Nuclei Segmentation of Fluorescence Microscopy Images Using Three Dimensional Convolutional Neural Networks

David Joon Ho
Purdue University
West Lafayette, Indiana

Chichen Fu
Purdue University
West Lafayette, Indiana

Paul Salama
Indiana University-Purdue University
Indianapolis, Indiana

Kenneth W. Dunn
Indiana University
Indianapolis, Indiana

Edward J. Delp
Purdue University
West Lafayette, Indiana

Abstract

Fluorescence microscopy enables one to visualize sub-cellular structures of living tissue or cells in three dimensions. This is especially true for two-photon microscopy using near-infrared light which can image deeper into tissue. To characterize and analyze biological structures, nuclei segmentation is a prerequisite step. Due to the complexity and size of the image data sets, manual segmentation is prohibitive. This paper describes a fully 3D nuclei segmentation method using three dimensional convolutional neural networks. To train the network, synthetic volumes with corresponding labeled volumes are automatically generated. Our results from multiple data sets demonstrate that our method can successfully segment nuclei in 3D.

1. Introduction

Fluorescence microscopy, a type of an optical microscopy, enables three dimensional visualization of subcellular structures of living specimens [1]. This is especially true for two-photon microscopy using near-infrared light which can image deeper into tissue. Since two-photon microscopy uses two photons simultaneously with less energy to excite electrons on fluorescent molecules, living samples may receive less damage from the data acquisition process [2, 3]. This allows acquisition of large amount of 3D data. Manually analyzing large image volumes requires a large amount of work and can be biased by individuals. Automatic 3D microscopy image analysis methods, particularly segmentation, are required for efficiency and accuracy to quantify biological structures.

Several segmentation methods on microscopy images have been developed in the past years. In [4] the use of active contours is described. Active contours has been widely investigated in microscopy image segmentation because it can segment structures with various shapes [5]. Contour initialization is important since final segmentation results and convergence time are highly depending on the initial contours. Initialization can be done manually but requires a large amount of time and effort for 3D image data sets. In [6], an automatic initialization technique was described by estimating the external energy field from the Poisson inverse gradient to generate better initial contours for noisy images. With automatic initialization, [7] developed a 3D active surface method, an extension of the Chan-Vese 2D region-based active contour model [8], to segment 3D cellular structures of a rat kidney. In [9] 3D active surfaces with correct for inhomogeneity is presented. A method known as Squassh [10, 11] minimizes an energy functional derived from a generalized linear model to segment and quantify 2D or 3D subcellular structures. These methods do not separate overlapping nuclei which may produce poor segmentation results.

In order to separate overlapping nuclei and count the number of nuclei in a volume several methods have been reported. In [12] a fully automatic segmentation method using coupled active surfaces to separate overlapping cells by multiple level set functions with a penalty term and a constraint of conserving volume is described. Here, it is assumed that volumes of cells are approximately constant. In [13] an approach which improves on the technique presented in [12] which used watershed techniques, a non-PDE-based energy minimization, and the Radon transform to separate touching cells is presented. Alternatively, [14] described model-based nuclei segmentation using primitives corresponding to nuclei boundaries and delineating nuclei using region growing. In [15] a discrete multi-region competition that can generate multiple labels to count the number of structures in a volume is described. Recently, [16] used midpoint analysis, a distance function optimiza-

tion for shape-fitting, and marked point processes (MPP) to segment and count nuclei in fluorescence microscopy images. These techniques have no ability to distinguish nuclei and other biological structures.

