Brain Tissue Segmentation Using NeuroNet With Different Pre-processing Techniques

1st Fakrul Islam Tushar
Erasmus+ Joint Master In Medical Imaging and Applications
University of Girona
Girona, Spain
f.i.tushar.eee@gmail.com

2nd Basel Alyafi
Erasmus+ Joint Master In Medical Imaging and Applications
University of Girona
Girona, Spain
u1951852@campus.udg.edu

3rd Md. Kamrul Hasan
Erasmus+ Joint Master In Medical Imaging and Applications
University of Girona
Girona, Spain
kamruleeekuet@gmail.com

4th Lavsen Dahal
Erasmus+ Joint Master In Medical Imaging and Applications
University of Girona
Girona, Spain
er.lavsen@gmail.com

***Abstract*—Automatic segmentation of MRI brain images is one of the vital steps for quantitative analysis of brain for further inspection. In this paper, NeuroNet has been adopted to segment the brain tissues (white matter (WM), gray matter (GM) and cerebrospinal fluid (CSF)) which uses Residual Network (ResNet) in encoder and Fully Convolution Network (FCN) in the decoder. To achieve the best performance, various hyper-parameters have been tuned, while, network parameters (kernel and bias) were initialized using the NeuroNet pre-trained model. Different pre-processing pipelines have also been introduced to get a robust trained model. The performance of the segmented test images was measured quantitatively using Dice Similarity Co-efficient (DSC) and is reported on average as 0.84 for CSF, 0.94 for GM, and 0.94 for WM.**

***Index Terms*—Magnetic resonance imaging (MRI), Brain tissue segmentation, NeuroNet, Residual Network (ResNet), Fully Convolution Network (FCN), IBSR18.**

I. Introduction

Brain image segmentation plays a crucial role in brain image analysis which extracts brain tissues, WM, GM, and CSF from a brain image by partitioning it into a set of disjoint regions. Pixels inside each of those regions should be homogeneous in space and intensity [1]. Segmentation of brain tissue helps to detect diseases like brain tumor, Alzheimers (AD), Parkinsons, dementia, schizophrenia, and traumatic injury. [2]. The performance of brain tissue segmentation methods depends on several factors such as location, size, shape, texture, and image contrast (due to inherent in the modalities used for image acquisition) [3].

Traditional brain tissue segmentation approaches can be classified into region-based, clustering-based, statistical based methods like gaussian mixture models and Atlas, and classification methods where features are finely engineered to be classified eventually using Support Vector Machines, for instance. In the first three algorithms, pipelines are usually used where a failure in one step might affect dramatically the final result. On the other hand, the fourth class suffers from hard feature selection and integration. Recently, Convolutional Neural Networks (CNNs) gained recognition in brain tissue segmentation [4]. Here, end-to-end segmentation is done by extracting features from images in an objective-oriented manner.

NeuroNet is a comprehensive brain image segmentation tool based on a novel multi-output CNN architecture which has been trained to reproduce simultaneously the output of multiple state-of-the-art neuroimaging tools [5]. Our contribution in this paper is to use NeuroNet to segment IBSR18 brain dataset and analyse the performance of the network under different pre-processing strategies. The paper is organized in such as section II is dedicated to describe the dataset used, while, section III is used to explain the preprocessing pipelines. We illustrate the proposed method in section IV. We describe the implementation in section V and experiment and results in section VI.

II. Dataset

The dataset used for this work, IBSR18, is a publicly available dataset by the Center for Morphometric Analysis at Massachusetts General Hospital [6]. The dataset is composed of 18 T1-W volumes with different slice thicknesses. The provided images are skull stripped and bias field corrected. For this work, the dataset was divided into three subsets: ten for training, five for validation, and three for testing. A brief description of the dataset is given in Table I. For the training and validation sets, corresponding ground truth labels (CSF, GM and WM) were provided. The training set was used for training the model, while, validation set was used to tune the proposed model. Fig. 1. shows the volume IBSR_01 and corresponding labels.

