3D U-net with Multi-level Deep Supervision: Fully Automatic Segmentation of Proximal Femur in 3D MR Images

Guodong Zeng[1], Xin Yang[2], Jing Li[1], Lequan Yu[2], Pheng-Ann Heng[2], and Guoyan Zheng[1](✉)

[1] Institute for Surgical Technology and Biomechanics,
University of Bern, Bern, Switzerland
guoyan.Zheng@istb.unibe.ch

[2] Department of Computer Science and Engineering,
The Chinese University of Hong Kong, Sha Tin, Hong Kong

Abstract. This paper addresses the problem of segmentation of proximal femur in 3D MR images. We propose a deeply supervised 3D U-net-like fully convolutional network for segmentation of proximal femur in 3D MR images. After training, our network can directly map a whole volumetric data to its volume-wise labels. Inspired by previous work, multi-level deep supervision is designed to alleviate the potential gradient vanishing problem during training. It is also used together with partial transfer learning to boost the training efficiency when only small set of labeled training data are available. The present method was validated on 20 3D MR images of femoroacetabular impingement patients. The experimental results demonstrate the efficacy of the present method.

Keywords: Deep learning · Proximal femur · MR images · Segmentation

1 Introduction

Femoroacetabular Impingement (FAI) is a cause of hip pain in adults and has been recognized recently as one of the key risk factors that may lead to the development of early cartilage and labral damage [1] and a possible precursor of hip osteoarthritis [2]. Several studies [2,3] have shown that the prevalence of FAI in young populations with hip complaints is high. Although there exist a number of imaging modalities that can be used to diagnose and assess FAI, MR imaging does not induce any dosage of radiation at all and is regarded as the standard tool for FAI diagnosis [4]. While manual analysis of a series of 2D MR images is feasible, automated segmentation of proximal femur in MR images will greatly facilitate the applications of MR images for FAI surgical planning and simulation.

The topic of automated MR image segmentation of the hip joint has been addressed by a few studies which relied on atlas-based segmentation [5], graph-cut [6], active model [7,8] or statistical shape models [9]. While these methods

Q. Wang et al. (Eds.): MLMI 2017, LNCS 10541, pp. 274–282, 2017.
DOI: 10.1007/978-3-319-67389-9_32

reported encouraging results for bone segmentation, further improvements are needed. For example, Arezoomand et al. [8] recently developed a 3D active model framework for segmentation of proximal femur in MR images and they reported an average recall of 0.88.

Recently, machine-learning based methods, especially those based on convolutional neural networks (CNNs) have witnessed successful applications in natural image processing [10,11] as well as in medical image analysis [12–15]. For example, Prasoon et al. [12] developed a method to use a triplanar CNN that can autonomously learn features from images for knee cartilage segmentation. More recently, 3D volume-to-volume segmentation networks were introduced, including 3D U-Net [13], 3D V-Net [14] and a 3D deeply supervised network [15].

In this paper, we propose a deeply supervised 3D U-net-like fully convolutional network (FCN) for segmentation of proximal femur in 3D MR images. After training, our network can directly map a whole volumetric data to its volume-wise label. Inspired by previous work [13,15], multi-level deep supervision is designed to alleviate the potential gradient vanishing problem during training. It is also used together with partial transfer learning to boost the training efficiency when only small set of labeled training data are available.

2 Method

Figure 1 illustrates the architecture of our proposed deeply-supervised 3D U-net-like network. Our proposed neural network is inspired by the 3D U-net [13]. Similar to 3D U-net, our network also consists of two parts, i.e., the encoder part(contracting path) and the decoder part(expansive path). The encoder part focuses on analysis and feature representation learning from the input data while the decoder part generates segmentation results, relying on the learned features

