3D intracranial artery segmentation using a convolutional autoencoder

Li Chen[1], Yanjun Xie[2], Jie Sun[3], Niranjan Balu[3], Mahmud Mossa-Basha[3], Kristi Pimentel[3], Thomas S. Hatsukami[4], Jenq-Neng Hwang[1], Chun Yuan[3†]

Department of 1. Electrical Engineering; 2. Mechanical Engineering; 3. Radiology; 4. Surgery
University of Washington
Seattle, WA, 98195, USA
{cluw, yanjunx. sunjie, ninja, mmossab, kristidb, tomhat, hwang, cyuan }@uw.edu
†Corresponding author

Abstract— **Automated segmentation of intracranial arteries on magnetic resonance angiography (MRA) allows for quantification of cerebrovascular features, which provides tools for understanding aging and pathophysiological adaptations of the cerebrovascular system. Using a convolutional autoencoder (CAE) for segmentation is promising as it takes advantage of the autoencoder structure in effective noise reduction and feature extraction by representing high dimensional information with low dimensional latent variables. In this paper, we trained an 8-layer CAE to learn a 3D segmentation model of intracranial arteries from 49 cases of MRA data. After parameter optimization and prediction refinement, our trained model was shown to perform better than the three traditional segmentation methods in both binary classification and visual evaluation.**

Keywords- convolutional autoencoder; deep neural network; machine learning; artery segmentation; Magnetic Resonance Angiography

I. INTRODUCTION

A healthy cerebrovascular system that delivers sufficient blood flow to all brain territories is of vital importance for maintaining brain health. Indeed, the pathogeneses of many common neurological diseases, such as stroke and dementia, have been revealed to involve significant vascular contributions [1]. Free from contrast medium and ionizing radiation, 3D time-of-flight (TOF) magnetic resonance angiography (MRA) is the most widely-used neurovascular imaging technique, which is able to display a comprehensive map of the cerebrovascular tree. Automated segmentation of the cerebrovascular tree on TOF MRA allows for quantification of global and territorial vascular features, which can be applied to serial images to understand changes in the cerebrovascular system under various physiological and pathological conditions.

However, it is challenging to accurately segment intracranial arteries on TOF MRA given the complex network of cerebral arteries with substantial inter-individual variations [2], and weak signals in small arteries due to slow or in-plane blood flow.

Previous automated intracranial artery segmentation and tracing methods have mostly relied on pattern recognition or model-based approaches such as vesselness filters [3], region growing [4] and fast marching [5]. Some of these methods were shown to be effective in a limited anatomical region or for a specific patient group. However, as traditional image processing methods depend heavily on human specified vessel descriptors, it remains difficult to generate a comprehensive cerebrovascular map with consistently high performance for general cases. A more robust method is needed for intracranial artery segmentation.

Recently, neural networks have shown advantages over traditional medical image segmentation methods, such as in skull stripping [6], and brain tumor segmentation [7]. These neural-network-based methods allow computational models composed of multiple processing layers to learn representations of data with multiple levels of abstraction and discover intricate structures in large data sets.

An auto-encoder (AE) model is a neural network used for unsupervised learning of efficient coding. The encoder part transforms an input into a typically lower-dimensional representation, and the decoder part is tuned to reconstruct the initial input from this representation [8]. The purpose of an AE was to learn a representation from a set of data, instead of predicting the target given input. However, Vincent et al. modified the AE to a denoising autoencoder (DAE) [9] to restore artificially corrupted input, as the model is trained to extract more useful features from the original input to solve the much harder denoising problem. For this application, the model is trained in a supervised way to predict the target rather than learning representations. When used in image processing, a convolutional autoencoder (CAE), one of the DAE models using convolutional layers and deconvolution layers, is an efficient and promising structure, as it fully utilizes the advantages of convolutional neural networks (CNNs), which have proven to be effective with noisy, shifted or corrupted image data [10], allowing patterns to be learned from local pixels and positioned at various locations.

In this paper, we used the CAE structure and trained the model in a supervised learning manner with labeled images as ground truth. The CAE structure was expected to denoise and extract useful features from the original images, and thus

978-1-5090-3050-7/17/$31.00 ©2017 IEEE

was optimized to perform the intracranial artery segmentation on 3D TOF MRA.