Training Dataset		
Volume Name	Volume	Spacing (mm)
IBSR 01, 03, 04, 06	$256 \times 128 \times 256$	$0.94 \times 1.5 \times 0.94$
IBSR 07,08,09	$256 \times 128 \times 256$	$1 \times 1.5 \times 1$
IBSR 16,18	$256 \times 128 \times 256$	$0.84 \times 1.5 \times 0.84$
Validation Dataset		
Volume Name	Volume	Spacing (mm)
IBSR 11, 12	$256 \times 128 \times 256$	$1 \times 1.5 \times 1$
IBSR 13,14	$256 \times 128 \times 256$	$0.94 \times 1.5 \times 0.94$
IBSR 17	$256 \times 128 \times 256$	$0.84 \times 1.5 \times 0.84$
Test Dataset		
Volume Name	Volume	Spacing (mm)
IBSR 02	$256 \times 128 \times 256$	$0.94 \times 1.5 \times 0.9375$
IBSR 10	$256 \times 128 \times 256$	$1 \times 1.5 \times 1$
IBSR 15	$256 \times 128 \times 256$	$0.84 \times 1.5 \times 0.84$

TABLE I
SUMMARY ON IBSR18 DATASET USED IN THIS PROJECT.

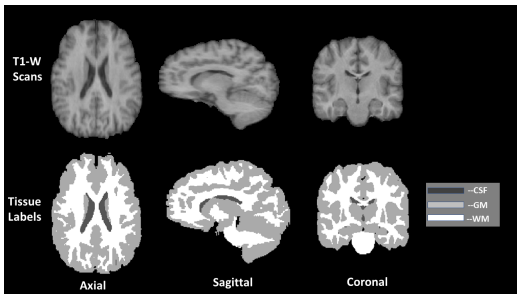

Fig. 1. Graphical Description of IBSR_01 volume and Corresponding label in axial, sagittal and coronal view.

III. PRE-PROCESING

The task of segmenting brain tissues become more challenging as different scanners are used with different parameters during acquisition. That usually leads to intensity heterogeneity, contrast variations, and different types of noise [7]. Therefore, data homogenization is often necessary.

Authors in [8] used intra-subject registration as the first step of pre-processing for Lupus segmentation. Dolz et al. in [9] used volume-wise intensity normalization, bias field correction and skull-stripping. Shakeri et al. [10] used normalization for subcortical brain structure segmentation. Nyul et al. proposed a method consist of a training stage to find standard parameters then matching the histograms to a standard histogram through a transformation stage [11].

In this work, two different pre-processing pipelines were implemented to see the effect on the performance of the deep CNN. The idea is to apply a basic standardization pipeline and a second registration-based pipeline, then measure the difference in performance. Fig. 2. shows an overview of both pipelines.

A. Pre-processing Pipeline-1

The input volumes were standardized to zero mean and unit standard deviation using the volume statistics as mentioned in the reference paper of Rajchl et al. [5]. It can be formulated as equation.1.

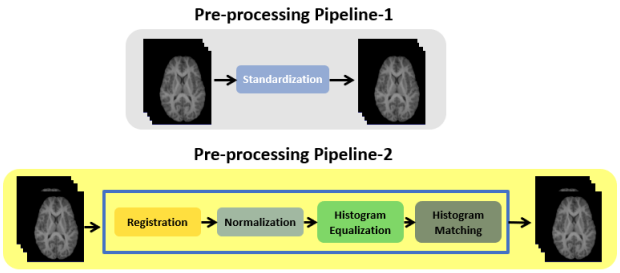

Fig. 2. Pre-processing pipelines.

$$V_{new} = \frac{V_{old} - \mu}{\sigma} \tag{1}$$

Where μ, σ represent the mean and standard deviation of all pixels in the corresponding volume, respectively.

B. Pre-processing Pipeline-2

1) Registration: is the alignment of two or more images in a common anatomical space [12], commonly called the fixed space. As shown in Table I, training, validation and test datasets have different spacing. To standardize the voxel spacing in the dataset, we registered the images and transformed the labels correspondingly to Montreal Neurological Institute (MNI) template (152 subjects, 1x1x1mm T1w, dimensions: $182 \times 218 \times 182$, skull stripped) [13]. Fig. 3 shows the process of this registration. Simple-ITK framework in Python was used for that purpose [14]. The idea is first to register the dataset to MNI template using rigid transformation, then save the corresponding final transformation matrix for later use. The available dataset was used as moving images, while, MNI template was used as the fixed image. After getting the final predicted labels, those should be brought back to the original space by using the inverse of the corresponding transformation matrix.