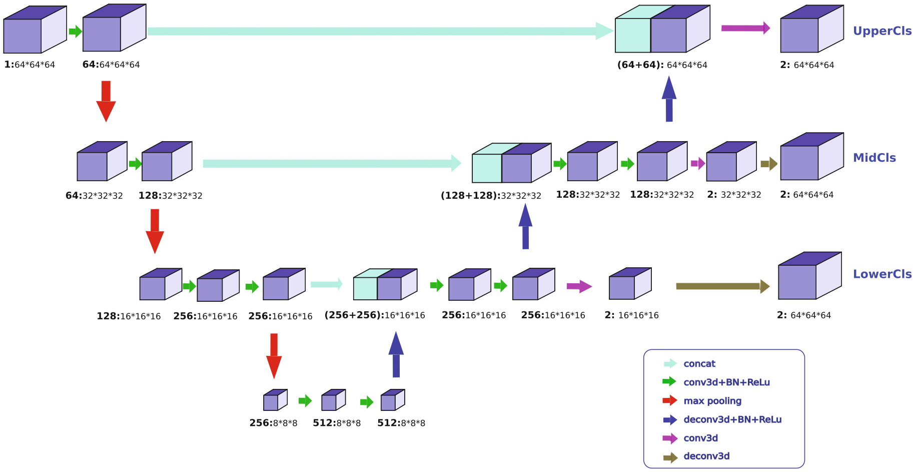

Fig. 1. Illustration of our proposed network architecture

from the encoder part. Shortcut connections are established between layers of equal resolution in the encoder and decoder paths. The difference between our network and the 3D U-net is the introduction of multi-level deep supervision, which gives more feedback to help training during back propagation process.

Previous studies show small convolutional kernels are more beneficial for training and performance. In our deeply supervised network, all convolutional layers use kernel size of $3 \times 3 \times 3$ and strides of 1 and all max pooling layers uses kernel size of $2 \times 2 \times 2$ and strides of 2. In the convolutional and deconvolutional blocks of our network, Batch normalization (BN) [16] and Rectified linear unit (ReLU) are adopted to speed up the training and to enhance the gradient back propagation.

2.1 Multi-level Deep Supervision

Training a deep neural network is challenging. As the matter of gradient vanishing, final loss cannot be efficiently back propagated to shallow layers, which is more difficult for 3D cases when only a small set of annotated data is available. To address this issue, we inject two branch classifiers into network in addition to the classifier of the main network. Specifically, we divide the decoder path of our network into three different levels: lower layers, middle layers and upper layers. Deconvolutional blocks are injected into lower and middle layers such that the low-level and middle-level features are upscaled to generate segmentation predictions with the same resolution as the input data. As a result, besides the classifier from the upper final layer ('UpperCls' in Fig. 1), we also have two branch classifiers in lower and middle layers ('LowerCls' and 'MidCls' in Fig. 1, respectively). With the losses calculated by the predictions from classifiers of different layers, more effective gradients back propagation can be achieved by direct supervision on the hidden layers.

Let W be the weights of main network and w^l, w^m, w^u be the weights of the three classifiers 'LowerCls', 'MidCls' and 'UpperCls', respectively. Then the cross-entropy loss function of a classifier is:

$$\mathcal{L}_c(\chi; W, w^c) = \sum_{x_i \in \chi} -\log p(y_i = t(x_i) | x_i; W, w^c)) \tag{1}$$

where $c \in \{l, m, u\}$ represents the index of the classifiers; χ represents the training samples; $p(y_i = t(x_i)|x_i; W, w^c)$ is the probability of target class label $t(x_i)$ corresponding to sample $x_i \in \chi$.

The total loss function of our deep-supervised 3D network is:

$$\mathcal{L}(\chi; W, w^l, w^m, w^u) = \sum_{c \in \{l,m,u\}} \alpha_c \mathcal{L}_c(\chi; W, w^c) + \lambda(\psi(W) + \sum_{c \in \{l,m,u\}} \psi(w^c)) \tag{2}$$

where $\psi()$ is the regularization term (L_2 norm in our experiment) with hyper parameter λ; $\alpha_l, \alpha_m, \alpha_u$ are the weights of the associated classifiers.

By doing this, classifiers in different layers can also take advantages of multi-scale context, which has been demonstrated in previous work on segmentation of

3D liver CT and 3D heart MR images [15]. This is based on the observation that lower layers have smaller receptive fields while upper layers have larger receptive fields. As a result, multi-scale context information can be learned by our network which will then facilitate the target segmentation in the test stage.

