Feedforward semantic segmentation with zoom-out features

Mohammadreza Mostajabi and Payman Yadollahpour and Gregory Shakhnarovich
Toyota Technological Institute at Chicago

We introduce a purely feed-forward architecture for semantic segmentation. We map small image elements (superpixels) to rich feature representations extracted from a sequence of nested regions of increasing extent. These regions are obtained by "zooming out" from the superpixel all the way to scene-level resolution. This approach exploits statistical structure in the image and in the label space without setting up explicit structured prediction mechanisms, and thus avoids complex and expensive inference. Instead superpixels are classified by a feedforward multilayer network. Our architecture achieves **69.6%** average accuracy on the PASCAL VOC 2012 test set.

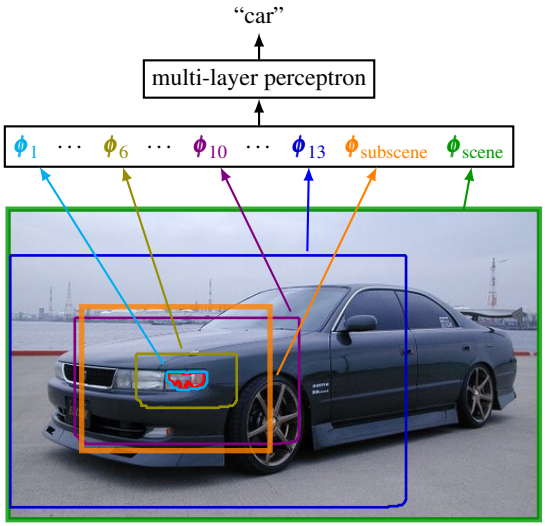

Figure 1: Schematic description of our approach. Multiple levels of "zoom-out" features (different colors) are extracted by layers of a deep convolutional neural networks; all of these are concatendated to form a zoom-out representation for a superpixel (red), which is classified by a multi-layer perceptron.

We depart from the commonly used approach to segmentation as structured prediction, and approach semantic segmentation as a single-stage classification task, in which each image element (superpixel) is labeled by a feedforward model, based on evidence computed from the image. The "secret" behind our method is that the evidence used in the feedforward classification is not computed from a small local region in isolation, but collected from a sequence of levels, obtained by "zooming out" from the close-up view of the superpixel. Starting from the superpixel itself, to a small region surrounding it, to a larger region around it and all the way to the entire image, we compute a rich feature representation at each level and combine all the features before feeding them to a classifier. This allows us to exploit statistical structure in the label space and dependencies between image elements at different resolutions without explicitly encoding these in a complex, and likely intractable, model.

The zoom-out architecture we propose is sketched out in Figure 1. We define a zoom-out level associated with the output of every convolutional layer in the recently proposed 16 layer network, pre-trained on ImageNet classification task [5]. A feature map computed by a convolutional layer with k filters assigns a k-dimensional feature vector to each receptive field of that layer. We upsample this feature map to the original image resolution, yielding a k-dimensional feature vector for every *pixel* in the image. Pooling these vectors over a superpixel gives us a k-dimensional feature vector describing that superpixel at a given zoom-out level. We have 13 such layers. In addition, we run a sub-image centered on the superpixel, as well as the entire image, through the convnet and take the activations of the last fully-connected layer as features for sub-scene and scene levels, respectively. Concatenating these features gives us a 12,416-dim. feature representation per superpixel, which is classified by a neural network with three fully connected layers. We train the classifier by stochastic gradient descent with weighted loss (superpixels from less frequent classes assigned higher weights).

Our work shares some ideas with other concurrent efforts. The main differences with [2, 3] are (i) that we incorporate a much wider range of zoom-out levels, (ii) we combine features, rather than predictions, across levels. Another difference is that these methods fine-tune the convnets on the segmentation task as part of an end-to-end learning, while we use a network pre-trained on the ImageNet classification task as-is. Despite this lack of fine-tuning, we achieve a significantly better performance on VOC 2012 test set (Table 1).

We also obtain state of the art results on Stanford Background data set, with mean IoU of 80.9 and per-pixel accuracy of 86.1%.

More recent work [1, 6] uses a combination of convnets for classification with a CRF framework to explicitly impose higher-order constraints, such as smoothness, on the predicted segmentation map. These methods achieve results better than ours, although the gap is small, considering that they, too, fine-tune the convnets to the task while we do not. Training the systems on the recently released COCO dataset further improves accuracy on VOC test. We plan to pursue all of these directions (end-to-end training, using additional data, and adding inference as a cleanup layer on top of the predictions) to improve our system.

Method	CRF	fine-tuned	IoU
ours	no	no	69.6
hypercolumns [2]	no	yes	62.6
FCN-8s[3]	no	yes	62.2
Oxford-TVG-CRF-RNN[6]	yes	yes	70.4
DeepLab-MSc-CRF-LargeFOV[1]	yes	yes	71.6

Table 1: Results on VOC 2012 test. IoU: intersection over union, averaged over 21 classes.

[1] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint* http://arxiv.org/abs/1412.7062, 2015.

[2] B. Hariharan, P. Aberláez an R. Girshick, and J. Malik. Hypercolumns for object segmentation and fine-grained localization. *arXiv preprint* http://arxiv.org/abs/1411.5752, 2015.

[3] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. *arXiv preprint* http://arxiv.org/abs/1411.4038, 2015.

[4] M. Mostajabi, P. Yadollahpour, and G. Shakhnarovich. Feedforward semantic segmentation with zoom-out features. *arXiv preprint* http://arxiv.org/abs/1412.0774, 2015.

[5] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint* http://arxiv.org/abs/1409.1556, 2014.

[6] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. S. Torr. Conditional random fields as recurrent neural networks. *arXiv preprint* http://arxiv.org/abs/1502.03240, 2015.

This is an extended abstract. The full paper is available at the Computer Vision Foundation webpage.A long version of the paper with additional experiments, examples and implementation details is available on arXiv [4].