Ensembles of Densely-Connected CNNs with Label-Uncertainty for Brain Tumor Segmentation

Richard McKinley(✉), Raphael Meier, and Roland Wiest

Support Centre for Advanced Neuroimaging,
University Institute of Diagnostic and Interventional Neuroradiology,
Inselspital, Bern University Hospital, Bern, Switzerland
richard.mckinley@gmail.com

Abstract. We introduce a new family of classifiers based on our previous DeepSCAN architecture, in which densely connected blocks of dilated convolutions are embedded in a shallow U-net-style structure of down/upsampling and skip connections. These networks are trained using a newly designed loss function which models label noise and uncertainty. We present results on the testing dataset of the Multimodal Brain Tumor Segmentation Challenge 2018.

1 Introduction

We present a network architecture for semantic segmentation, heavily inspired by the recent Densenet architecture for image classification [7], in which pooling layers are replaced by heavy use of dilated convolutions [16]. Densenet employs dense blocks, in which the output of each layer is concatenated with its input before passing to the next layer. A typical Densenet architecture consists of a number of dense blocks separated by transition layers: the transition layers contain a pooling operation, which allows some degree of translation invariance and downsamples the feature maps. A Densenet architecture adapted for semantic segmentation was presented in [8], which adopted the now standard approach of U-net [15]: a downsampling path, followed by an upsampling path, with skip connections passing feature maps of the sample spatial dimension from the downsampling path to the upsampling path.

In a previous paper [12], we described an alternative architecture adapting Densenet for semantic segmentation: in this architecture, which we called DeepSCAN, there are no transition layers and no pooling operations. Instead, dilated convolutions are used to increase the receptive field of the classifier. The absence of transition layers means that the whole network can be seen as a single dense block, enabling gradients to pass easily to the deepest layers. While we believe that this approach offers many advantages over U-net, by avoiding pooling and upscaling, this comes at the price of very high memory consumption, since all feature maps are present at the resolution of the final segmentation image. This restricts the possible depth, batch size, and input patch size of the network.

A. Crimi et al. (Eds.): BrainLes 2018, LNCS 11384, pp. 456–465, 2019.
https://doi.org/10.1007/978-3-030-11726-9_40

In this paper we describe a family of CNN models for segmentation which represent a continuum from our previously described DeepSCAN models to U-net-like models, in which a pooling-free dense net is embedded inside a U-net style network. This allows the dense part of the network to operate at a lower resolution, improving memory efficiency while maintaining many good properties of the original DeepSCAN architecture.

We describe the general architecture of the family of DeepSCAN models, plus the particular features of the network as applied to brain tumor segmentation, including pre-processing, data augmentation, and a new uncertainty-motivated loss function. We report preliminary results on the validation portion of the BRATS 2018 dataset.

2 The DeepSCAN Family of Models

We describe here the constituent parts of the DeepSCAN family of models.

2.1 Densely Connected Layers and Densenet

Densenet [7] is a recently introduced architecture for image classification. The fundamental unit of a Densenet architecture is the densely connected block, or dense block. Such a block consists of a number of consecutive dense units, as pictured in Fig. 1. In such a unit, the output of each convolutional layer (where a layer here means some combination of convolutional filters, non-linearities and batch normalization) is concatenated to its input before passing to the next layer. The goal behind Densenet is to build an architecture which supports the training of very deep networks: the skip connections implicit in the concatenation of filter maps between layers allows the flow of gradients directly to those layers, providing an implicit deep supervision of those layers.

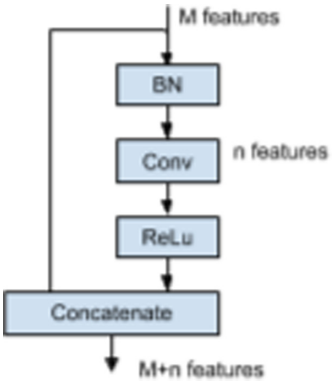

Fig. 1. A Dense unit, as used in the DeepSCAN architecture

In the original Densenet architecture, which has state-of-the-art performance on the CIFAR image recognition task, dense blocks are combined with transition blocks: non-densely connected convolutional layers, followed by a maxpooling layer. This helps to control parameter explosion (by limiting the size of the input to each dense block), but also means that the deep supervision is not direct, at the lowest layers of the network. This Dense-plus-transition architecture was also adopted by Jegou et al. [8], whose Tiramisu network is a U-net-style variation of the Densenet architecture designed for semantic segmentation.