2.2 Partial Transfer Learning

It is difficult to train a deep neural network from scratch because of limited annotated data. Training deep neural network requires large amount of annotated data, which are not always available, although data augmentation can partially address the problem. Furthermore, randomly initialized parameters make it more difficult to search for an optimal solution in high dimensional space. Transfer learning from an existing network, which has been trained on a large set of data, is a common way to alleviate the difficulty. Usually the new dataset should be similar or related to the dataset and tasks used in the pre-training stage. But for medical image applications, it is difficult to find an off-the-shelf 3D model trained on a large set of related data of related tasks.

Previous studies [17] demonstrated that weights of lower layers in deep neural network is generic while higher layers are more related to specific tasks. Thus, the encoder path of our neural network can be transferred from models pre-trained on a totally different dataset. In the field of computer vision, lots of models are trained on very large dataset, e.g., ImageNet [18], VGG16 [19], Googlenet [20], etc. Unfortunately, most of these models were trained on 2D images. 3D pre-trained models that can be freely accessed are rare in both computer vision and medical image analysis fields.

C3D [21] is one of the few 3D models that has been trained on a very large dataset in the field of computer vision. More specifically, C3D is trained on the Sports-1M dataset to learn spatiotemporal features for action recognition. The Sports-1M dataset consists of 1.1 million sports videos, and each video belongs to one of 487 sports categories.

In our experiment, C3D pre-trained model was adopted as the pre-trained model for the encoder part of our neural network. For the decoder parts of our neural network, they were randomly initialized.

2.3 Implementation Details

The proposed network was implemented in python using TensorFlow framework and trained on a desktop with a 3.6 GHz Intel(R) i7 CPU and a GTX 1080 Ti graphics card with 11 GB GPU memory. The source code is publicly available at github[1].

[1] https://github.com/zengguodong/FemurSegmentation3DFCN.

3 Experiments and Results

3.1 Dataset and Pre-processing

We evaluated our method on a set of unilateral hip joint data containing 20 T1-weighted MR images of FAI patients. We randomly split the dataset into two parts, ten images are for training and the other ten images are for testing. Data augmentation was used to enlarge the training samples by rotating each image (90, 180, 270) degrees around the z axis of the image and flipped horizontally (y axis). After that, we got in total 80 images for training.

3.2 Training Patches Preparation

All sub-volume patches to our neural network are in the size of $64 \times 64 \times 64$. We randomly cropped sub-volume patches from training samples whose size are about $300 \times 200 \times 100$. In the phase of training, during every epoch, 80 training volumetric images were randomly shuffled. We then randomly sampled patches with batch size 2 from each volumetric image for n times ($n = 5$). Each sampled patch was normalized as zero mean and unit variance before fed into network.

3.3 Training

We trained two different models, one with partial transfer learning and the other without. More specifically, to train the model with partial transfer learning, we initialized the weights of the encoder part of the network from the pre-trained C3D [21] model and the weights of other parts from a Gaussian distribution ($\mu = 0, \sigma = 0.01$). In contrast, for the model without partial transfer learning, all weights were initialized from Gaussian distribution ($\mu = 0, \sigma = 0.01$).

Each time, the model was trained for 14,000 iterations and the weights were updated by the stochastic gradient descent (SGD) algorithm (momentum = 0.9, weight decay = 0.005). The initial learning rate was 1×10^{-3} and halved by 3000 every training iterations. The hyper parameters were chosen as follows: $\lambda = 0.005$, $\alpha_l = 0.33$, $\alpha_m = 0.67$, and $\alpha_u = 1.0$.

3.4 Test and Evaluation

Our trained models can estimate labels of an arbitrary-sized volumetric image. Given a test volumetric image, we extracted overlapped sub-volume patches with the size of $64 \times 64 \times 64$, and fed them to the trained network to get prediction probability maps. For the overlapped voxels, the final probability maps would be the average of the probability maps of the overlapped patches, which were then used to derive the final segmentation results. After that, we conducted morphological operations to remove isolated small volumes and internal holes as there is only one femur in each test data. When implemented with Python using TensorFlow framework, our network took about 2 min to process one volume with size of $300 \times 200 \times 100$.

