Multi-pass 3D convolutional neural network segmentation of prostate MRI images

Bruno Sciolla*[1], Matthieu Martin[1], and Philippe Delachartre[1]

[1]CREATIS, INSA Lyon

August 7, 2017

Abstract

We propose a deep neural network for the segmentation of the prostate in MRI images. The segmentation is performed using a residual fully convolutional neural network. Automatic shape learning is allowed using a Compositional Pattern-Producing Network. Moreover, a multi-pass architecture is designed to foster self-consistent segmentation. The model is trained and tested on the dataset of the challenge PROMISE12.

1 Introduction

In the following, we introduce the deep neural network architecture for the segmentation of the prostate in 3D images of T2-axial MRI for the PROMISE12 challenge [1].

The following is a concise and dry description of the algorithm used to obtain the results in the challenge. The methodological contributions proposed here are not justified in careful detail nor compared against previous methods. Extensive comparisons and explanations will be given in future publications.

The proposed method has the following features:

- An integrated multi-pass (multi-hierarchical) residual architecture
- A series of 2D and 1D convolutions along the axial direction are performed. This takes into account the anisotropic nature of the images yet giving access to 3D structure of the data.
- Shape learning with shape generation via CPPN
- Multiplicative weight-enhanced ELU activations
- Extensive data augmentation

Fully convolutional methods have been introduced recently as a way to provide dense output from images [2, 3, 4, 5, 6]. It is still an open question, to decide how to properly include contextual information in CNNs. Some proposals include using Conditional Random Markov Fields [7], or multiscale methods [8, 3, 4, 9, 10].

2 Methods

2.1 Building blocs of the neural network

The neural network takes as input a 3D image $I(x, y, z, \alpha)$ with $x \in [\![1, N_x]\!]$, $y \in [\![1, N_y]\!]$, $z \in [\![1, N_z]\!]$ with one input channel $\alpha = 1$, namely the T2 axial MRI volume. x is the frontal axis, y the transverse axis, z the longitudinal axis.

*Email: bruno.sciolla@insa-lyon.fr

In a standard convolutional network with ELU activations [11], the convolution and activation is defined as:

$$C^{\beta,\alpha}_{l\times m\times n,r}[X] = \text{ELU}\left(\sum_{\alpha} W^{\alpha,\beta}_{(x,y,z)} * X^{\alpha}_{(x,y,z)} + b^{\beta}\right)$$

where β (α) are the number of channels after (before) convolution. Convolution kernels $W^{\alpha,\beta}_{(x,y,z)}$ are of size $l\times m\times n$ and biases b^{β}, the activation function is an exponential-linear unit ELU. $l\times m\times n$ is the spatial extent of each convolution kernel $W_{x,y,z}$. The convolutions are dilated/atrous convolutions [12] of rate r, where $r-1$ is the number of zeros added between values in the three axis x, y, z. The rate (or scale) of the dilated/atrous convolution [12] is r, where $r-1$ is the number of zeros added between values in the three axis x, y, z. For an original filter of size $3\times 3\times 1$, the dilated/atrous convolution of rate r is of size $(1+2r)\times(1+2r)\times 1$.

We use augmented convolutions defined as:

$$C^{\beta,\alpha}_{l\times m\times n,r}[X] = \omega_G\ \text{ELU}\left(\omega^{\beta}_{C}\left[\sum_{\alpha}\omega^{\alpha,\beta}_{F} W^{\alpha,\beta}_{x,y,z} * X^{\alpha}_{x,y,z} + b^{\beta}\right]\right) \tag{1}$$

This design is a variant of weight renormalization [13], in which the amplitude of filters is decoupled. In contrast to [13], the kernels are not normalized. The multiplicative factors are defined at different degrees of granularity:

- ω_G is the global (G) multiplicative factor
- ω^{β}_{C} is the channel-wise (C) (β) multiplicative factor
- $\omega^{\alpha,\beta}_{F}$ is the filter-wise (F) (α,β) multiplicative factor

2.2 Multi-pass convolutional neural network

The architecture of the proposed neural network is depicted in Fig. 1. The neural network starts with three 2D convolutional layers in the (x, y) plane:

$$X(x, y, z, \beta) = C^{64,32}_{3\times3\times1,r=2} \circ C^{32,16}_{3\times3\times1,r=2} \circ C^{16,4}_{3\times3\times1,r=2}(I) \tag{2}$$

where $\circ$ is the composition. The 2D convolution is equivalent to a 3D convolution with size $5\times5\times1$ (notice the rate $r=2$).

The resulting feature map is concatenated with the output from the CPPN network, defined in the following.

