Deep learning

8.4. Networks for semantic segmentation

François Fleuret

https://fleuret.org/dlc/

The historical approach to image segmentation was to define a measure of similarity between pixels, and to cluster groups of similar pixels. Such approaches account poorly for semantic content.

The deep-learning approach re-casts semantic segmentation as pixel classification, and re-uses networks trained for image classification by making them fully convolutional.

Shelhamer et al. (2016) proposed the FCN ("Fully Convolutional Network") that uses a pre-trained classification network (e.g. VGG 16 layers).

The fully connected layers are converted to 1×1 convolutional filters, and the final one retrained for 21 output channels (VOC 20 classes + "background").

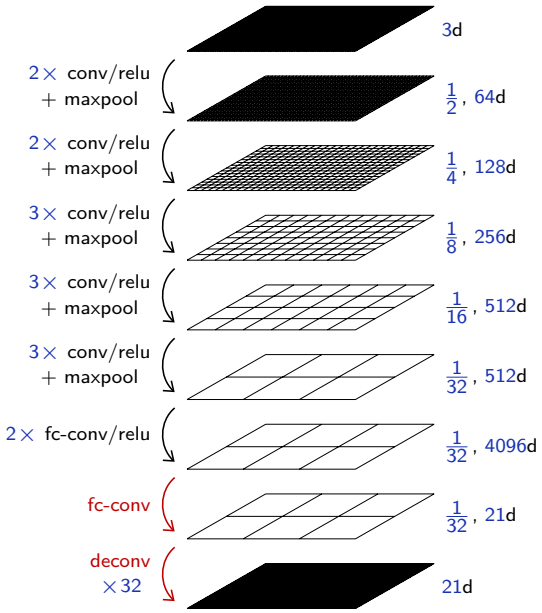
Since VGG16 has 5 max-pooling with 2×2 kernels, with proper padding, the output is $1/2^5 = 1/32$ the size of the input.

This map is then up-scaled with a transposed convolution layer with kernel 64×64 and stride 32×32 to get a final map of same size as the input image.

Training is achieved with full images and pixel-wise cross-entropy, starting with a pre-trained VGG16. All layers are fine-tuned, although fixing the up-scaling transposed convolution to bilinear does as well.

Notes

The added "background" class is added for pixels that do not belong to any of the defined object and avoid forcing the network to make a inconsistent choice.
Since segmentation aims at classifying the individual pixels, the size of the final tensor should be of the same size as the input image. Since the activation maps have been reduced by pooling operations, the size has to be increase back.
As seen in lecture 7.1. "Transposed convolutions" up-sampling an activation map can be done with bilinear interpolation, or transposed convolution layers.

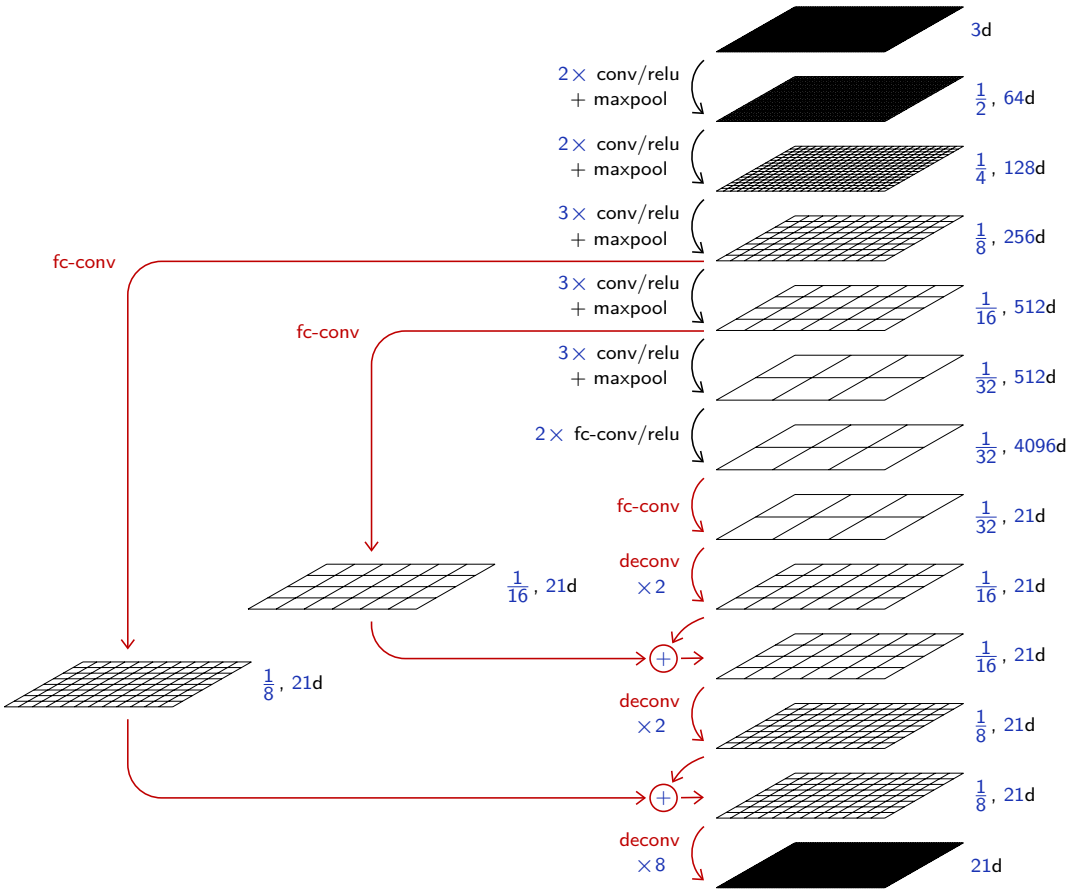


Notes

The last fc-conv makes the prediction output of the 21 classes from the activation maps which is $1/32$ of the original image size. Then, the classification maps are up-sampled by a 32 factor up to the original image size by the transposed convolution layer.