The segmented results were compared with the associated ground truth segmentation which was obtained via a semi-automatic segmentation using the commercial software package called Amira[2]. Amira was also used to extract surface models from the automatic segmentation results and the ground truth segmentation. For each test image, we then evaluated the distance between the surface models extracted from different segmentation as well as the volume overlap measurements including DICE overlap coefficient [22], Jaccard coefficient [22], precision and recall.

Table 1. Quantitative evaluation results on testing datasets

ID	Surface distance (mm)			Volume overlap measurement			
	Mean	STD	Hausdorff distance	DICE	Jaccard	Precision	Recall
Pat01	0.17	0.31	3.8	0.989	0.978	0.992	0.985
Pat02	0.27	0.46	5.3	0.986	0.973	0.985	0.987
Pat03	0.19	0.35	4.1	0.987	0.975	0.995	0.979
Pat04	0.23	0.67	13.0	0.987	0.974	0.992	0.982
Pat05	0.12	0.21	4.3	0.989	0.979	0.991	0.988
Pat06	0.14	0.26	4.5	0.990	0.980	0.995	0.985
Pat07	0.41	0.95	7.0	0.978	0.958	0.984	0.973
Pat08	0.39	0.93	5.2	0.981	0.963	0.994	0.968
Pat09	0.12	0.17	11.0	0.990	0.981	0.990	0.990
Pat10	0.15	0.28	5.3	0.988	0.976	0.991	0.984
Average	0.22	–	6.4	0.987	0.974	0.991	0.982

3.5 Results

Table 1 shows the segmentation results using the model trained with partial transfer learning. In comparison with manually annotated ground truth data, our model achieved an average surface distance of 0.22 mm, an average DICE coefficient of 0.987, an average Jaccard index of 0.974, an average precision of 0.991 and an average recall of 0.982. Figure 2 shows a segmentation example and the color-coded error distribution of the segmented surface model.

We also compared the results achieved by using the model with partial transfer learning with the one without partial transfer learning. The results are presented in Table 2, which clearly demonstrate the effectiveness of the partial transfer learning.

[2] http://www.amira.com/.

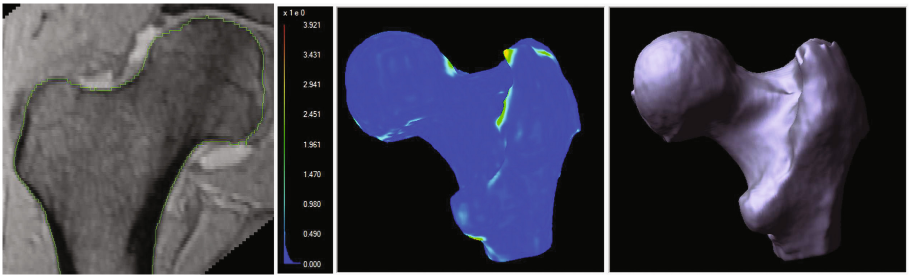

Fig. 2. A segmentation example (left) and the color-coded error distribution of the surface errors (right).

Table 2. Comparison of the average results of the proposed network on the same test dataset when trained with and without transfer learning

Learning method	Surface distance (mm)			Volume overlap measurement			
	Mean	STD	Hausdorff distance	DICE	Jaccard	Precision	Recall
Without transfer learning	0.67	–	12.4	0.975	0.950	0.985	0.964
With transfer learning	0.22	–	6.4	0.987	0.974	0.991	0.982

4 Conclusion

We have introduced a 3D U-net-like fully convolutional network with multi-level deep supervision and successfully applied it to the challenging task of automatic segmentation of proximal femur in MR images. Multi-level deep supervision and partial transfer learning were used in our network to boost the training efficiency when only small set of labeled 3D training data were available. The experimental results demonstrated the efficacy of the proposed network.

Acknowledgment. This study was partially supported by the Swiss National Science Foundation via project 205321_163224/1.

