End-to-End Integration of a Convolutional Network, Deformable Parts Model and Non-Maximum Suppression

Li Wan, David Eigen, Rob Fergus
Dept. of Computer Science, Courant Institute, New York University

Deformable Parts Models and Convolutional Networks each have achieved notable performance in object detection. Yet these two approaches find their strengths in complementary areas: DPMs are well-versed in object composition, modeling fine-grained spatial relationships between parts; likewise, ConvNets are adept at producing powerful image features, having been discriminatively trained directly on the pixels.

In this paper, we propose a new model (shown in Fig. 1) that combines these two approaches, obtaining the advantages of each. We train this model using a new structured loss function that considers all bounding boxes within an image, rather than isolated object instances. This enables the non-maximal suppression (NMS) operation, previously treated as a separate post-processing stage, to be integrated into the model. This allows for discriminative training of our combined Convnet + DPM + NMS model in end-to-end fashion. We evaluate our system on PASCAL VOC 2007 and 2011 datasets, achieving competitive results on both benchmarks.

For a given input image x, we first construct an image pyramid of appearance features $\phi_A(x)$ using the first five layers of a Convolutional Network pre-trained for the ImageNet Classification task. We first train an eight layer classification model, which is composed of five convolutional feature extraction layers, plus three fully-connected classification layers. After this network has been trained, we throw away the three fully-connected layers, replacing them instead with the DPM. The five convolutional layers are then used to extract appearance features.

The deformation layer finds the optimal part locations, accounting for both apperance and a deformation cost that models the spatial relation of the part to the root. Given appearance scores $F^{\text{part}}_{v,p}$, part location p relative to the root, and deformation parameters $w_{D,v,p}$ for each part, the deformed part responses are the following (input variables (x_s, y) omitted):

$$F^{\text{def}}_{v,p} = \max_{\delta_i,\delta_j} F^{\text{part}}_{v,p}[p_i+\delta_i, p_j+\delta_j] - w^{\text{part}}_{D,v,p}\phi_D(\delta_i,\delta_j) \quad (1)$$

where $F^{\text{part}}_{v,p}[p_i+\delta_i, p_j+\delta_j]$ is the part response map $F^{\text{part}}_{v,p}(x_s,y)$ shifted by spatial offset $(p_i+\delta_i, p_j+\delta_j)$, and $\phi_D(\delta_i,\delta_j) = [|\delta_i|, |\delta_j|, \delta_i^2, \delta_j^2]^T$ is the shape deformation feature. $w^{\text{part}}_{D,y,v} \geq 0$ are the deformation weights.

Combining the scores of root, parts and object views is done using an AND-like accumulation over parts to form a score F_v for each view v, followed by an OR-like maximum over views to form the final object score F:

$$F_v(x_s,y) = F^{\text{root}}_v(x_s,y) + \sum_{p\in\text{parts}} F^{\text{def}}_{v,p}(x_s,y) \quad (2)$$

$$F(x_s,y) = \max_{v\in\text{views}} F_v(x_s,y) \quad (3)$$

$F(x_s,y)$ is then the final score map for class y at scale s, given the image x as shown in Fig. 2.

Final-prediction loss that takes into account the NMS step used in inference. In contrast to bootstrapping with a hard negative pool, such as in [2] [1], we consider each image individually when determining positive and negative examples, accounting for NMS and the views present in the image itself.

NMS stage produces a set of assignments predicted by the model $A = \{(b_i, y_i, r_i)_{i=1\ldots n}\}$ from the set B of all possible assignments. We compose the loss using two terms, $C(A)$ and $C(A')$. The first, $C(A)$, measures the cost incurred by the assignment currently predicted by the model, while $C(A')$ measures the cost incurred by an assignment close to the ground truth. The

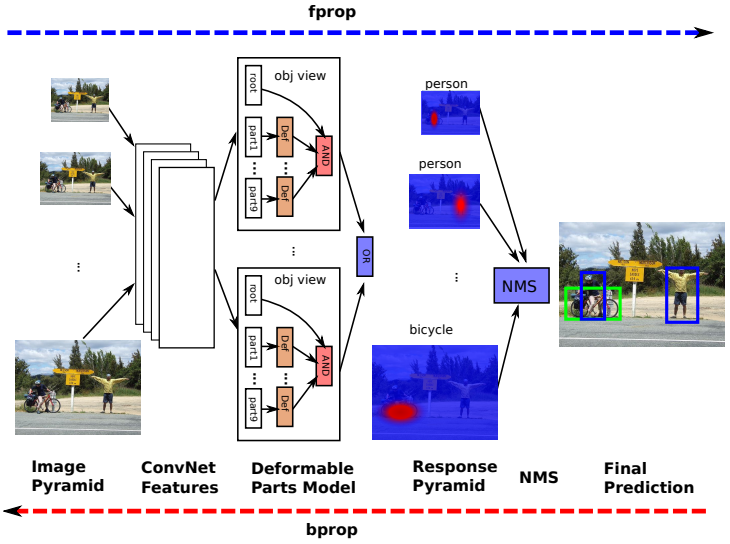

Figure 1: An overview of our system: (i) a convolutional network extracts features from an image pyramid; (ii) a set of deformable parts models (each capturing a different view) are applied to the convolutional feature maps; (iii) non-maximal suppression is applied to the resulting response maps, yielding bounding box predictions.

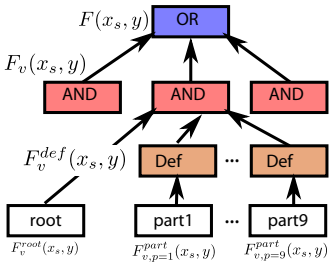

Figure 2: Overview of the top part of our network architecture

current prediction cost $C(A)$ is:

$$C(A) = \underbrace{\sum_{(b_i,y_i,r_i)\in A} H(r_i,y_i)}_{C^P(A)} + \underbrace{\sum_{(b_j,y_j,r_j)\in S(A)} H(r_j,0)}_{C^N(A)} \quad (4)$$

where $H(r,y) = I(y>0)\max(0,1-r)^2 + I(y=0)\max(0,r+1)^2$ i.e. a squared hinge error.

[1] Pedro F. Felzenszwalb, Ross B. Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(9):1627–1645, September 2010. ISSN 0162-8828. doi: 10.1109/TPAMI.2009.167. URL http://dx.doi.org/10.1109/TPAMI.2009.167.

[2] Long (Leo) Zhu, Yuanhao Chen, Alan Yuille, and William Freeman. Latent hierarchical structural learning for object detection. *2010 IEEE Conference on Computer Vision and Pattern Recognition*, 0:1062–1069, 2010. doi: http://doi.ieeecomputersociety.org/10.1109/CVPR.2010.5540096.

This is an extended abstract. The full paper is available at the Computer Vision Foundation webpage.