There is then a series of residual layers, of two types:

- 2D residual bloc:

$$\text{Res\_2D}_{(r)}(X) = X + C^{64,32}_{3\times3\times1,r} \circ C^{32,64}_{3\times3\times1,r}(X) \tag{3}$$

  with the rate parameter $r=\{2,4,8,16\}$, for a 3D receptive field of size $(2r+1)\times(2r+1)\times1$.

- "3D" Residual bloc:

$$\text{Res\_3D}(X) = X + C^{64,64}_{1\times1\times3,r=1}(X) \tag{4}$$

  with a vector of size 3 along axis z and no dilation $r=1$. We refer to this bloc as a 3D layer because it captures image features and labelling consistency along the z axis.

Residual blocs are stacked with growing rate $r=\{2,4,8,16\}$ which gives access to features on a larger scale for deeper layers. The multi-pass architecture consists in doing a second pass from lower scale $r=\{2,8,16\}$.

The final prediction layer is the standard pixelwise labelling $i\in[\![1,L]\!]$ ($L=2$ is the number of labels):

$$L(x, y, z, i) = \text{Pred}[X] = S \circ C^{L,256}_{1\times1} \circ D_p \circ C^{256,64}_{1\times1}(X) \tag{5}$$

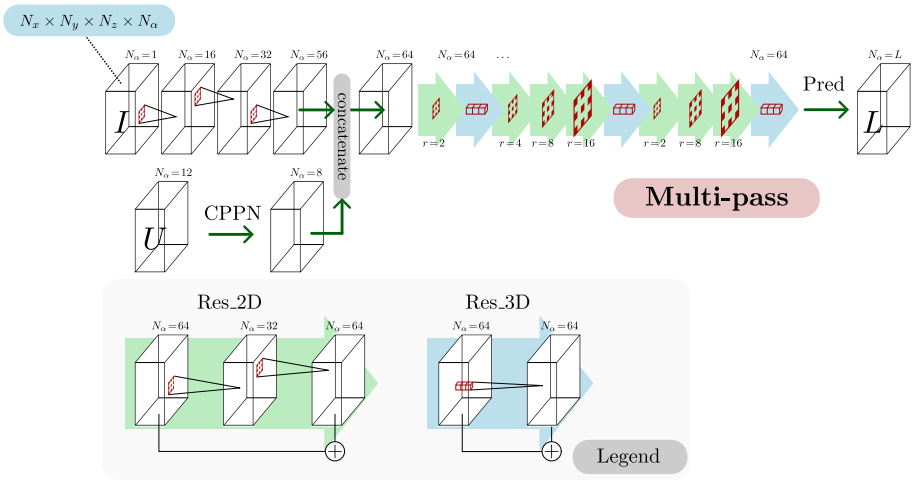

Figure 1: Processing chain. Since there is no downsampling, each volume has the same spatial size and a number of channels indicated by N_α. The arrows represent the action of a Residual bloc featuring 2D convolutions Res_2D or a one dimensional convolution along the axial direction Res_3D. All transformations depicted in this picture are found in equations (3), (4) and (5) in the text.

with D_p *dropout* with keeping rate p, S is a softmax layer:

$$S(X(x,y,z,i)) = \exp(X(x,y,z,i))/\sum_{i\in[\![1,L]\!]} \exp(X(x,y,z,i)) \tag{6}$$

After the final inference layer, a simple postprocessing is added at test time: the largest single component in 3D is kept, followed by slight smoothing using an anisotropic gaussian 3D filter and fixed threshold.

2.3 CPPN shape learning

The neural network is able to learn spatial priors using a Compositional Pattern-Producing Network (CPPN) [14] which takes as input normalized coordinates X, Y, Z and generates as output 8 learned maps, which are then injected as extra features for the residual network to process.

Specifically, the CPPN network that we use takes as input the three normalized coordinate maps $U(x,y,z,\alpha)$ with X, Y, Z in the components $\alpha = \{1,2,3\}$ respectively, and some non-linear functions X^2, Y^2, Z^2, XY, YZ, ZX, $\sin(2\pi X + \pi/2)$, $\sin(2\pi Y + \pi/2)$, $\sin(2\pi Z + \pi/2)$ in channels $\alpha = \{4,\ldots,12\}$. The coordinates maps are then processed pixelwise (with trivially local $1\times1\times1$ convolutions)

$$O(x,y,z,\beta) = C^{8,16}_{1\times1\times1} \circ C^{16,16}_{1\times1\times1} \circ C^{16,12}_{1\times1\times1}(U) \tag{7}$$

which generates 8 learned geometrical feature maps.