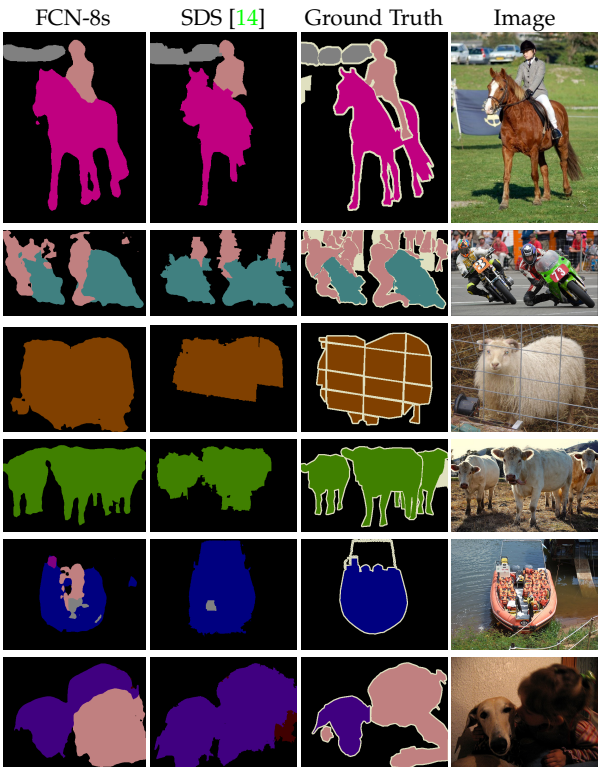
Although the FCN achieved almost state-of-the-art results when published, its main weakness is the coarseness of the signal from which the final output is produced (1/32 of the original resolution).

Shelhamer et al. proposed an additional element, that consists of using the same prediction/up-scaling from intermediate layers of the VGG network.

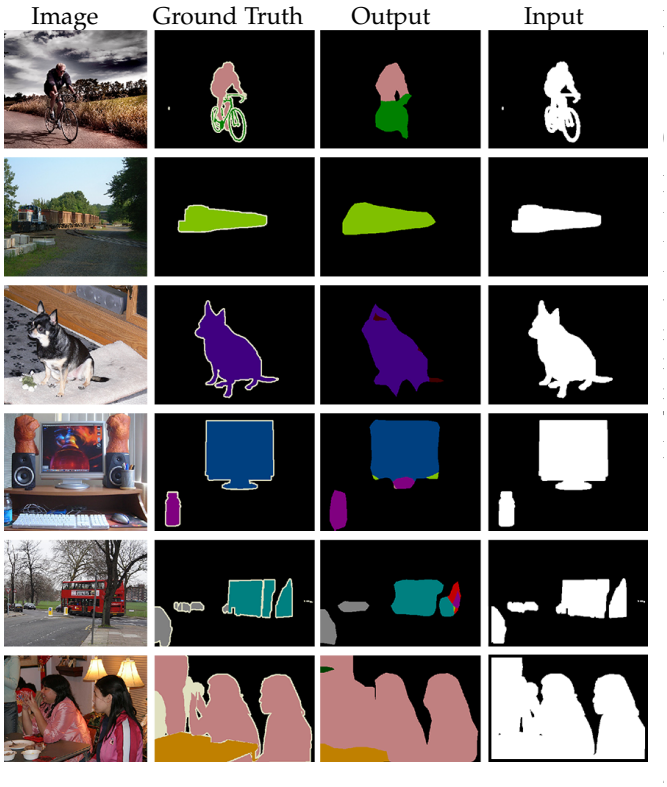


Notes

The coarseness of the prediction is reduced by adding intermediate predictions which are less refined in term of features, but of greater resolution.



Left column is the best network from Shelhamer et al. (2016).



Results with a network trained from mask only (Shelhamer et al., 2016).

The most sophisticated object detection methods achieve **instance segmentation** and estimate a segmentation mask per detected object.

Mask R-CNN (He et al., 2017) adds a branch to the Faster R-CNN model to estimate a mask for each detected region of interest.

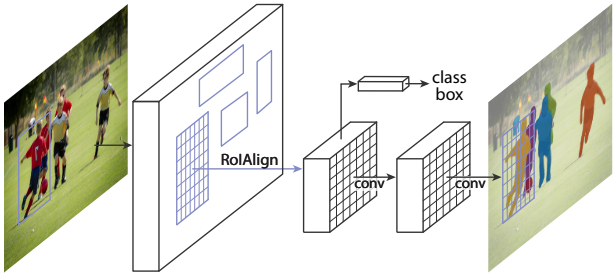


Figure 1. The **Mask R-CNN** framework for instance segmentation.

(He et al., 2017)

Notes

Instance segmentation aims at not only classifying the individual pixels in the image but also the instance of the class when the same object is present multiple times, e.g. "car #1", "car #2".

Figure 5. More results of **Mask R-CNN** on COCO test images, using ResNet-101-FPN and running at 5 fps, with 35.7 mask AP (Table 1).

(He et al., 2017)

It is noteworthy that for detection and semantic segmentation, there is an heavy re-use of large networks trained for classification.

The models themselves, as much as the source code of the algorithm that produced them, or the training data, are generic and re-usable assets.

References

K. He, G. Gkioxari, P. Dollár, and R. Girshick. **Mask R-CNN**. In International Conference on Computer Vision, pages 2980–2988, 2017.

E. Shelhamer, J. Long, and T. Darrell. **Fully convolutional networks for semantic segmentation**. CoRR, abs/1605.06211, 2016.