Published in final edited form as:

Proc SPIE Int Soc Opt Eng. 2019 February ; 10949: . doi:10.1117/12.2512529.

Two-level Training of a 3d U-Net for Accurate Segmentation of the Intra-cochlear Anatomy in Head CTs with Limited Ground Truth Training Data

Dongqing Zhang, Rueben Banalagay, Jianing Wang, Yiyuan Zhao, Jack H. Noble, Benoit M. Dawant

Dept. of Electrical Engineering and Computer Science, Vanderbilt University, Nashville, TN 37235, USA

Abstract

Cochlear implants (CIs) use electrode arrays that are surgically inserted into the cochlea to treat patients with hearing loss. For CI recipients, sound bypasses the natural transduction mechanism and directly stimulates the neural regions, thus creating a sense of hearing. Post-operatively, CIs need to be programmed. Traditionally, this is done by an audiologist who is blind to the positions of the electrodes relative to the cochlea and only relies on the subjective response of the patient. Multiple programming sessions are usually needed, which can take a frustratingly long time. We have developed an image-guided cochlear implant programming (IGCIP) system to facilitate the process. In IGCIP, we segment the intra-cochlear anatomy and localize the electrode arrays in the patient's head CT image. By utilizing their spatial relationship, we can suggest programming settings that can significantly improve hearing outcomes. To segment the intra-cochlear anatomy, we use an active shape model (ASM)-based method. Though it produces satisfactory results in most cases, sub-optimal segmentation still happens. As an alternative, herein we explore using a deep learning method to perform the segmentation task. Large image sets with accurate ground truth (in our case manual delineation) are typically needed to train a deep learning model for segmentation but such a dataset does not exist for our application. To tackle this problem, we use segmentations generated by the ASM-based method to pre-train the model and fine-tune it on a small image set for which accurate manual delineation is available. Using this method, we achieve better results than the ASM-based method.

Keywords

Cochlear implant; image segmentation; 3d deep neural networks

INTRODUCTION

Cochlear implants (CIs) are one of the most successful neural prosthetics that have been developed in the past few decades. They are used to treat patients with severe-to-profound hearing loss. In a CI surgery, an electrode array is inserted into the cochlea. Post-operatively, the CI is programmed. The spatial location of the electrode array relative to the cochlea is important for CI programming. Traditionally, this process is done by an audiologist who is blind to this information. As a result, the programming is a trial-and-error process. Multiple

sessions are usually needed and can take a frustratingly long time. In recent years, our group has developed an image-guided cochlear implant programming (IGCIP) system [1]. In IGCIP, we use image processing algorithms to automatically segment the intra-cochlear anatomy, typically in the patient's pre-operative CT image [2], and to localize the electrode array in the post-operative CT image [3,4,5]. Using such knowledge derived from the CT images, we can suggest programming settings that have been shown to significantly improve hearing outcomes [6,7,8].

The intra-cochlear anatomy includes three main structures: the scala tympani (ST), the scala vestibuli (SV), and the modiolus (MD). Segmenting them in pre-operative CTs is very challenging because these structures are small and the resolution of clinical CTs is too low to clearly show their boundaries. Currently, we use an active shape model (ASM)-based method [2]. In this method, a deformable shape model is trained from a group of anatomy examples, each represented as a set of points on the anatomical surface. These shapes are obtained by manual delineation in a group of high-resolution microCTs (μ CTs) of ear specimens. To segment the intra-cochlear anatomy in a new patient's pre-operative CT, we fit the model to the external walls of the cochlea. Once the model is fitted it permits inferring the location of the intra-cochlear structures. Although this algorithm produces satisfactory results in most cases and we are using it on a regular basis to support clinical studies, sub-optimal segmentations can still be observed.

Deep learning-based methods, especially convolutional neural networks, have achieved impressive performances in a variety of image processing tasks. Notably, a 3d U-Net [9] has been proposed to produce 3d medical image dense segmentation when it is trained using a dataset in which only sparse delineation is available. Inspired by its successful application to image segmentation and landmark detection [10] since then, we propose to use it to segment the intra-cochlear anatomy. However, due to the large number of parameters in deep neural networks, training a model from scratch requires a large set of images for which the ground truth is known. For our application, such a dataset does not exist. This is because the intra-cochlear anatomy cannot be seen in clinical CTs. It can thus not be manually localized in these images. The only way to build a ground truth data set is to acquire μ CT images of temporal bone specimens and to delineate structures in these images. It is an extremely tedious process that cannot be done on a large scale.