II. METHODS

A. Data

Forty-nine brain TOF MRA images (620*620*243 in size, resolution of 0.3 mm^3 per voxel) were acquired on a 3T Philips Ingenia MRI scanner using a standard head coil in our lab. Imaging parameters were: TR/TE = 14.7/3.5ms, flip angle = 18°. The study was performed with IRB approval and informed consents were obtained for all participating subjects.

B. Preprocessing

We applied intensity correction [11] to correct image inhomogeneity and intensity normalization [12] to unify intensity ranges of all subjects between 0 to 1.

All the images were processed semi-automatedly [13]. Arteries were first traced by a deformable model. Then, an experienced radiologist checked vascular connections to finalize the artery tracing results (3D positions along the centerline with radius). The traces were further verified by another radiologist. 3D labeled images were then created as ground truth by filling vessel surfaces rendered in the Visualization ToolKit (VTK) [14]. A representative case of the generated labeled image patches, along with the original image patches, is shown in Fig. 1.

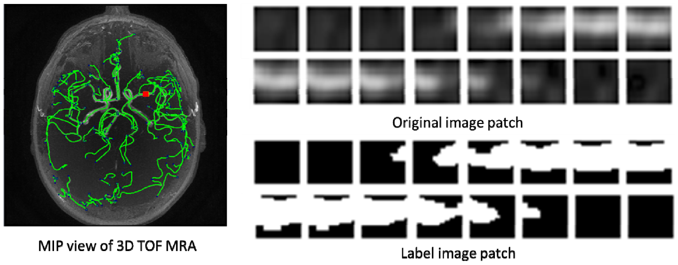

Fig. 1. A representative case of the original image patches (top right) and the corresponding labeled image patches (bottom right) shown in 2D slices. The origin of the patch is shown in a red square in the Maximum Intensity Projection (MIP) view of TOF MRA with arteries traced in green (left).

C. Data separation

Forty-nine pairs of original and corresponding labeled images were randomly separated into three individual sets for unbiased validation and evaluation: 46 pairs for the training set, 2 pairs for the validation set (tuning network parameters), 1 pair for the test set (evaluating performance).

D. Network structure

With 3D TOF MRA images as input and binary coded 3D labeled images as output, our task can be categorized as a supervised learning for a two-class classification task.

Instead of using the whole 3D images as input and output, we used a cubic window of 16*16*16 voxels to slide through original and labeled images to extract a batch of image patches for training and prediction. The reasons for using patches instead of the whole 3D images are as follows: 1) Intracranial arteries have similar tubular structures; 2) Patch extraction significantly increases training samples and reduces the neural network scale; 3) The diameter of largest arteries rarely exceeds 16 pixels; 4) Computational efficiency and memory saving are considered.

We used an 8-layer convolutional autoencoder structure, as shown in Fig. 2. The encoder part consists of two consecutive blocks of convolution layers followed by max pooling layers. In the decoder part, we use deconvolution with strides of 2 to restore the dimension. Size of convolution/deconvolution kernels are cubes of 3. Kernel numbers are 32 for the first and last layer, and 16 for middle layers.

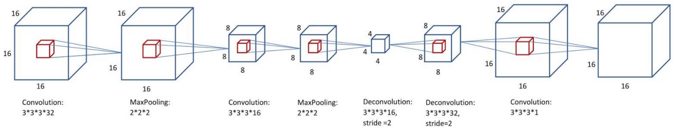

Fig. 2. Structure of the proposed convolutional autoencoder (CAE) model.

E. Training process

Due to hardware limitations, only about 20,000 patches of the original and labeled images could be trained at a time. However, even one original image can be cut into at most 80 million patches. The sliding stride is critical, as a small stride may cause similar patches to be selected repeatedly whereas a large stride may miss many regions with artery voxels.

We selected patches using a varied sliding window. A smaller stride of 5 first extracts patches if there is at least one voxel labeled as vascular region in labeled patches; then a larger stride of 14 is estimated according to the selection rate in the previous step to extract a similar number of all non-artery patches.

We trained the CAE model using patches from one image at a time. After all images were trained, the process was repeated.