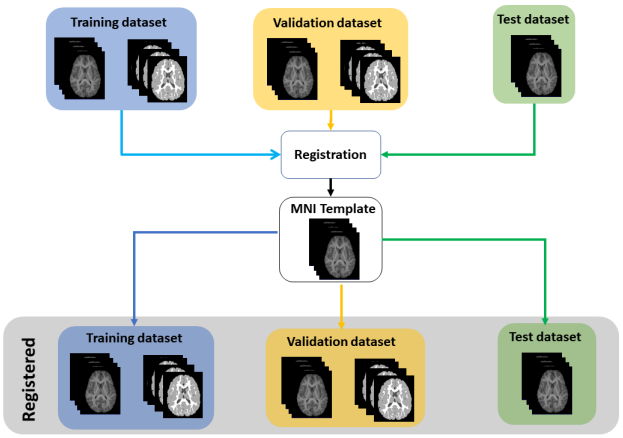

Fig. 3. Registration to MNI template.

2) Normalization: In this step, intensity rescaling was applied on each volume. The intensity range of each volume

v was rescaled to the range $[0, 1]$. It can be formulated as equation 2.

$$v_{new} = \frac{v - min(v)}{max(v) - min(v)} \tag{2}$$

3) Histogram pre-processing: In this step, the procedure was as follows:

a) Reference Selection: by analyzing tissues distribution for all volumes, image IBSR_07 was nominated due to the following reasons:
- It has a relatively wide spectrum of intensity values, see Figure 4 left part.
- The overlapping between GM and WM is acceptable, see Figure 4 middle part.
- GM and WM had comparable shares, this will help in the next step of histogram equalization of the reference image.

b) Histogram Equalization of the Reference Volume: adaptive histogram equalization was applied on the reference normalized volume IBSR_07. It's a process of adaptive image-contrast enhancement based on a generalization of histogram equalization (HE) [15].

c) Histogram Matching to the Reference: apply histogram matching of all the dataset (minus IBSR_07) to the reference volume. As a last step of the pre-processing pipeline-2, histogram matching was performed to map all volumes' histogram distributions to the reference volume's one. Fig. 5. shows 5 raw volumes and the pre-processed volumes after applying pre-processing pipeline-2.

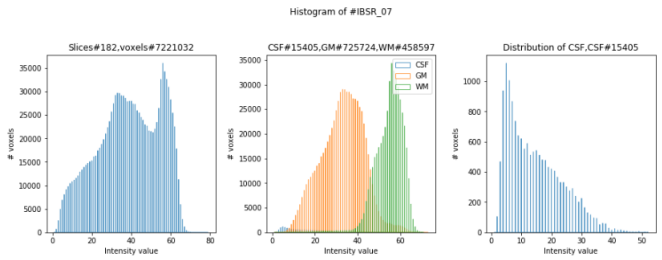

Fig. 4. tissues intensity distribution. On the left, the complete distribution. In the middle, white matter and grey matter distributions. On the right, CSF distribution.

IV. Method

A. Network Architecture

In this paper, we adopted the NeuroNet [5] architecture for which the code is available at the Github repository of DLTK models [16]. NeuroNet is a deep convolutional neural network with multi outputs, which was trained on 5,000 T1-weighted brain MRI scans from the UK Biobank Imaging Study that have been automatically segmented into brain tissue and cortical and sub-cortical structures using the standard neuroimaging pipelines [5]. For our work, as the desired output is tissue segmentation only, the architecture was modified to an updated FCN architecture [17] with a

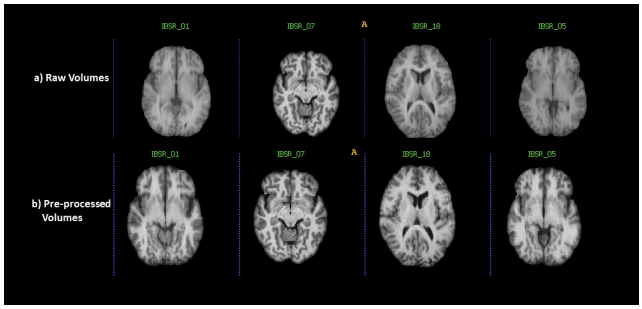

Fig. 5. The effect of pre-processing the dataset. At the top, 4 cases before applying the pre-processing pipelines. At the bottom, the final pre-processed dataset.