To circumvent difficulties caused by the size of the training set required, we propose to first use the automatic segmentations generated by our ASM-based method on a large-scale image set as weak supervision to pre-train the deep learning model. Second, we fine-tune it using a small dataset for which accurate manual segmentation of the intra-cochlear anatomy is available. Similar concepts can be found in [11][12].

DATA

We have collected 346 clinical head CTs, each from one unique patient, obtained prior to surgery. The ST, SV and MD of each ear which is not implanted with a CI are segmented using the ASM-based method. For each ear, we crop a region of interest using a bounding box which includes the three structures with an additional 5mm-margin at each of the

anterior, posterior, superior, inferior, left and right directions. We denote this set as dataset #1. We also have another set that includes 11 clinical CT- μ CT image pairs. Each pair is obtained from the same ear specimen. The voxel dimensions are 0.3mm isotropic in the clinical CTs and are 36 μ m or 37.6 μ m isotropic in the μ CTs. In the μ CTs, one can visually identify the boundaries of the intra-cochlear anatomy. To create the ground truth manual segmentations from μ CT images, we proceed as follows: first, in each μ CT image, the three anatomical structures of interest are manually drawn as binary masks. Second, we generate a mesh for each structure from the segmentation masks, using the marching cubes algorithm [13]. Third, since the meshes created directly from binary masks do not have the same number of vertices across specimens, we establish point correspondence between different specimens. The point correspondence is important for our application because (1) creating the ASM requires point correspondence. And (2) for every new patient, frequency values need to be assigned to the neural regions of the segmentation meshes for CI programming purposes. Currently this is done by direct transfer from a reference specimen's segmentation meshes where the frequencies have already been defined and such transfer also requires point correspondence. Point correspondence is established in the following way: (1) a specimen and its manual segmentation meshes are chosen as the reference specimen and reference meshes, respectively. (2) For a new specimen (target specimen), we take its μ CT, perform a sequence of rigid and non-rigid registrations [14] from the reference specimen's μ CT to it and deform the reference meshes accordingly. To compensate for the imperfection of automatic registration, the deformed reference meshes are manually adjusted until they are well aligned with the target specimen meshes. (3) For each point on the reference meshes, the closest point on the target specimen's meshes is found and used as a new point to re-generate the target meshes. An example of the target meshes and the reference meshes before and after manual adjustment is shown in Figure 1.

After the ground truth segmentation masks and meshes are created in the μ CT image of the specimen, the μ CT is rigidly registered to its clinical CT counterpart and the manual segmentations are projected onto the clinical CT space via the same transformation matrix. We denote this dataset as dataset #2. An example of registered CT- μ CT image pair is shown in Figure 2. Dataset #2 consists of dataset #2A (5 specimens) and dataset #2B (6 specimens). The manual segmentation meshes in dataset #2A are used to build the ASM model which we use to do coarse segmentation in dataset #1, as well as to fine-tune the deep learning model. Dataset #2B is used to test our algorithm and to fine-tune the deep learning model in a leave-one-out way. Since the ASM-based method only generates meshes, for each image in both datasets, the meshes are converted to binary segmentation masks. All images and their segmentation masks are resampled to $0.1 \times 0.1 \times 0.1 \text{ mm}^3/\text{voxel}$.

METHODS

In this work, we use the 3d U-Net as the base network architecture. It is shown in Figure 3. The input volume size is set to $96 \times 96 \times 96$. The output has 4 channels, corresponding to the ST, SV, MD and the background. Suppose that $S^g \in \mathbf{B}^{96 \times 96 \times 96 \times 4}$ ($\mathbf{B} = \{0, 1\}$) is the binary segmentation mask generated either by the ASM-based method or manual delineation for training. $S^g_{i,j,k,n} = 1$ if voxel (i, j, k) is in the n^{th} structure and $S^g_{i,j,k,n} = 0$ if voxel (i, j, k) is

not in the n^{th} structure. Here, $n = 1, 2, 3, 4$ corresponds to the ST, SV, MD and background, respectively. $\hat{S} \in \mathbf{R}^{96 \times 96 \times 96 \times 4}$ is the probability map generated by the 3d U-Net. We use the following loss function [15]:

$$loss(S^g, \hat{S}) = 1 - \frac{2 \sum_{n=1}^4 w_n \sum_{i=1}^{96} \sum_{j=1}^{96} \sum_{k=1}^{96} (S_{i,j,k}^g \hat{S}_{i,j,k}^n)}{\sum_{n=1}^4 w_n \sum_{i=1}^{96} \sum_{j=1}^{96} \sum_{k=1}^{96} (S_{i,j,k}^g + \hat{S}_{i,j,k}^n)}$$