F. Prediction process

Test images were cut into patches with the same sizes as the training data with half size overlapping in any direction, as input to the CAE. The prediction of CAE is a probability image showing likelihood of voxels to be artery regions. The predicted values from overlapped voxels were averaged when combining patches back to the original image size.

We applied a threshold to the probability image to generate binary prediction results for comparison with the labeled image. The threshold value giving the highest accuracy rate from the validation sets was applied for prediction in the test image.

We further applied a closing operation (dilate then erode) with a cubic structure-element of 3*3*3 after thresholding.

G. Parameter optimization

Several different scenarios are tried for optimizing a list of parameters for the CAE. Four metrics were used to evaluate the performance: 1) top Youden index (true positive rate – false positive rate); 2) area under the curve (AUC); 3) highest accuracy; and 4) calculation and memory cost for implementation.

The training parameters we optimized are: 1) activation function: rectified linear unit (ReLU), hyperbolic tangent (tanh), and the sigmoid function; 2) down/up sampling blocks number; 3) kernel numbers of convolution/deconvolution layers; 4) down-sampling method: max-pooling followed by convolution, or convolution with stride; 5) up-sampling method: up-sampling with neighbor voxels followed by deconvolution, or deconvolution with stride; 6) number of training images; 7) repetition: the times all the images were extracted and trained once.

H. Performance evaluation

The post-processed binary image was compared with the labeled image voxel by voxel. Binary classification performance was evaluated by accuracy, sensitivity, specificity, precision, and the dice similarity co-efficient (DSC), defined as

$$DSC = \frac{2(A \cap B)}{(A + B)}$$

where A is the ground truth result and B is the segmentation result. DSC ranges from 0 (no overlap) to 1 (identical results). DSC > 0.7 indicates excellent agreement [15].

As a reference, we also evaluated segmentation results by three traditional image processing methods: 1) Renyi Entropy method [16], the best among 16 threshold methods in the ImageJ toolkit [17]; 2) Phansalkar local threshold [18], the best among 9 local threshold methods in ImageJ; 3) the Frangi vesselness filter [3].

I. Computational setup

Codes are written using Tensorflow [19] and Keras [20]. The optimized CAE model was trained on a workstation with Intel® Xeon® CPU E5-2630 v3 @2.4GHz 8 cores, 32 GB Memory, NVIDIA GeForce GTX TITAN X on Windows 7.

III. EXPERIMENTAL RESULT

A. Parameter optimization

Training parameters were tuned and metrics were calculated for each option. The final optimized choices are: activation with ReLU, down/up sampling blocks of 2, kernel numbers: 32 and 16, downsample with max-pooling, upsample using deconvolution with stride, training with all images in training set with repetition of 2.

B. Comparison with traditional methods

TABLE I BINARY CLASSIFICATION EVALUATION BETWEEN CAE (WITH THRESHOLD) AND CLASSICAL SEGMENTATION METHODS

Models	Accuracy	Sensitivity	Specificity	Precision	DSC
CAE (0.55)	0.99788	0.63138	0.99961	0.88550	0.73716
CAE (0.55) closing	0.99789	0.63422	0.99961	0.88438	0.73869
Renyi	0.99500	0.52331	0.99724	0.47272	0.49673
Phansalkar	0.85897	0.92017	0.85868	0.02992	0.05795
Frangi	0.99731	0.56023	0.99937	0.80856	0.66187
CAE (0.03)	0.99485	0.92264	0.99520	0.47634	0.62830

The binary classification results from CAE as well as traditional segmentation methods are shown in TABLE I.

C. Visual evaluation of segmented results

A threshold of 0.55 was selected and a morphological closing operation was then applied to the binary image. We found the closing operation is able to connect discontinued small arteries and fill small holes inside artery regions.

The generated MIP views of the labeled image, predicted image by CAE, and segmentation results using traditional methods of the test set in VTK as shown in Fig. 3.