ResNet encoder [18] as presented in [19]. Fig. 6 shows the original NeuroNet Architecture and the adopted architecture in this work. One initial convolution was performed on input

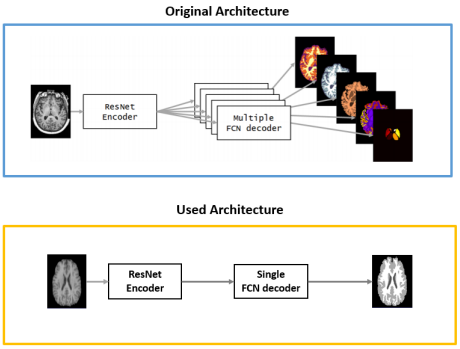

Fig. 6. Network Architectures

volumes, afterwards, features were extracted using the ResNet encoder [18] [19]. Features were extracted in encoder part with two residual units (N_{unit}=number of residual units at each scale), $U_i^{S_j} = \left\{U_1^{S_j}, U_2^{S_j}\right\}$ on each of the resolution scales $S_j = \{S_1, ..., S_{N_{scales}}\}$, where N_{scales} is the number of resolutions scales. We used $N_{scales} = 4$, as in the default implementation [5]. Leaky ReLu (with leakiness = 0.1) was used as the activation function [20] with preceding batch normalization. At each scale, the down-sampling was performed using stride convolution [21] where the strides were $s_j = \{1, 2, 2, 2\}$ operate on each spatial dimension. As defined in the reference paper [5], a fixed number of filters for all the convolutions in any U^{S_j} was used doubling the number at each scale: 16, 32, 64 and 128.

In image segmentation, Fully Convolutional Networks (FCNs) are among widely used networks which typically reconstruct the prediction with the same size of the input given. In the original Neuronet architecture, the decoding part was based on multi-decoder architecture on FCN upscore operations [17]. The prediction was reconstructed at each resolution scale S_{j-1} by up-sampling the prediction linearly at S_j scale and adding skip connection from the output of the last residual unit $U_2^{S_j}$. The output of the last residual unit at decoder serves as output of the network. A prediction was

Model Configuration					Dice Coefficient Validation Set		
Model No.	#Training Steps	Patch Size	Samples	Weights Initializations	CSF	GM	WM
1.1	1000	128x128x128	200	NeuroNet Pretrained	0.43±0.40	0.90±0.01	0.88±0.06
1.2	5000	128x128x128	400	NeuroNet Pretrained	0.80±0.12	0.89±0.05	0.89±0.03
2.1	4000	32x32x32	200	Uniform Distribution	0.07±0.03	0.71±0.04	0.72±0.06
2.2	4000	64x64x64	200	Uniform Distribution	0.80±0.05	0.90±0.02	0.89±0.03
2.3	4000	128x128x128	50	Uniform Distribution	0.89±0.02	0.93±0.01	0.93±0.01
2.4	2000	128x128x128	50	NeuroNet Pretrained	0.89±0.02	0.94±0.01	0.93±0.01
2.5	**4000**	**128x128x128**	**50**	**NeuroNet Pretrained**	**0.90±0.02**	**0.94±0.01**	**0.93±0.02**

TABLE II
VALIDATION DSC OF DIFFERENT NETWORKS TRAINED WITH DIFFERENT HYPER-PARAMETERS. BOLD FONT HIGHLIGHTS HIGHESTS DSC

obtained after a softmax layer. The loss was calculated using categorical cross-entropy loss for all prediction outputs $\hat{y}$ at voxel locations v.

$$L(\hat{y}, y) = -\sum \hat{y}(v) \log y(v) \tag{3}$$

Where y is the true label volume.

V. IMPLEMENTATION AND ENVIRONMENT

NeuroNet is implemented using Deep Learning Toolkit (DLTK) for Medical Image Analysis [19] on TensorFlow [22] with SimpleITK [23]. Pre-prpcessing pipelines used in this work were implemented using DLTK and SimpleITK frameworks as well. The training was carried on GeForce GTX 1080 GPU with memory of 2.7GB.

VI. EXPERIMENTS AND RESULTS

Dice Similarity Coefficient (DSC) was used for evaluating predicted labels and guiding the tuning process. Our main idea of the experiments was: analysing the performance of the model with default parameters suggested in [5] with proposed pre-processing pipelines and tune the parameters to improve the results.The corresponding results are shown in Table II.

Firstly, we trained our model with the processed data that went through only Standardization (zero mean and unit standard deviation) mentioned in section III-A. The model was trained from scratch, but in the case of training deep models, enough care needs to be taken to initialize the parameters as shown in [24]. We used the pre-trained weights of Neuronet [5] as the initial ones to avoid gradient vanishing problem. We trained the model for 1000 steps with 200 randomly extracted patches of size $128 \times 128 \times 128$. The reason beyond choosing such a big patch size was that authors in [5] used it in the original implementation. Performance on the validation data is shown in Table II corresponding to model 1.1.