It measures the disagreement between two segmentation masks. Here, since the number of voxels in each structure and the background are highly imbalanced, we use a weight w_n ($n = 1, 2, 3, 4$) to ensure that the four classes contribute equally to the loss function. It is defined as

$$w_n = \frac{1}{\sum_{i=1}^{96} \sum_{j=1}^{96} \sum_{k=1}^{96} S_{i,j,k}^g}, \quad (n = 1, 2, 3, 4)$$

Since the image volumes vary in size, to fit them in the neural network and for data augmentation purpose, we crop 10 image cubes with a size of $96 \times 96 \times 96$ from each cropped region of interest at different locations as the input. We use a two-step training strategy. In step one, we use the images and segmentation masks in dataset #1 to pre-train the model of which the convolutional kernels are randomly initialized. In this step, 80% of the volumes are randomly selected as the training set and the remaining 20% as the validation set. In step two, we fine-tune the model using images and the corresponding manual segmentations in dataset #2A. Since dataset #2A is small, to leverage our ground truth segmentation data, as we have mentioned, we also use images in data set #2B for fine-tuning, in a leave-one-out way. That is to say, to test each image in dataset #2B, we fine-tune the coarse segmentation model using dataset #2A and all the other images in dataset #2B. During fine-tuning, only the last two convolutional layers are set to trainable. We split the volumes into a training set (80 volumes) and a validation set (20 volumes) in the same fashion as we do in step one. In both training steps, we set the batch size to 1 and use the Adam [16] optimizer with an initial learning rate of 0.0001. A flowchart of our two-level training scheme is shown in Figure 4.

In the testing phase, since the test volumes do not necessarily fit in the 3d U-Net, we use a sliding window with a stride of 10 voxels in each of the three orthogonal directions to crop the original test volumes and feed them to the network. The outputs are concatenated according to the spatial positions of the corresponding input cropped cubes. In overlapping regions, the responses are averaged.

In the segmentations produced by the network, noticeable false segmentations can happen. Specifically, random regions that are not part of the three anatomical structures can be classified as one of them. An example is shown in Figure 5. Since such regions are typically small, they are simply removed by keeping only the largest connected component, i.e., the true anatomy in the segmentation mask.

For comparison purpose, we segment each image in dataset#2B using the ASM-based method (the model is created with 9 training examples that include the 5 dataset #2A). We also do segmentations using the pre-trained (1-level) model to show the effectiveness of the 2-level training scheme. For evaluation, we use three criteria. First, we compute the DICE similarity coefficients (DSC) between the automatic segmentations and the ground truth. Second, we compute the average surface distance (ASD) between the automatic segmentations and the ground truth segmentations. Specifically, suppose two meshes M_1 and M_2 . For each point p on M_1 , its distance to M_2 is

$$d(p, M_2) = \inf_{q \in M_2} \|p - q\|_2$$

The mean distance from M_1 to M_2 is thus,

$$d(M_1, M_2) = \text{mean}_{p \in M_1} d(p, M_2)$$

We define the ASD between M_1 and M_2 as

$$ASD(M_1, M_2) = \frac{d(M_1, M_2) + d(M_2, M_1)}{2}$$

The smaller the ASD is with the ground truth, the better the segmentation quality is. Since the 3d U-Net generates image masks, we convert them into meshes to compute ASDs. Third, as we have discussed in the **DATA** section, because point correspondence between segmentation meshes of different subjects is important for frequency assignment and CI programming, we compute the point-to-point (P2P) errors. For the ASM-based method, we use the point correspondence encoded in the meshes. This cannot be done with the deep learning-generated meshes because point correspondence with the reference specimen mesh is not established. To establish point correspondence for these meshes, we use the ASM-generated meshes of the same specimen. We first perform an iterative closest point (ICP) registration between the two meshes. For each point on the transformed ASM-generated meshes, we find the closest point on the deep learning-generated meshes. Finally, the P2P distance is computed between the ground truth meshes and the automatic segmentation meshes for each specimen, i.e.,

$$P2P(M_a, M_g) = \text{mean}_{0 < i \leq V} \|p_a^i - p_g^i\|$$

Here, M_a and M_g represent the automatic segmentation mesh and the ground truth segmentation mesh, respectively. V is the number of vertices. p_a^i and p_g^i are the i^{th} point of M_a and M_g , respectively.