In our previous paper [12], we dispensed with the transition layers: this means, in effect that the whole network (except for the final one by one convolutions) is a single dense block. This led to networks which were highly parameter efficient, but which had a very large memory footprint. In the current paper we hybridize this approach with the down/up-sampling approach of U-net [15].

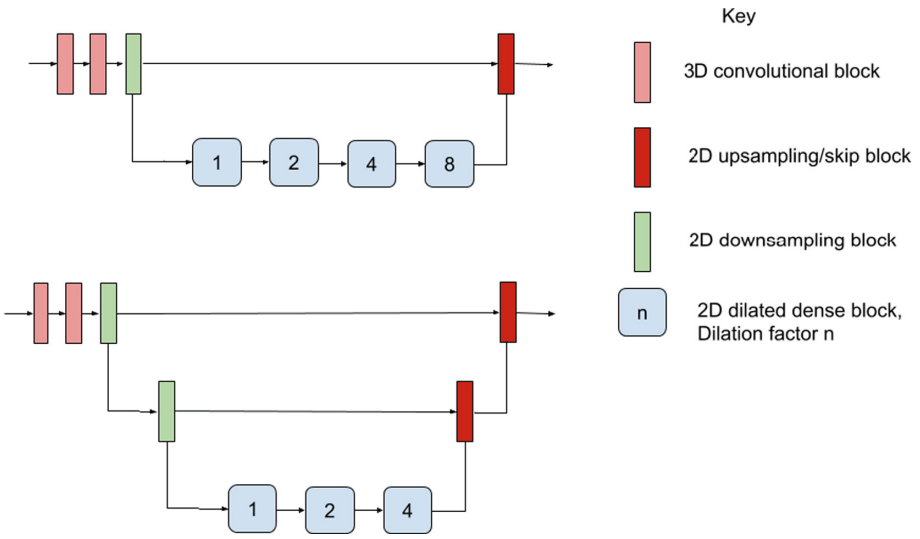

Fig. 2. Two DeepSCAN architectures, as applied to brain tumor segmentation

2.2 Dilated Convolutions

Some kind of pooling is found in almost all CNNs for image classification. The principal reason to use pooling is to efficiently increase the receptive field of the network at deeper levels without exploding the parameter space, but another common justification of pooling, and maxpooling in particular, is that it enables some translation invariance. Translation invariance is of course undesirable in semantic segmentation problems, where what is needed is instead translation *equivariance*: a translated input corresponding to a translated output. To that end, we use layers with dilated convolutions to aggregate features at multiple