In the second attempt, the training steps were increased five times than before and the model was trained for 5000 steps with double the number of patches (400 patches). That caused a huge improvement in the model's performance in CSF and slight increase and decrease in WM and GM respectively. Results are shown in Table II corresponding to model 1.2. Afterwards, more training was performed but no significant improvement was achieved, which led to looking for different pre-processing strategies.

Thirdly, the proposed pre-processing pipeline-2 was applied on the datasets explained in section III-B. The pipeline consists of registration, normalization, adaptive histogram equalization and histogram matching steps. We tried three different cases using pipeline-2 for pre-processing, those cases were as follows:

- Case 1 : patch size: 32x32x32, 200 samples.
- Case 2 : patch size: 64x64x64, 200 samples.
- Case 3 : patch size: 128x128x128, 50 samples.

Table II, model 2.1, 2.2 and 2.3 shows Case 1, Case 2 and Case 3 results respectively. model 2.1, 2.2 and 2.3 were trained from end to end using the Uniform distribution as weight initialization. Model 2.1, which was trained on patches of size 32x32x32, performed poorly and failed to segment CSF region, shown Table II. Training the model with bigger patch sizes turned out to be better as used in the original implementation. Overall, this model, 2.5, proved to have the best combination of parameters compared to all other experiments shown in Table II.

Finally, we performed the test on the test set using the best model and achieved average DSC 0.84 for CSF, 0.94 for GM and 0.94 for WM.

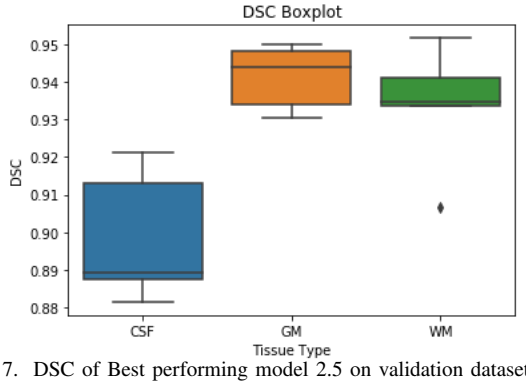

Fig. 7. DSC of Best performing model 2.5 on validation dataset

VII. CONCLUSION

In this work, we proposed using NeuroNet architecture for 3D brain tissue segmentation. Analysis the performance of the CNN under different hyper-parameter tuning was investigated.To the best of our knowledge, this study is the first effort to use Neuronet with IBSR 2018 data set. Our results show that pre-processing pipeline 2 was clearly effective in boosting the performance where histogram equalization and matching played a great role. Registration assisted mainly in unifying the voxel spacing to make all the inputs have similar

spatial spaces. The use of NeuroNet pre-trained weights was significantly helping the network to start from a good initial point. As compared to starting with random initialization, the pretrained weights were performing enormously better even though the size of the used dataset was relatively small (10 volumes compared to 5000 original size). Patch sizes had important effect on the performance as well. The original patch size was the best fit, while, smaller sizes just did not work as good.

Future work includes extending the analysis with 3D Unet and modified ResNet-Unet architectures. We also plan to ensemble different models to improve the DSC for CSF specially. Finally, we want to extend the use of same CNN in disease segmentation.

REFERENCES

[1] Lingraj Dora, Sanjay Agrawal, Rutuparna Panda, and Ajith Abraham. State-of-the-art methods for brain tissue segmentation: A review. *IEEE Reviews in Biomedical Engineering*, 10:235–249, 2017.

[2] Sudip Kumar Adhikari, Jamuna Kanta Sing, Dipak Kumar Basu, and Mita Nasipuri. Conditional spatial fuzzy c-means clustering algorithm for segmentation of mri images. *Appl. Soft Comput.*, 34:758–769, 2015.

[3] Lei Wen, Xingce Wang, Zhongke Wu, Mingquan Zhou, and Jesse S. Jin. A novel statistical cerebrovascular segmentation algorithm with particle swarm optimization. *Neurocomputing*, 148:569 – 577, 2015.

[4] Mohammad Havaei, Axel Davy, David Warde-Farley, Antoine Biard, Aaron C. Courville, Yoshua Bengio, Christopher Joseph Pal, Pierre-Marc Jodoin, and Hugo Larochelle. Brain tumor segmentation with deep neural networks. *Medical image analysis*, 35:18–31, 2017.

[5] Martin Rajchl, Nick Pawlowski, Daniel Rueckert, Paul M. Matthews, and Ben Glocker. NeuroNet: Fast and Robust Reproduction of Multiple Brain Image Segmentation Pipelines. *arXiv e-prints*, page arXiv:1806.04224, June 2018.