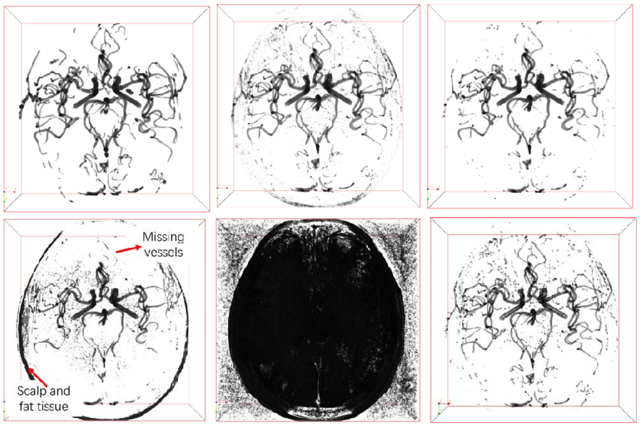

Fig. 3. Top Left: Labeled image. Top middle: Predicted image (with possibility from 0 to 1) by CAE. Top Right: Predicted image after thresholding. Bottom Left to Right: Segmentation results by Renyi Entropy threshold, Phansalkar local threshold, and Frangi filtering.

The posterior communicating artery (PComm), a small artery that is easily missed but clinically important, is examined by observing segmentation results as shown in Fig. 4. CAE correctly identified the PComm artery when it was missed by the Renyi method. Frangi filtered arteries had discontinuity problems near bifurcations.

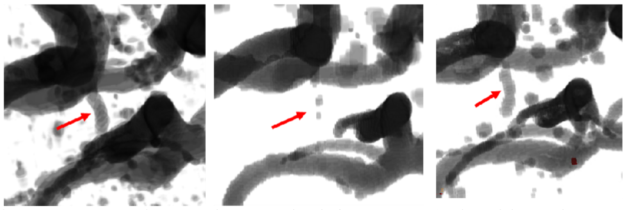

Fig. 4. Segmentation results near the left PComm (pointed by red arrows) predicted by CAE (Left), Renyi Entropy Threshold (Middle), and Frangi Filter (Right). Phansalkar method is too noisy for visualization.

D. Processing time

The training time for all the 46 MRA sets was 592 minutes. Prediction for each image took about one minute.

IV. DISCUSSION

A. Network structure

Autoencoder models have typically been used as an unsupervised learning approach to extract useful features from unlabeled data. For our supervised learning application, the idea of extracting useful features from voxels is similar, but we are more interested in the performance of the final decoded results.

Vascular segmentation is different from tumor or skull segmentation in that the target is relatively small in volume and complex in shape. To segment fine details, a 3D neural network model is necessary. In some 3D applications [6], 2D slices of 3D images were trained, predicted, then stacked together. We used 3D patches and operations in our solution. Despite increased calculation time, additional information between slices contributed to consistent segmentation between slices. The use of patches reduces the model complexity and demand for training size.

One of the benefits of the neural network structure used in CAE is its adaptiveness to data. With little need for optimization, this model is also applicable to other images and has potential for segmentation tasks in other modalities.

B. *Training process*

Accurately labeled images are critical in training CAE models. However, labels are difficult to acquire for intracranial artery segmentation. It is time-consuming for humans to manually segment 3D artery regions, and it is equally challenging for automated algorithms to consistently generate high quality artery segmentation results. To balance time efficiency and segmentation quality, we used a semi-automated method to trace arteries in 3D space with human intervention and correction as the ground truth. It still takes more than one hour to trace and validate one case.

C. *Comparison with traditional methods*

As accuracy alone is not enough to evaluate the model performance when 99.4% of voxels are non-artery regions, five metrics were used for comparison between models.

We found CAE outperformed the Renyi Entropy and Frangi filter methods in all the five metrics, while the Phansalkar method has high sensitivity but low precision. If the threshold is set to 0.03, CAE outperforms the Phansalkar method in all metrics.

From visual evaluation of the segmentation results, we found the CAE method removed most scalp and fat tissue around the brain while maintained the main structure of artery tree and even small arteries, such as the PComm. Still, some arteries were not segmented continuously, especially for peripheral segments of middle cerebral arteries where signal intensity becomes relatively low. Nonetheless, the overall visualization effect of the CAE method is superior to the traditional methods.

D. *Future work*

While CAE showed excellent agreement with ground truth (DSC: 0.74) in intracranial artery segmentation, additional improvement is possible. Future improvements may target better network training, incorporating patch location information, and improved network models.