Segmenting nuclei and distinguishing them from different subcellular structure may be possible with convolutional neural networks (CNN) if the networks are trained with many training images where the nuclei are manually segmented providing ground truth image volumes. Convolutional neural networks have been known for many years [17, 18]. The first successful application using a CNN is LeNet [19] for hand-written digits recognition. In [20] the rectified linear unit (ReLU) is described to achieve the best results on the ImageNet classification benchmark. CNNs are widely used in many segmentation problems such as [21]. To segment neuron membranes for electron microscopy images, [22] developed a segmentation technique that uses a CNN structure with a max-pooling operation. Note that the max-pooling operation preserves features while downsampling the feature maps. In [23] the detection of Tyrosine Hydroxylase-containing cells in zebrafish brain images from wide-field microscopy using a convolutional neural network with a support vector machine (SVM) [24] to preselect training candidates is investigated. U-Net from [25] uses two-dimensional convolutional neural networks with an encoder-decoder architecture to segment neuronal structures in electron microscopy and cells in light microscopy. Note that an elastic deformation technique was used to generate more training images from a limited number of ground truth images. In [26] nuclei segmentation method on histopathology images using deep CNN with selection-based sparse shape model is described. While producing good results in 2D images, these techniques cannot utilize the z-directional (or depth) information in a volume.

Instead of examining two-dimensional segmentation using a 2D CNN, segmentation techniques using a "2D$^+$ CNN" has been introduced where 2D$^+$ CNN is semi-3D segmentation using 2D CNN architectures. In [27] a CNN model that combines three 2D CNNs that were trained in horizontal, frontal, and sagittal planes independently is used to segment 3D biological structures. In [28] the method described in [27] is simplified by using three orthogonal planes as three channel images. A similar method was introduced in [29] with a refinement process to utilize 3D information of a volume by a majority voting technique. Although [27, 28, 29] achieved good segmentation results, a 2D$^+$ CNN cannot fully incorporate 3D information. In [30], a volumetric segmentation method, 3D U-Net, was developed to generate 3D dense maps by expanding the 2D U-Net architecture [25]. Although [30] claimed to implement a 3D CNN architecture, the system was trained with manually annotated 2D slices, which does not fully use 3D infor-

mation in an image volume. Expanding a 2D CNN to 3D CNN is not a simple task because 3D ground truth volumes in biomedical data are extremely limited.

One of the most daunting problems in using CNNs is the very large amount of training images required. One way to address this problem is through the use of data augmentation methods where linear and nonlinear transforms are done on the training data to create "new" training images. Typical transformations include spatial flipping, warping and other deformations [25, 29] or contrast transformation [29]. These methods may help, but there is a still limitation if there are only a few 3D ground truth image volumes.

In this paper, we present a "fully" 3D CNN architecture to segment nuclei in fluorescence microscopy volumes. Instead of training a CNN with 2D ground truth images or augmented volumes from limited real 3D ground truth volumes, we generate a set of synthetic microscopy volumes and synthetic ground truth volumes containing multiple nuclei. To evaluate the results of our method, manually generated ground truth volumes on real fluorescence microscopy volumes are used. Image volumes as our testing data are collected using two-photon microscopy from a rat kidney labeled with Hoechst 33342.
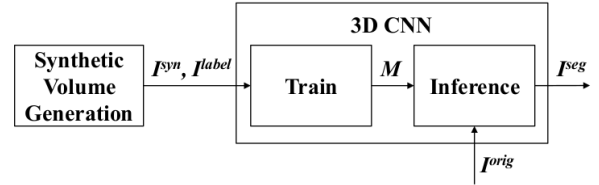
2. Proposed Method