[6] Ibsr dataset. https://www.nitrc.org/frs/?group$_id = 48. Accessed : 2019 - 01 - 05.

[7] Xiaofei Sun, Lin Shi, Yishan Luo, Wei Yang, Hongpeng Li, Peipeng Liang, Kuncheng Li, Vincent C. T. Mok, Winnie C. W. Chu, and Defeng Wang. Histogram-based normalization technique on human brain magnetic resonance images from different acquisitions. *BioMedical Engineering OnLine*, 14(1):73, Jul 2015.

[8] Eloy Roura, Nicolae Sarbu, Arnau Oliver, Sergi Valverde, Sandra Gonzlez-Vill, Ricard Cervera, Nria Bargall, and Xavier Llad. Automated detection of lupus white matter lesions in mri. *Frontiers in Neuroinformatics*, 10:33, 2016.

[9] Jose Dolz, Christian Desrosiers, and Ismail Ben Ayed. 3d fully convolutional networks for subcortical segmentation in mri: A large-scale study. *NeuroImage*, 170:456 – 470, 2018. Segmenting the Brain.

[10] Mahsa Shakeri, Stavros Tsogkas, Enzo Ferrante, Sarah Lippe, Samuel Kadoury, Nikos Paragios, and Iasonas Kokkinos. Sub-cortical brain structure segmentation using F-CNN's. In *ISBI 2016: International Symposium on Biomedical Imaging*, Prague, Czech Republic, 2016.

[11] L. G. Nyul, J. K. Udupa, and Xuan Zhang. New variants of a method of mri scale standardization. *IEEE Transactions on Medical Imaging*, 19(2):143–150, Feb 2000.

[12] Arno Klein, Jesper Andersson, Babak A. Ardekani, John Ashburner, Brian Avants, Ming-Chang Chiang, Gary E. Christensen, D. Louis Collins, James Gee, Pierre Hellier, Joo Hyun Song, Mark Jenkinson, Claude Lepage, Daniel Rueckert, Paul Thompson, Tom Vercauteren, Roger P. Woods, J. John Mann, and Ramin V. Parsey. Evaluation of 14 nonlinear deformation algorithms applied to human brain mri registration. *NeuroImage*, 46(3):786 – 802, 2009.

[13] VS Fonov, AC Evans, RC McKinstry, CR Almli, and DL Collins. Unbiased nonlinear average age-appropriate brain templates from birth to adulthood. *NeuroImage*, 47:S102, 2009. Organization for Human Brain Mapping 2009 Annual Meeting.

[14] Ziv Yaniv, Bradley C. Lowekamp, Hans J. Johnson, and Richard Beare. Simpleitk image-analysis notebooks: a collaborative environment for education and reproducible research. *Journal of Digital Imaging*, 31(3):290–303, Jun 2018.

[15] J. A. Stark. Adaptive image contrast enhancement using generalizations of histogram equalization. *IEEE Transactions on Image Processing*, 9(5):889–896, May 2000.

[16] Dltk models. https://github.com/DLTK/models/tree/master/ukbb_neuronet_brain_segmentation.htm. Accessed: 2019-01-05.

[17] Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440, 2015.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[19] Nick Pawlowski, Sofia Ira Ktena, Matthew C. H. Lee, Bernhard Kainz, Daniel Rueckert, Ben Glocker, and Martin Rajchl. DLTK: state of the art reference implementations for deep learning on medical images. *CoRR*, abs/1711.06853, 2017.

[20] Andrew L. Maas, Awni Y. Hannun, and Andrew Y. Ng. Rectifier Nonlinearities Improve Neural Network Acoustic Models.

[21] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin A. Riedmiller. Striving for simplicity: The all convolutional net. *CoRR*, abs/1412.6806, 2014.

[22] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Gregory S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian J. Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Józefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Gordon Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul A. Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda B. Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *CoRR*, abs/1603.04467, 2016.

[23] Bradley Lowekamp, David Chen, Luis Ibanez, and Daniel Blezek. The design of simpleitk. *Frontiers in Neuroinformatics*, 7:45, 2013.

[24] Konstantinos Kamnitsas, Christian Ledig, Virginia F.J. Newcombe, Joanna P. Simpson, Andrew D. Kane, David K. Menon, Daniel Rueckert, and Ben Glocker. Efficient multi-scale 3d cnn with fully connected crf for accurate brain lesion segmentation. *Medical Image Analysis*, 36:61 – 78, 2017.