RESULTS

The DSCs, ASDs and P2P distances are shown in Table 1. As can be seen, in terms of DSC and ASD, the proposed 2-level trained model yields the best results and is substantially better than the ASM-based method and the 1-level trained model. For each method, the P2P errors are substantially larger than ASDs. The three methods have close P2P distances and the proposed method does better than the other two only by a narrow margin according to this measure. We provide an explanation for this using Figure 6. For a point P_g on the ground truth mesh, its corresponding points on the ASM-generated mesh and the deep learning-generated mesh are denoted as P_{asm} and P_{dl} respectively. The P2P error can be seen as an aggregation of the error along the normal direction, denoted as the “normal error” and the error in the tangent direction, denoted as “tangential error”. The tangential error is due to the shape rotation happening in the ASM algorithm, making the P2P error substantially larger than APD, even if the rotation is slight. Since point correspondence for the deep learning-generated meshes is established using the ASM-generated meshes, they have similar tangential errors. P2P errors are substantially larger than ASD errors for the ASM method, which indicates that the tangential error is the dominant error component for this approach. Tangential error caused by the ASM cannot be eliminated from the P2P deep learning method, which explains why the P2P error difference between ASM and the proposed method is not as large as the difference observed with DSC and APD. In Table 2, we show the p values of paired t tests when comparing different methods in segmenting each structure. For the DSCs and APDs the proposed method is statistically significantly different from the ASM method and the 1-level model.

A representative example illustrating the point-wise distances from the ground truth mesh to the automatic segmentations is shown in Figure 7. An example showing the segmentations superimposed on the image is shown in Figure 8. As can be seen, the proposed method produces a segmentation that is closer to the ground truth than the ASM-based method and the 1-level model. Areas in which the proposed method produces noticeably better results are encircled.

CONCLUSIONS

In this work, we propose to use a two-level training strategy to accurately segment the intra-cochlear anatomy in ear specimen clinical CTs using a very limited set of training images for which manually delineated ground truth segmentations are available. To circumvent the training data paucity problem, we use automatically generated segmentations obtained with our previous ASM-based method to train a weak segmentation model and use a small dataset for which ground truth segmentations are available to fine-tune it. The results show that the proposed method is more accurate than the ASM-based method on this data set in terms of DICE and ASD. To assess the robustness of the proposed method, we are currently conducting a large-scale evaluation study in which the ASM-based approach and the proposed method will be compared on clinical CTs.

ACKNOWLEDGMENTS

This study is conducted under the support of NIH grants R01DC014037, R01DC008408, R01DC014462 and Advanced Computing Center for Research and Education (ACCRE) of Vanderbilt University. We are grateful for the support of NVIDIA Corporation for providing us with a GPU used to support this work. We would like to thank Yubo Fan, my colleague in Medical Image Processing lab, Vanderbilt University, for manually doing mesh point correspondence. The content is solely the responsibility of the authors and does not necessarily represent the official views of this institute.

REFERENCES

- [1]. Noble JH, Labadie RF, Gifford RH and Dawant BM, "Image-guidance enables new methods for customizing cochlear implant stimulation strategies." *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 215 (2013): 820–829. [PubMed: 23529109]
- [2]. Noble JH, Labadie RF, Majdani O, and Dawant BM, "Automatic segmentation of intracochlear anatomy in conventional CT," *IEEE Transactions on Biomedical Engineering*, vol. 58, pp. 2625–32, 2011.
- [3]. Noble JH and Dawant BM, "Automatic graph-based localization of cochlear implant electrodes in CT," *Medical Image Computing and Computer Assisted Intervention*, pp. 152–159, 2015. [PubMed: 27158686]
- [4]. Zhao Y, Dawant BM, Labadie RF, and Noble JH, "Automatic localization of cochlear implant electrodes in CT," *Medical Image Computing and Computer Assisted Intervention*, pp. 331–338, 2014. [PubMed: 25333135]
- [5]. Zhao Y, Dawant BM, and Noble JH, "Automatic localization of cochlear implant electrodes in CTs with a limited intensity range", *Proc. SPIE 10133, Medical Imaging 2017: Image Processing*, 101330T, 2017
- [6]. Noble JH, Gifford RH, Hedley-Williams AJ, Dawant BM and Labadie RF. "Clinical evaluation of an image-guided cochlear implant programming strategy." *Audiology and Neurotology* 196 (2014): 400–411 [PubMed: 25402603]
- [7]. Noble JH, Hedley-Williams AJ, Sunderhaus L, Dawant BM, Labadie RF, Camarata SM and Gifford RH. "Initial results with image-guided cochlear implant programming in children." *Otology & Neurotology*, 372 (2016): e63. [PubMed: 26756157]
- [8]. Labadie RF, Noble JH, Hedley-Williams AJ, Sunderhaus LW, Dawant BM, and Gifford RH, "Results of Postoperative, CT-based, Electrode Deactivation on Hearing in Prelingually Deafened Adult Cochlear Implant Recipients," *Otology & Neurotology*, vol. 37, pp. 137–45, 2016. [PubMed: 26719955]
- [9]. Çicek O, Abdulkadir A, Lienkamp SS, Bronx T and Ronneberger O. "3D U-Net: learning dense volumetric segmentation from sparse annotation." *Medical Image Computing and Computer Assisted Intervention*, pp. 424–432, 2016.
- [10]. Zhang D, Wang J, Noble JH, Dawant BM. "Accurate Detection of Inner Ears in Head CTs Using a Deep Volume-to-Volume Regression Network with False Positive Suppression and a Shape-Based Constraint". *Medical Image Computing and Computer Assisted Intervention*, pp. 703–711, 2018
- [11]. Mahajan D, Girshick R, Ramanathan V and He K. "Exploring the Limits of Weakly Supervised Pretraining." *arXiv preprint arXiv:1805.00932*(2018).
- [12]. Huo Y, Xu Z, Abound K, Parvathaneni P, Bao S, Bermudez C, Resnick SM, Cutting LE, Landman BA, et al. "Spatially Localized Atlas Network Tiles Enables 3D Whole Brain Segmentation from Limited Data". *Medical Image Computing and Computer Assisted Intervention*, pp. 698–705. 2018
- [13]. Lorensen WE and Cline HE. "Marching cubes: A high resolution 3D surface construction algorithm." *ACM siggraph computer graphics*. Vol. 21 No. 4 ACM, 1987
- [14]. Rohde GK, Aldroubi A, and Dawant BM. "The adaptive bases algorithm for intensity-based non-rigid image registration." *IEEE transactions on medical imaging* 2211 (2003): 1470–1479. [PubMed: 14606680]