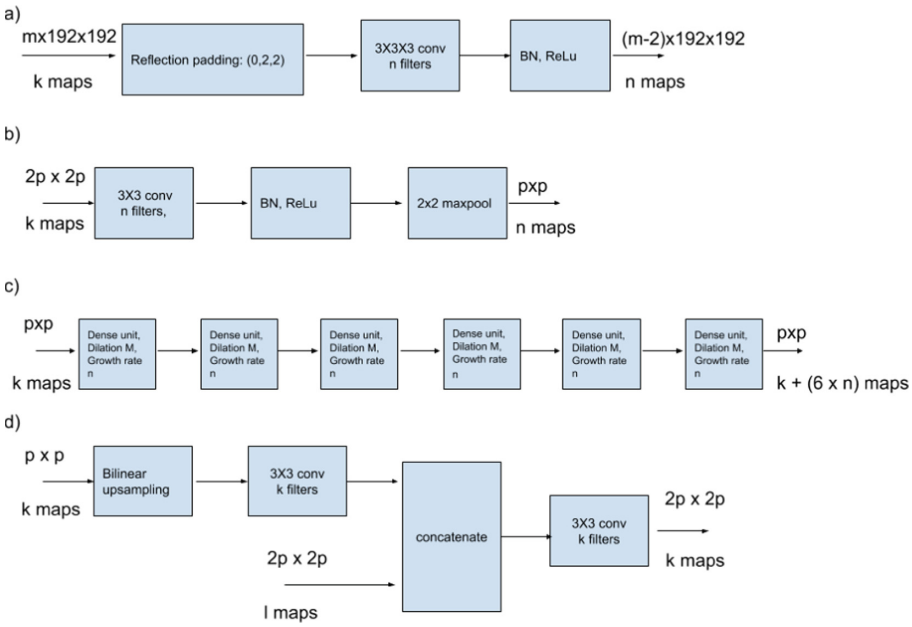

Fig. 3. Units of the DeepSCAN architecture: (a) 3D convolutional blocks, (b) Downsampling block, (c) Dense block, with dilation M, (d) upsampling block. Except in the 3D block, all convolutions are preceded by 2 by 2 reflection padding.

scales. Dilated convolutions, sometimes called atrous convolutions, can be best visualized as convolutional layers "with holes": a 3 by 3 convolutional layer with dilation 2 is a 5 by 5 convolution, in which only the centre and corner values of the filter are nonzero, as illustrated in Fig. 4. Dilated convolutions are a simple way to increase the receptive field of a classifier without losing spatial information.

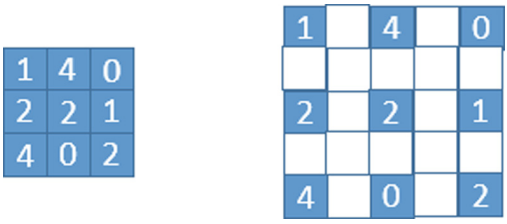

Fig. 4. Left, a 3 by 3 kernel. Right, a 3 by 3 kernel with dilation 2, visualized as a 5 by 5 kernel

2.3 Label-Uncertainty Loss

We introduce a new loss function, which we call label-uncertainty loss, inspired by the recent trend in networks able to quantify their own uncertainty. In brief, for each voxel, and each segmentation sub-task (whole tumor, tumor core, and enhancing tumor) the network outputs two probabilities: the probability p that the label is positive, and the probability q that the label predicted does not correspond to the label in the ground-truth annotation (i.e., the probability of a 'label flip'). IF BCE stands for the standard binary cross-entropy loss, and x is the target label, then the loss function we minimize is:

$$BCE(p, (1-x) * q + x * (1-q)) + BCE(q, z) \tag{1}$$

where

$$z = (p > 0.5) * (1-x) + (p < 0.5) * x \tag{2}$$

If q is close to zero, and the label is correct, the first term is approximately the ordinary BCE loss: if q is close to 0.5 (representing total uncertainty as to the correct label) the first term tends to zero. This loss therefore attenuates loss in areas of high uncertainty, in a similar fashion to the heteroscedastic loss of [10]. However, in [10] the uncertainty in the classification is modeled by assuming that logits have a Gaussian distribution, and estimating the variance of that Gaussian: this cannot be performed directly by gradient descent, instead requiring Monte Carlo sampling of the Gaussian distribution to perturb the output of the network. By contrast, label-uncertainty can be incorporated directly into the loss-function of the network. In fact, the label-uncertainty q can also be viewed as a variance: if we assume that the logit of p follows not a Gaussian but a logistic distribution (as is the standard assumption in classical statistical learning) with mean $\text{logit}(p)$, then if the probability that a sample from that distribution is below zero is q, the variance of the logistic distribution is $\text{abs}(\text{logit}(p)/\text{logit}(q))$.

Since the label-uncertainty loss incorporates the current prediction in evaluating the probability of a label flip, it is important to apply the loss to a network which has already been pre-trained with ordinary BCE loss: for each of our networks we trained to convergence with ordinary BCE loss (typically 10–20 epochs) then switched to using label uncertainty loss. We observed more stability when using both ordinary BCE and label uncertainty. Further, to counter the effects of label imbalance, we adopt the technique of *focal loss* from [11]: therefore, the final loss function used was

$$(1-p_x)^{\gamma}(BCE(p,x) + BCE(p, (1-x) * q + x * (1-q)) + BCE(q,z)) \tag{3}$$

where p_x is p if x is 1 and (1-p) otherwise. For our experiments the value of γ used was 2.