The features maps are concatenated with low-level computed features in the convolutional network as depicted in Fig. 1 with the symbol "concatenate".

2.4 Training

The model is trained end-to-end for 150k steps. For the first 5k steps, the cost function is the cross-entropy, afterwards the cost function is the Dice coefficient [15] for the rest of the training. The gradient descent is performed with ADAM optimizer, with parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$, and learning rate $\Delta = 10^{-4}$. The keeping rate in the Dropout layer is of $p = 0.7$.

The model is trained with batches of 2 cases, with a sub-region of the original volume of size $96 \times 96 \times 9$ pixels. The batches are chosen at random, with the constraint that at least 90% of the windows of the batches must contain a portion of the prostate. In order to avoid boundary effects, the cost function was only computed in a window of 10 pixels away from the boundaries in the x, y plane during the training phase.

During training and testing, all volumes are interpolated to size $320 \times 320 \times 54$ and the resulting segmentation is interpolated back to the original image size using trilinear interpolation on the signed distance transform of the segmentation mask.

During the training phase, the data is augmented using several transformations: flip along the middle sagittal plane (the only physiological symmetry of the prostate), random rotations, affine transformations and non-linear multiscale deformation using scale-free gaussian random fields. The last transformations are realized in the x, y plane identically along the z axis. Also, random shifting of the initial coordinate system is added.

3 Results

The dataset of 50 cases is split into 4 validation cases and 46 training cases. The Dice index over the entire volume for the training and the validation set is reported in Table 1.

Table 1:

	Dice (mean)	Dice (median)	Dice (std)
Validation	0.89	0.89	0.02
Training	0.954	0.955	0.007

Acknowledgment

This work was funded by the ANR-14-LAB3-0006-01 LabCom AtysCrea and was supported by the LABEX CeLyA (ANR-10-LABX-0060) of Université de Lyon, within the "Investissements d'Avenir" program (ANR-11-IDEX-0007) operated by the French National Research Agency (ANR).

References

[1] G. Litjens, R. Toth, W. van de Ven, C. Hoeks, S. Kerkstra, B. van Ginneken, G. Vincent, G. Guillard, N. Birbeck, J. Zhang, R. Strand, F. Malmberg, Y. Ou, C. Davatzikos, M. Kirschner, F. Jung, J. Yuan, W. Qiu, Q. Gao, P. E. Edwards, B. Maan, F. van der Heijden, S. Ghose, J. Mitra, J. Dowling, D. Barratt, H. Huisman, and A. Madabhushi, "Evaluation of prostate segmentation algorithms for mri: The promise12 challenge," *Medical Image Analysis*, vol. 18, no. 2, pp. 359 – 373, 2014.

[2] S. S. Kruthiventi, K. Ayush, and R. V. Babu, "Deepfix: A fully convolutional neural network for predicting human eye fixations," in *arXiv:1510.02927*, 2015.

[3] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2650–2658.

[4] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *ICLR 2016*. arXiv preprint arXiv:1511.07122, 2015.

[5] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2015, pp. 234–241.

[6] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[7] G. Lin, C. Shen, I. Reid *et al.*, "Efficient piecewise training of deep structured models for semantic segmentation," *arXiv preprint arXiv:1504.01013*, 2015.

[8] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Hypercolumns for object segmentation and fine-grained localization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 447–456.

[9] P. Moeskops, M. A. Viergever, A. M. Mendrik, L. S. de Vries, M. J. Benders, and I. Išgum, "Automatic segmentation of mr brain images with a convolutional neural network," *IEEE transactions on medical imaging*, vol. 35, no. 5, pp. 1252–1261, 2016.

[10] T. Brosch, L. Y. Tang, Y. Yoo, D. K. Li, A. Traboulsee, and R. Tam, "Deep 3D convolutional encoder networks with shortcuts for multiscale feature integration applied to multiple sclerosis lesion segmentation," *IEEE transactions on medical imaging*, vol. 35, no. 5, pp. 1229–1239, 2016.

[11] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," *arXiv preprint arXiv:1511.07289*, 2015.

[12] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, no. 99, pp. 1–1, 2017.

[13] T. Salimans and D. P. Kingma, "Weight normalization: A simple reparameterization to accelerate training of deep neural networks," in *Advances in Neural Information Processing Systems*, 2016, pp. 901–909.

[14] K. O. Stanley, "Compositional pattern producing networks: A novel abstraction of development," *Genetic programming and evolvable machines*, vol. 8, no. 2, pp. 131–162, 2007.

[15] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *3D Vision (3DV), 2016 Fourth International Conference on*. IEEE, 2016, pp. 565–571.