Figure 1. Block diagram of the proposed method for 3D nuclei segmentation

Figure 1 shows a block diagram of our proposed method to segment nuclei in three dimension. We denote a 3D image volume of size $X \times Y \times Z$ by I and the pth focal plane image along the z-direction, of size $X \times Y$, by I_{z_p}, where $p \in \{1, \ldots, Z\}$. For example, $I_{z_{67}}^{orig}$ is the 67th original image. Also, we denote a sub-volume of I, whose x-coordinate is $q_i \leq x \leq q_f$, y-coordinate is $r_i \leq y \leq r_f$, z-coordinate is $p_i \leq z \leq p_f$, by $I_{(q_i:q_f,r_i:r_f,p_i:p_f)}$, where $q_i \in \{1, \ldots, X\}$, $q_f \in \{1, \ldots, X\}$, $r_i \in \{1, \ldots, Y\}$, $r_f \in \{1, \ldots, Y\}$, $p_i \in \{1, \ldots, Z\}$, $p_f \in \{1, \ldots, Z\}$, $q_i \leq q_f$, $r_i \leq r_f$, and $p_i \leq p_f$. For example, $I_{(241:272,241:272,131:162)}^{seg}$ is a sub-volume of a segmented volume, I^{seg}, where the sub-volume is cropped between 241st slice and 272nd slice in x-direction, between 241st slice and 272nd slice in y-direction, and between 131st slice and 162nd slice in z-direction.

Synthetic volumes containing nuclei, I^{syn}, with their corresponding labeled volumes (synthetic ground truth vol-

umes), I^{label}, are first randomly generated as described below. The synthetic volumes and labeled volumes then used to train a 3D CNN, M. Finally, the 3D CNN is used to segment nuclei in fluorescence microscopy images volumes, I^{orig}. The segmented image volume is denoted by I^{seg}. Note that Voxx [31] is used for 3D visualization in this paper.
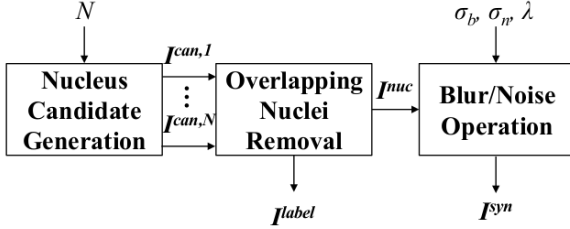
2.1. Synthetic Volume Generation

Figure 2. Block diagram for Synthetic Volume Generation used generate synthetic 3D image volumes, I^{syn}, with their corresponding labeled volumes, I^{label}

Figure 2 is a block diagram of the Synthetic Volume Generation stage. As we noted earlier, one of the major challenges in segmentation of biomedical images/volumes using a CNN is the lack of training images/volumes [18]. Manual segmentation to generate ground truth, especially in three dimension, is inefficient and laborious. Our approach is to use synthetic image volumes to train the CNN.

In synthetic image volumes, voxels belonging to nuclei which we are interested in segmenting are defined as foreground and other voxels are defined as background. We assume that nuclei can be modeled as an ellipsoid. Let N be the number of nucleus candidates. The jth nucleus candidate, $I^{can,j}$, is first generated where $1 \leq j \leq N$, with randomly generated lengths of the semi-axes, translation, rotation, and intensity of the ellipsoid. Then for each j, $I^{can,j}$ is included in the nuclei synthetic image volume, I^{nuc}. Here, I^{nuc} is initialized to a volume with all voxel intensities to zero, unless it overlaps with other nucleus candidates. Note that I^{nuc} is composed of non-overlapping nuclei with randomly assigned nonzero intensities and voxel intensities in background are zero. From I^{nuc}, a labeled volume, I^{label}, is generated as a binary volume where foreground voxels (non-overlapping nuclei) are 1 and background voxels are 0. Fluorescence microscopy volumes are blurred and noisy due to the point spread function from microscope and noise from the detector [32, 12, 33]. We observe that the background in florescence microscopy volumes are not completely dark. To make our synthetic image volume resemble a fluorescence microscopy volume, we first increase the voxel values in the background to a randomly generated non-zero value. The volume is then filtered using a Gaus-

sian filter with standard deviation, σ_b is blur it and then Gaussian noise with standard deviation, σ_n, and a Poisson noise with mean, λ, are added to generate the final synthetic image volume, I^{syn}.

The synthetic image volumes will be used to train our 3D CNN. If the volumes are too small, not enough information can be learned. If the volumes are too big, the training time will take long. Therefore, we select the size of the synthetic volumes to be $64 \times 64 \times 64$. More detail is provided below.