2.4 The DeepSCAN Architecture

The design principles of the DeepSCAN models are (i) non-isotropic input volumes, with one dimension being rather small (in this case, 5 by 192 by 192)

(ii) initial application of enough 3D convolutions to reduce the short dimension to length 1, and (iii) a subsequent hybrid of 2D U-net and 2D Densenet, in which one or steps of convolution and maxpooling are followed by a number of densely connected blocks of dilated convolutions, with the dilation factor increasing with increasing depth, and then finally U-net-style upsampling blocks with skip connections from the previous downward path. The building blocks of these networks are shown in Fig. 3, and two architectures built from these blocks are shown in Fig. 2.

3 Initial Application to Brain Tumor Segmentation

Brain Tumor segmentation has become a benchmark problem in medical image segmentation, due to the existence since 2012 of a long-running competition, BRATS, together with a large curated dataset [1–3,13] of annotated images. Both fully-automated and semi-automatic approaches to brain-tumor segmentation are accepted to the challenge, with supervised learning approaches dominating the fully-automated part of the challenge. A good survey of approaches which dominated BRATS up to 2013 can be found here [5]. More recently, CNN-based approaches have dominated the fully-automated approaches to the problem [6,9,14].

We trained two networks, as pictured in Fig. 2. The networks were built using Pytorch, and trained using the Adam optimizer. Rather than using a softmax layer to classify the three labels (edema, enhancing, other tumor) we employ a multi-task approach to hierarchically segment the tumor into the three overlapping targets: whole tumor, tumor core and enhancing: thus the output of the network is three logits, one for each target. In addition, as per the label uncertainty loss, for each target the network outputs one label-flip logit.

3.1 Data Preparation and Homogenization

The raw values of MRI sequences cannot be compared across scanners and sequences, and therefore a homogenization is necessary across the training examples. In addition, learning in CNNs proceeds best when the inputs are standardized (i.e. mean zero, and unit variance). To this end, the nonzero intensities in the training, validation and testing sets were standardized, this being done across individual volumes rather than across the training set. This achieves both standardization and homogenization.

4 Cascaded Non-brain-tissue Removal

The BRATS dataset was assembled from a large number of data sources, and does not comprise raw imaging data: the volumes are re-sampled to 1 mm isovoxels, and in addition have been automatically skull-stripped. Unfortunately, the results of this skull-stripping vary: see Fig. 5 for an example with large amounts of

residual skull tissue. Other examples have remnants of the dura or optic nerves. This remaining tissue can confound classification in two ways: it can be misidentified by the classification algorithm (though this is increasingly less likely as classifiers improve) and it can affect the distribution of the intensities in a volume, adversely impacting the global standardization of voxel values. To combat this effect, we used a cascade of networks to first segment the parenchymia from the poorly skull-stripped images, followed by a second network which identifies the tumor compartments as above. The ground truth for the brain mask was obtained by applying FSL-FAST to the T1 post Gadolinium imaging, as this tended to have the best definition in all three planes. The brain tissue label was assembled by taking the union of tumor, white matter and grey matter labels, and then taking the largest connected component.

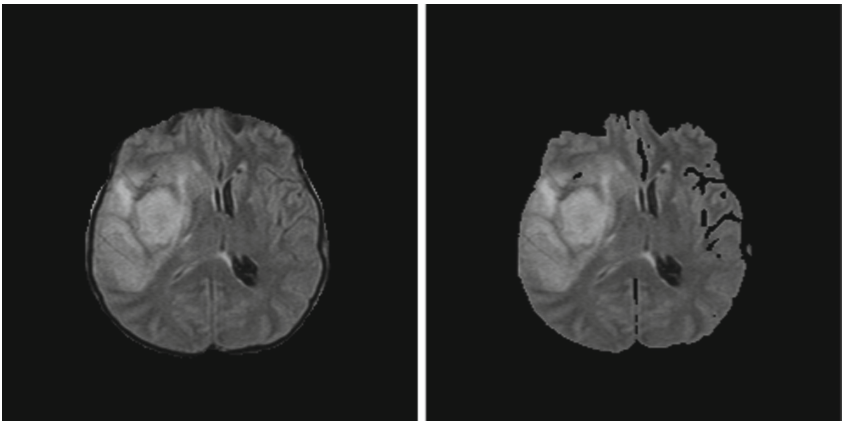

Fig. 5. A FLAIR image from the BRATS2018 testing dataset before (Left) and after (Right) additional brain extraction by our method

This brain-mask tissue label was used during training to ensure the training of networks robust to the presence or absence of non-brain tissue. In addition, we added a brain-mask label to the existing labels in the ground-truth for training, so that during testing a brain-mask for additional skull-stripping could be generated.