References

1. Laborie, L., Lehmann, T., Engesæter, I., et al.: Prevalence of radiographic findings thought to be associated with femoroacetabular impingement in a population-based cohort of 2081 healthy young adults. Radiology **260**, 494–502 (2011)
2. Leunig, M., Beaulé, P., Ganz, R.: The concept of femoroacetabular impingement: current status and future perspectives. Clin. Orthop. Relat. Res. **467**, 616–622 (2009)
3. Clohisy, J., Knaus, E., Hunt, D.M., et al.: Clinical presentation of patients with symptomatic anterior hip impingement. Clin. Orthop. Relat. Res. **467**, 638–644 (2009)

4. Perdikakis, E., Karachalios, T., Katonis, P., Karantanas, A.: Comparison of MR-arthrography and MDCT-arthrography for detection of labral and articular cartilage hip pathology. Skeletal Radiol. **40**, 1441–1447 (2011)
5. Xia, Y., Fripp, J., Chandra, S., Schwarz, R., Engstrom, C., Crozier, S.: Automated bone segmentation from large field of view 3D MR images of the hip joint. Phys. Med. Biol. **21**, 7375–7390 (2013)
6. Xia, Y., Chandra, S., Engstrom, C., Strudwick, M., Crozier, S., Fripp, J.: Automatic hip cartilage segmentation from 3D MR images using arc-weighted graph searching. Phys. Med. Biol. **59**, 7245–66 (2014)
7. Gilles, B., Magnenat-Thalmann, N.: Musculoskeletal MRI segmentation using multi-resolution simplex meshes with medial representations. Med. Image Anal. **14**, 291–302 (2010)
8. Arezoomand, S., Lee, W.S., Rakhra, K., Beaule, P.: A 3D active model framework for segmentation of proximal femur in MR images. Int. J. CARS **10**, 55–66 (2015)
9. Chandra, S., Xia, Y., Engstrom, C., et al.: Focused shape models for hip joint segmentation in 3D magnetic resonance images. Med. Image Anal. **18**, 567–578 (2014)
10. Krizhevsky, A., Sutskever, I., Hinton, G.: Imagenet classification with deep convolutional neural networks. In: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 25, pp. 1097–1105. Curran Associates, Inc. (2012)
11. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3431–3440 (2015)
12. Prasoon, A., Petersen, K., Igel, C., Lauze, F., Dam, E., Nielsen, M.: Deep Feature Learning for Knee Cartilage Segmentation Using a Triplanar Convolutional Neural Network. In: Mori, K., Sakuma, I., Sato, Y., Barillot, C., Navab, N. (eds.) MICCAI 2013. LNCS, vol. 8150, pp. 246–253. Springer, Heidelberg (2013). doi:10.1007/978-3-642-40763-5_31
13. Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O.: 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation. In: Ourselin, S., Joskowicz, L., Sabuncu, M.R., Unal, G., Wells, W. (eds.) MICCAI 2016. LNCS, vol. 9901, pp. 424–432. Springer, Cham (2016). doi:10.1007/978-3-319-46723-8_49
14. Milletari, F., Navab, N., Ahmadi, S.A.: V-net: fully convolutional neural networks for volumetric medical image segmentation. In: Proceedings of 2016 International Conferece on 3D Vision (3DV), pp. 565–571. IEEE (2016)
15. Dou, Q., Yu, L., Chen, H., Jin, Y., Yang, X., Qin, J., Heng, P.A.: 3D deeply supervised network for automated segmentation of volumetric medical images. Med. Image Anal. **41**, 40–54 (2017)
16. Ioffe, S., Szegedy, C.: Batch normalization: accelerating deep network training by reducing internal covariate shift. In: Proceedings of ICML (2015)
17. Yosinski, J., Clune, J., Bengio, Y., Lipson, H.: How transferable are features in deep neural networks? In: Advances in Neural Information Processing Systems, pp. 3320–3328 (2014)
18. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: a large-scale hierarchical image database. In: CVPR 2009 (2009)
19. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. CoRR (2014)
20. Szegedy, C., Liu, W., Jia, Y., et al.: Going deeper with convolutions. In: CVPR 2015, pp. 1–9. IEEE (2015)

21. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3D convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision (CVPR), pp. 4489–4497 (2015)
22. Karasawa, K., Oda, M., Kitasakab, T., et al.: Multi-atlas pancreas segmentation: Atlas selection based on vessel structure. Med. Image Anal. **39**, 18–28 (2017)