2.1.1 Nucleus Candidate Generation

In order to generate a synthetic volume with multiple nuclei, we first generate a single nucleus which can be potentially included in the synthetic volume. We first assume that nuclei are ellipsoidal shape. To train our CNN with various synthetic nuclei, we generate nuclei which have random size, are located in random positions, are oriented in random directions, and have random intensity. In this step, N nucleus candidates are generated in a volume of size $64 \times 64 \times 64$.

To generate the jth nucleus candidate, where $1 \leq j \leq N$, we first find the translated and rotation coordinates, $(\tilde{x}, \tilde{y}, \tilde{z})$, from the original coordinates, (x, y, z) with a random translation vector, $\mathbf{t} = (t_x, t_y, t_z)$, and a random rotation vector, $\mathbf{r} = (r_x, r_y, r_z)$. Here, t_x, t_y, t_z, r_x, r_y, r_z are the translations in x, y, and z-direction and the rotations around x, y, and z-axes, respectively.

$$\begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \end{bmatrix} = R_z(r_z)R_y(r_y)R_x(r_x) \begin{bmatrix} x - t_x \\ y - t_y \\ z - t_z \end{bmatrix} \quad (1)$$

Here, $R_x(\theta)$, $R_y(\theta)$, $R_z(\theta)$ are rotation matrices around the x, y, z-axes, respectively, with an angle θ.

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(-\theta) & -\sin(-\theta) \\ 0 & \sin(-\theta) & \cos(-\theta) \end{bmatrix} \quad (2)$$

$$R_y(\theta) = \begin{bmatrix} \cos(-\theta) & 0 & \sin(-\theta) \\ 0 & 1 & 0 \\ -\sin(-\theta) & 0 & \cos(-\theta) \end{bmatrix} \quad (3)$$

$$R_z(\theta) = \begin{bmatrix} \cos(-\theta) & -\sin(-\theta) & 0 \\ \sin(-\theta) & \cos(-\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4)$$

The jth nucleus candidate, $I^{can,j}$, is then generated with random length of semi-axes, $\mathbf{a} = (a_x, a_y, a_z)$ and random intensity, i:

$$I^{can,j} = \begin{cases} i, & \text{if } \frac{\tilde{x}^2}{a_x} + \frac{\tilde{y}^2}{a_y} + \frac{\tilde{z}^2}{a_z} < 1 \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

In this paper, N is set to be 100. For each j, with uniform distribution, **a** is randomly selected between 4 and 6, **t** is randomly selected between 1 and 64, **r** is randomly selected between 1 and 360, and i is randomly selected between 200 and 255. Those parameters are set based on nuclei in original microscopy volumes.

2.1.2 Overlapping Nuclei Removal

A synthetic volume with multiple nuclei, I^{nuc}, can now be generated by adding N nuclei candidate volumes, $I^{can,j}$, generated in the previous step. However, in a biological structure, no nuclei are physically overlapping. So it is necessary to remove nuclei overlapping with other nuclei.

First of all, I^{nuc} is initialized to zeros with the size of $64 \times 64 \times 64$. Note that I^{nuc} is initialized to be all background (non-nuclei region). For $1 \leq j \leq N$, in a sequential order, the jth single nucleus candidate, $I^{can,j}$, would be added in the synthetic volume, I^{nuc}, if there is no intersection between foreground region in $I^{can,j}$ and foreground region in I^{nuc}. However, $I^{can,j}$ would not be added in I^{nuc} if $I^{can,j}$ have intersection between foreground region in $I^{can,j}$ and foreground region in I^{nuc}. After this step, no nuclei will overlap to other nuclei in I^{nuc}.