4.1 Data Augmentation

During training, we applied the following data augmentation: randomly flipping along the midline, random rotations in a randomly chosen principal axis, and random shifting and scaling of the standardised intensity values. In addition, the classifier was randomly shown either the original images, or images masked with the brain-mask generated as above.

4.2 Training

The network segments the volume slice-by slice: the input data is five consecutive slices from all four modalities, Ground truth for such a set of slices is the lesion mask of the central slice. Input images were initially cropped to remove as much empty space as possible. Batch size during training was 2. As a result, the input tensor to the model has dimensions 2 * 4 * 5 * 192 * 192. Models were trained using a cosine-annealing learning rate schedule, in which the learning rate was varied between 1e−5 and 1e−9 during each epoch.

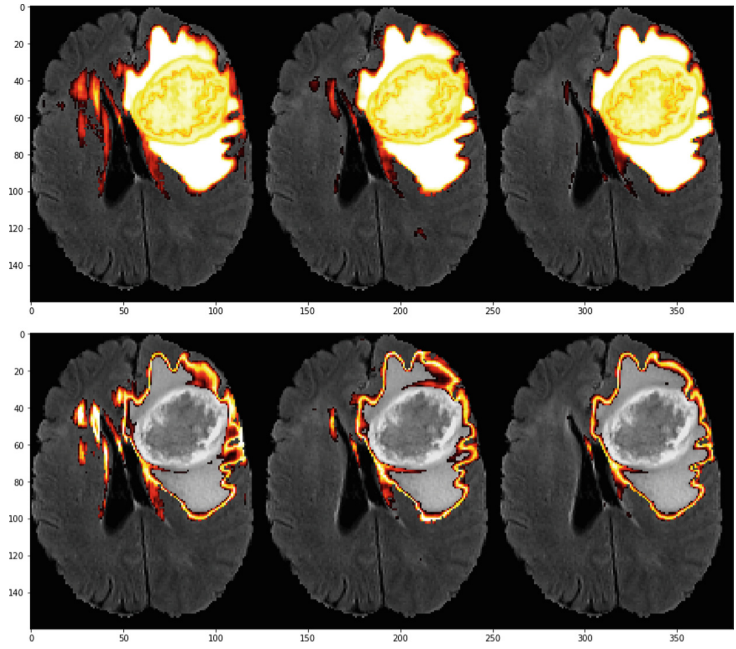

Fig. 6. Above: Whole tumor classified in the (sagittal, coronal, axial) plane. Below: Label-flip probability of the (sagittal, coronal, axial) segmentations

Slices from all three directions (sagittal, axial, coronal) were fed to the classifier for training. Examples of the different segmentations in those three directions (just for the whole tumor label) can be seen in Fig. 6.

4.3 Application of the Classifier

The initial application of the classifier is as follows: the volume is classified in the axial, sagittal and coronal planes separately, by both trained networks. This yields six logit maps, and six label-flip logit maps, for each target label. The logit maps were binarized with a threshold of 0 (corresponding to a standard threshold of 0.5 on the sigmoid of the logit).

Voxel-wise label confidence weights were then derived from the label-flip logits as the minimum of 0 and the negative of the label-flip logit, so that very confident classifications (corresponding to very large negative label-flip logits) contribute more than less-confident classifications. These weights were than used to ensemble the binarized maps.

The brain-mask label from this ensembled classification was then used to mask the input modalities, and the volume was again classified by both networks in all three directions. This yielded another six logit maps (with corresponding label-flip logit maps) for each tissue compartment. The final segmentations for each compartment were produced by the same uncertainty weighted ensembling as above, over all twelve label maps.

5 Results

Results of an ablation study are shown in Table 1, where we show results with and without label-uncertainty-based ensembling and brain extraction. While no single model showed dominance, the model with both novel features achieved the best results in one of Dice or Hausdorff distance for all three compartments, so was selected as the final model. Results on the BRATS 2018 testing data are shown in Table 2: this method gained joint 3rd place in the challenge [4].

