in case of a Gaussian kernel K we have:

$$\frac{\partial m(\mathbf{x}_0)}{\partial c(\mathbf{x})}=w(\mathbf{x},\mathbf{x}_0)\frac{1-m(\mathbf{x}_0)}{1-w(\mathbf{x})c(\mathbf{x})},\quad \frac{\partial m(\mathbf{x}_0)}{\partial o_x(\mathbf{x})}=\frac{\partial m(\mathbf{x}_0)}{\partial w(\mathbf{x},\mathbf{x}_0)}K'\big(\mathbf{x}_0-[\mathbf{x}+\mathbf{o}(\mathbf{x})]\big)\big[x_0-[x+o_x(\mathbf{x})]\big]$$

where $x_0,x,o_x(\mathbf{x})$ are the horizontal components of $\mathbf{x}_0,\mathbf{x},\mathbf{o}[\mathbf{x}]$ respectively.

2 On additive vs noisy-or aggregation rule

The main problem with the additive aggregation schema is that we cannot simultaneously guarantee that the input and output fields both lie in $[0,1]$, so that they can be trained with the cross-entropy loss, and that a confident posterior at $\mathbf{x}$ will confidently support its displaced replica at $\mathbf{x}+\mathbf{o}(\mathbf{x})$, i.e. $K(\mathbf{0})=1$.

Method	AP^{100}	AP^{100}_{50}	AP^{100}_{75}	AP^{100}_{M}	AP^{100}_{L}	AR^{1}	AR^{10}	AR^{100}	AR^{100}_{M}	AR^{100}_{L}
Mask R-CNN, bb	51.5	82.5	55.0	59.4	68.5	18.2	52.2	59.8	66.9	76.3
Mask R-CNN, bb+mask	52.2	83.1	55.9	59.8	69.7	18.4	52.8	60.4	66.9	77.2
Mask R-CNN, bb+keypoints	51.6	81.4	55.3	60.1	69.7	18.3	52.4	60.0	67.4	77.0
Mask R-CNN-MDN, bb+keypoints	52.0	81.8	55.9	60.7	70.0	18.5	52.9	60.3	67.7	77.3
Mask R-CNN, bb+mask+keypoints	51.7	81.6	55.6	60.1	69.8	18.4	52.6	60.3	67.6	77.2
Mask R-CNN-MDN, bb+mask+keypoints	52.2	81.6	56.4	60.6	71.1	18.7	53.3	61.0	68.1	78.3

Table 1: Object detection performance (bounding box AP/AR) on COCO `minival`, *person* class.

The voting transformation is described in Eg. 1 of the main manuscript as follows:

$$m(\mathbf{x}_o) = \sum_{\mathbf{x}} K_\sigma(\mathbf{x}_o - [\mathbf{x} + \mathbf{o}(\mathbf{x})])c(\mathbf{x}). \tag{3}$$

If one interprets both $c(\cdot)$ and $m(\cdot)$ as fields of posterior probability values, one has:

$$c(\mathbf{x}) = 1, \;\; \mathbf{o}(\mathbf{x}) = \mathbf{x} - \mathbf{x}_o \;\; \rightarrow m(\mathbf{x}_o) = \sum_{\mathbf{x}} K(\mathbf{x}) > 1. \tag{4}$$

In this case, ensuring that $m(\mathbf{x}_o) \leq 1$ would mean that we must use a normalized kernel, e.g. $K(\mathbf{x}) = \frac{1}{2\pi\sigma^2} \exp(-\frac{\|\mathbf{x}\|^2}{2\sigma^2})$, as used in [1]. One counter-intuitive resulting property is that the input-output mapping function defined by Eq. 1 of the manuscript can result in a decrease, rather accumulation of evidence. Consider in particular a perfectly-localized and perfectly-confident local evidence signal expressed in the form of a delta function centered at $\mathbf{x}$:

$$c(\mathbf{x}) \;\; = \;\; \begin{cases} 1, & \mathbf{x} = \mathbf{x} \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

The result of voting according to Eq. 1 of the main manuscript would then be a blurred support map that only yields a maximal support of $\frac{1}{2\pi\sigma^2}$ to $\mathbf{x} + \mathbf{o}(\mathbf{x})$:

$$m(\mathbf{x}) \;\; = \;\; \frac{1}{2\pi\sigma^2} \exp\left(-\frac{\|\mathbf{x} - (\mathbf{x} + \mathbf{o}(\mathbf{x}))\|^2}{2\sigma^2} \right). \tag{6}$$

For a large value of σ this can result in an arbitrarily low value of $m(\mathbf{x})$, which is counter-intuitive, given the originally strong evidence at $\mathbf{x}$. At the root of this problem lies the operation of summing probabilities, which is a common operation when marginalizing over hidden variables, but does not make sense as a method of accumulating evidence [3].

3 Additional experiments

Finally, we perform an ablation study in the multi-task setting to analyze the effect of the introduced cross-part MDN module on the performance of other brances of Mask R-CNN, namely bounding box regressor (Table 1) and predictor of binary masks (Table 2) for the *person* class from COCO `minival`. In both cases, we observed consistent improvements in performance across the whole set of evaluation metrics. However, in the presence of the MDN module, activating the mask branch does not further improve the quality of pose estimation as in the baseline case.

Method	AP^{100}	AP^{100}_{50}	AP^{100}_{75}	AP^{100}_{M}	AP^{100}_{L}	AR^{1}	AR^{10}	AR^{100}	AR^{100}_{M}	AR^{100}_{L}
Mask R-CNN, bb+mask	44.8	79.4	45.9	50.5	64.4	16.7	47.0	53.3	59.4	70.7
Mask R-CNN, bb+mask+keypoints	45.0	78.5	47.3	51.4	65.2	16.8	47.3	53.8	60.6	71.4
Mask R-CNN-MDN, bb+mask+keypoints	45.6	78.3	48.1	52.0	66.1	17.0	48.1	54.5	61.3	72.3

Table 2: Instance segmentation performance (mask AP/AR) on COCO `minival`, *person* class.

References

[1] George Papandreou, Tyler Zhu, Nori Kanazawa, Alexander Toshev, Jonathan Tompson, Chris Bregler, and Kevin P. Murphy. Towards accurate multi-person pose estimation in the wild. *CVPR*, 2017.

[2] Paul A. Viola, John C. Platt, and Cha Zhang. Multiple instance boosting for object detection. *NIPS*, 2005.

[3] Christopher K. I. Williams and Moray Allan. On a connection between object localization with a generative template of features and pose-space prediction methods. Technical report, Edinburgh University, 2006.