A labeled volume (a synthetic ground truth volume), I^{label}, corresponding to the synthetic volume is generated by assigning 1 to foreground voxels and 0 to background voxels:

$$I^{label}(x,y,z) = \begin{cases} 1, & \text{if} \quad I^{nuc}(x,y,z) \neq 0 \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

2.1.3 Blur and Noise

When images of specimens are acquired from fluorescence microscope, images are degraded by blurring and noise. First of all, the point spread function (PSF) from microscope causes blurring [32]. Additionally, fluorescence microscopy images contain a combination of Gaussian noise and Poisson noise because only a limited number of photons are detected in the detector of microscope due to photobleaching, low fluorophore concentration, and short exposure time [12, 33]. Therefore, it is necessary to include blur and noise in our generate synthetic image volumes.

First, background voxel intensities of I^{nuc} are set to b because voxel intensities of the background region for real fluorescence microscopy images are not completely zero. We let b be randomly selected between 50 and 100 with an uniform distribution. We then use a simple synthetic PSF to blur the volumes where we assume the PSF is a normalized Gaussian filter with window size of $5 \times 5 \times 5$ and standard deviation of $\sigma_n = 20$. Lastly, zero-mean Gaussian noise with $\sigma_n = 5$ and Poisson noise with $\lambda = 5$ are added to the

blurred volume to generate I^{syn}. Figure 3 shows three examples of synthetic image volumes with their corresponding labeled volumes. Figure 4 compares two original image volumes at various depths and a synthetic image volume with the size of $64 \times 64 \times 64$. The size of the nuclei, the intensity of the nuclei, and the background noise of synthetic image volumes are close to original image volumes.
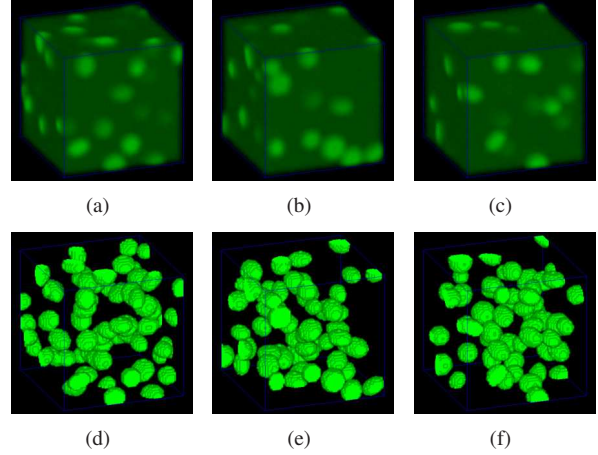

Figure 3. Examples of synthetic image volumes and their labeled volume with one sub-volume of real data (a) $I^{syn,1}$, (b) $I^{syn,2}$, (c) $I^{syn,3}$, (d) $I^{gt,1}$, (e) $I^{gt,2}$, (f) $I^{gt,3}$
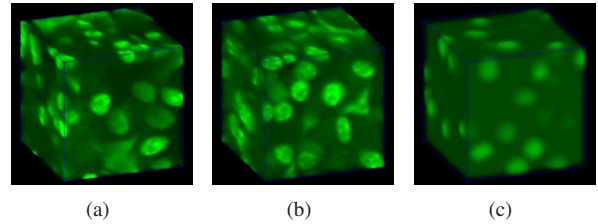

Figure 4. Comparison between original volumes and a synthetic volume (a) $I^{orig}_{(225:288,225:288,71:134)}$, (b) $I^{orig}_{(225:288,225:288,171:234)}$, (c) $I^{syn,1}$
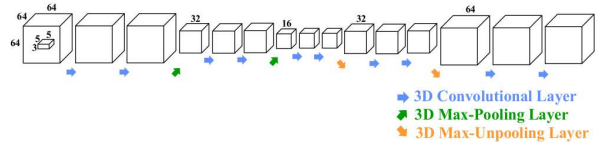
2.2. 3D CNN Training