Table 1. Results on the BRATS 2018 validation set using the online validation tool. Base denotes the ensemble of two DeepSCAN models over three directions, where ensembling is achieved by averaging logits. "+ U" denotes using averaging over label uncertainty instead of logits. "+ BE" denotes averaging over both original and brain-extracted inputs.

	Dice-ET	Dice-WT	Dice-TC	HD95-ET	HD95-WT	HD95-TC
Base	0.795	0.901	**0.854**	3.61	4.26	5.37
Base + U	0.792	0.901	0.847	3.60	**4.06**	4.99
Base + BE	**0.797**	0.901	0.851	3.60	4.41	5.58
Base + U + BE	0.796	**0.903**	0.847	**3.55**	4.17	**4.93**

Table 2. Results of the ensemble with brain extraction and uncertainty-driven ensembling on the BRATS 2018 testing set

Label	Dice-ET	Dice-WT	Dice-TC	HD95-ET	HD95-WT	HD95-TC
Mean	0.73189	0.88593	0.79926	3.48082	5.5185	5.5347
StdDev	0.27443	0.10182	0.26008	5.52176	9.34294	8.14881
Median	0.83199	0.91786	0.90847	1.73205	3.0	2.82843
25quantile	0.73922	0.87113	0.82327	1.41421	2.23607	1.73205
75quantile	0.88342	0.9396	0.93653	2.82843	5.09902	5.52101

References

1. Bakas, S., et al.: Advancing the cancer genome atlas glioma MRI collections with expert segmentation labels and radiomic features. Nature Sci. Data **4**, 170117 (2017)
2. Bakas, S., et al.: Segmentation labels and radiomic features for the pre-operative scans of the TCGA-GBM collection. In: The Cancer Imaging Archive (2017)
3. Bakas, S., et al.: Segmentation labels and radiomic features for the pre-operative scans of the TCGA-LGG collection. In: The Cancer Imaging Archive (2017)
4. Bakas, S., Reyes, M., Menze, B., et al.: Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the brats challenge. 1811.02629 (2015). https://arxiv.org/abs/1811.02629
5. Bauer, S., Wiest, R., Nolte, L.L., Reyes, M.: A survey of mri-based medical image analysis for brain tumor studies. Phys. Med. Biol. **58**(13), R97–129 (2013)
6. Havaei, M., et al.: Brain tumor segmentation with deep neural networks. Med. Image Anal. **35**, 18–31 (2017)
7. Huang, G., Liu, Z., van der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2017)
8. Jégou, S., Drozdzal, M., Vázquez, D., Romero, A., Bengio, Y.: The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation. vol. abs/1611.09326 (2016). http://arxiv.org/abs/1611.09326
9. Kamnitsas, K., et al.: Efficient multi-scale 3D CNN with fully connected crf for accurate brain lesion segmentation. Med. Image Anal. **36**, 61–78 (2017)
10. Kendall, A., Gal, Y.: What uncertainties do we need in bayesian deep learning for computer vision? In: NIPS (2017)
11. Lin, T.Y., Goyal, P., Girshick, R.B., He, K., Dollár, P.: Focal loss for dense object detection. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 2999–3007 (2017)
12. McKinley, R., Jungo, A., Wiest, R., Reyes, M.: Pooling-free fully convolutional networks with dense skip connections for semantic segmentation, with application to brain tumor segmentation. In: Crimi, A., Bakas, S., Kuijf, H., Menze, B., Reyes, M. (eds.) BrainLes 2017. LNCS, vol. 10670, pp. 169–177. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-75238-9_15
13. Menze, B.H., et al.: The multimodal brain tumor image segmentation benchmark (BRATS). IEEE Trans. Med. Imaging **34**(10), 1993–2024 (2015)
14. Pereira, S., Pinto, A., Alves, V., Silva, C.A.: Brain tumor segmentation using convolutional neural networks in MRI images. IEEE Trans. Med. Imaging **35**(5), 1240–1251 (2016)
15. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
16. Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. In: Proceedings of International Conference on Learning Representations (ICLR 2017) (2017)