- [15]. Sudre CH, Li W, Vercauteren T, Ourselin S, and Cardoso MJ. "Generalised Dice overlap as a deep learning loss function for highly unbalanced segmentations". In Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support (pp. 240–248). Springer, Cham 2017
- [16]. Kingma DP, and Ba J. "Adam: A method for stochastic optimization." arXiv preprint arXiv: 1412.6980(2014).

Figure 1.
Target ground truth meshes and reference meshes: (a) after rigid and non-rigid registration and (b) after manual adjustment

Figure 2.

A pair of registered clinical CT image and μ CT image in axial view. Manual delineation is shown.

Figure 3.
The neural network architecture we employ

Figure 4.
A flowchart of our proposed algorithm

Figure 5.
An example of wrong segmentation

Figure 6.

A sketch of the surface distances and P2P distances between the ground truth mesh and the automatically-generated meshes: e_{asm}^n and e_{dl}^n are the normal errors for the ASM and the deep learning method, respectively. e^t is the tangential error. e_{asm} and e_{dl} are the P2P errors of the ASM and the deep learning method, respectively.

Figure 7.

An example of surface distances from the ground truth meshes to the automatic segmentation meshes (in *mm*), displayed on the ground truth

Figure 8.

An example of the results obtained with the three methods. Areas with a noticeable difference are encircled.

Table 1.

DSCs, ASDs and P2P distances of the segmentations using the three methods

	ST			SV			MD		
	ASM	1-level	Proposed	ASM	1-level	Proposed	ASM	1-level	Proposed
Mean DSC	0.76	0.77	0.87	0.75	0.71	0.86	0.60	0.63	0.83
Std of DSC	0.012	0.025	0.016	0.019	0.058	0.017	0.049	0.026	0.017
Mean ASD *	0.15	0.14	0.077	0.14	0.15	0.084	0.21	0.18	0.09
Std of ASD	0.027	0.017	0.016	0.021	0.028	0.017	0.039	0.01	0.027
Mean of P2P	0.55	0.58	0.52	0.56	0.59	0.54	0.58	0.57	0.53
Std of P2P	0.2	0.22	0.23	0.22	0.25	0.25	0.13	0.13	0.14

*: All distance values in this table are in *mm*

Table 2.

Values of p using paired t tests for different comparisons.

	DSC			ASD			P2P		
	ST	SV	MD	ST	SV	MD	ST	SV	MD
Proposed vs ASM	0.0002	0.0002	0.0002	0.014	0.012	0	0.052	0.083	0.059
1-level vs ASM	0.13	0.077	0.17	0.68	0.12	0.26	0.013	0.055	0.078
1-level vs proposed	0 *	0.0001	0.0001	0	0.0001	0.0001	0.0003	0.0003	0.12

*. Any value that is below 0.0001 is shown as 0. A value in bold indicates statistical significance (<0.05)