V. CONCLUSION

We trained a convolutional autoencoder for intracranial artery segmentation on brain TOF MRA. Our model outperformed three traditional segmentation methods in both binary classification and visual evaluation. Accurate segmentation of the whole cerebral vasculature will facilitate quantification of global and territorial vascular features.

ACKNOWLEDGMENT

This research is supported by grants from the National Institutes of Health (R01 NS083503, R01 NS092207, and R01 HL103609) and the American Heart Association (17MCPRP33671077). We are grateful to the support of NVIDIA Corporation with donation of the Titan X GPU.

REFERENCES

[1] Z. Arvanitakis et. al, "Relation of cerebral vessel disease to Alzheimer's disease dementia and cognitive function in elderly people: a cross-sectional study," Lancet Neurol., vol. 15, no. 9, pp. 934–943, 2016.

[2] B. J. ALPERS et. al, "Anatomical Studies of the Circle of Willis in Normal Brain," Arch. Neurol. Psychiatry, vol. 81, no. 4, p. 409, Apr. 1959.

[3] A. F. Frangi et. al, "Multiscale vessel enhancement filtering," Medial Image Comput. Comput. Invervention - MICCAI'98. Lect. Notes Comput. Sci. vol 1496, vol. 1496, no. 6, pp. 130–137, 1998.

[4] F. Mut, et. al, "Morphometric, geographic, and territorial characterization of brain arterial trees," Int. j. numer. method. biomed. eng., vol. 30, no. 7, pp. 755–766, 2014.

[5] C. Y. Hsu et. al, "Automatic recognition of subject-specific cerebrovascular trees," Magn. Reson. Med., pp. 398–410, Jan-2017.

[6] J. Kleesiek et. al, "Deep MRI brain extraction: A 3D convolutional neural network for skull stripping," Neuroimg, vol.129, pp.460–469, 2016.

[7] A. Pinto et. al, "Brain Tumor Segmentation using Convolutional Neural Networks in MRI Images," IEEE Trans. Med. Imaging, vol. 35, no. 5, pp. 1240–1251, 2016.

[8] H. Bourlard and Y. Kamp, "Auto-Association by Multilayer Perceptrons and Singular Value Decomposition," Biol. Cybern, vol. 59, pp. 291–294, 1988.

[9] P. Vincent et. al, "Extracting and composing robust features with denoising autoencoders," Proc. 25th Int. Conf. Mach. Learn. - ICML '08, pp. 1096–1103, 2008.

[10] Y. LeCun et. al, "Gradient-based learning applied to document recognition," Proc. IEEE, vol. 86, no. 11, pp. 2278–2323, 1998.

[11] C. Han et. al, "A multi-scale method for automatic correction of intensity non-uniformity in MR images," J. Magn. Reson. Imaging, vol. 13, no. 3, pp. 428–436, Mar. 2001.

[12] L. G. Nyul et. al, "New variants of a method of MRI scale standardization," IEEE Trans. Med. Img, vol. 19, no. 2, pp. 143–150, 2000.

[13] L. Chen et. al., "Development of a Quantitative Intracranial Vascular Features Extraction Tool on 3D MRA Using Semi-automated Open-Curve Active Contour Vessel Tracing," Magn. Reson. Med. 10.1002/mrm.26961.

[14] W. Schroeder et. al, The visualization toolkit : an object-oriented approach to 3D graphics. Kitware, 2006.

[15] L. R. Dice, "Measures of the Amount of Ecologic Association Between Species," Ecology, vol. 26, no. 3, pp. 297–302, Jul. 1945.

[16] J. N. Kapur et.al, "A new method for gray-level picture thresholding using the entropy of the histogram," Comput. Vision, Graph. Image Process., vol. 29, no. 3, pp. 273–285, 1985.

[17] J. Schindelin et. al, "The ImageJ ecosystem: An open platform for biomedical image analysis," Mol. Reprod. Dev., vol. 82, no. 7–8, pp. 518–529, 2015.

[18] N. Phansalkar et. al, "Adaptive local thresholding for detection of nuclei in diversity stained cytology images," ICCSP 2011 - 2011 Int. Conf. Commun. Signal Process., pp. 218–220, 2011.

[19] M. Abadi et. al., "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems," 2016.

[20] F. Chollet, "Keras," GitHub repository, 2015.