Figure 5. Architecture of our 3D CNN

Figure 5 shows our architecture of a 3D CNN with an encoder-decoder structure. Each 3D convolutional layer consists of a convolutional operation with a $5 \times 5 \times 3$ kernel with $2 \times 2 \times 1$ voxel padding followed by 3D batch normalization [34] and a rectified-linear unit (ReLU) activation function. The kernel size is chosen accordingly be-

cause the resolution along the z-direction is smaller than along the x and y-directions. Voxel padding is used to maintain the same volume size during the convolutional operation. A 3D max-pooling layer uses $2 \times 2 \times 2$ window with stride of 2 to preserve feature information while proceeding to the deep of the architecture. In the decoder, a 3D max-unpooling layer is used to retrieve feature information according to the indices that saved in the corresponding 3D max-pooling layer. An input volume to the network is a single channel volume with size of $64 \times 64 \times 64$ and a 3D voxelwise classification with size of $64 \times 64 \times 64$ is generated as an output volume of the network. Our architecture, M, is implemented in Torch [35]. To train our model, M, stochastic gradient descent (SGD) with a fixed learning rate of 10^{-6} and a momentum of 0.9 is used. 100 pairs of synthetic image volumes, I^{syn}, and labeled image volumes (ground truth volumes for synthetic image volumes), I^{label}, are used to train the model. For each iteration, a randomly-selected training volume is used to train M.
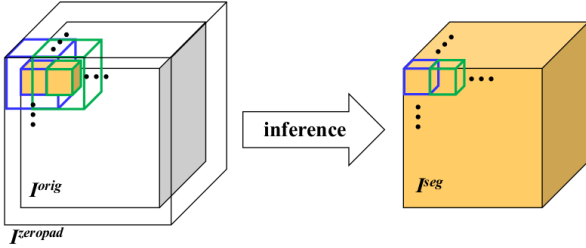
2.3. 3D CNN Inference

Figure 6. Inference

Our 3D CNN, M, segments a sub-volume of size of $64 \times 64 \times 64$. While cropping the sub-volume from the original volume, some nuclei on the boundary of the sub-volume may be partially included and lose their shape information. This may lead to incomplete segmentation near the boundary of the sub-volume. To avoid this, we use the central sub-volume of the output volume with size of $32 \times 32 \times 32$ to make sure nuclei in the output volume are completely segmented using their entire information. Additionally, if X, Y, or Z of I^{orig} is greater than 64, it is necessary to slide a 3D window with size of $64 \times 64 \times 64$ to segment the entire I^{orig}.

Figure 6 describes our method for 3D CNN inference. In order to have I^{seg} the same size as I^{orig}, we zero-pad I^{orig} by 16 voxels on boundaries, denoted as $I^{zeropad}$. In this case, the size of $I^{zeropad}$ would be $(X + 32) \times (Y + 32) \times (Z + 32)$. Placing a 3D window on the top, left, frontal corner of $I^{zeropad}$ (see blue window in Figure 6), $I^{zeropad}_{(1:64,1:64,1:64)}$ becomes the input volume of the 3D CNN, M, with size of $64 \times 64 \times 64$ to generate $I^{seg}_{(1:32,1:32,1:32)}$, a sub-volume of I^{seg} on the top, left, frontal corner with

size of $32 \times 32 \times 32$. Next, the 3D window is slided to x-direction by 32 voxels (see green window in Figure 6), then $I^{zeropad}_{(33:96,1:64,1:64)}$ becomes the next input volume of the 3D CNN, M, to generate $I^{seg}_{(33:64,1:32,1:32)}$, a sub-volume slided to x-direction by 32 voxels from the previous sub-volume, $I^{seg}_{(1:32,1:32,1:32)}$. This operation continues to x, y, and z-direction until the entire volume is processed.

3. Experimental Results

We tested our method on four different rat kidney data sets. All data sets consist of grayscale images of size $X = 512 \times Y = 512$. Data-I consists of $Z = 512$ images, Data-II of $Z = 36$, Data-III of $Z = 41$, and Data-IV of $Z = 23$ images. Figure 7 shows the segmented images on Data-I located at various depths.
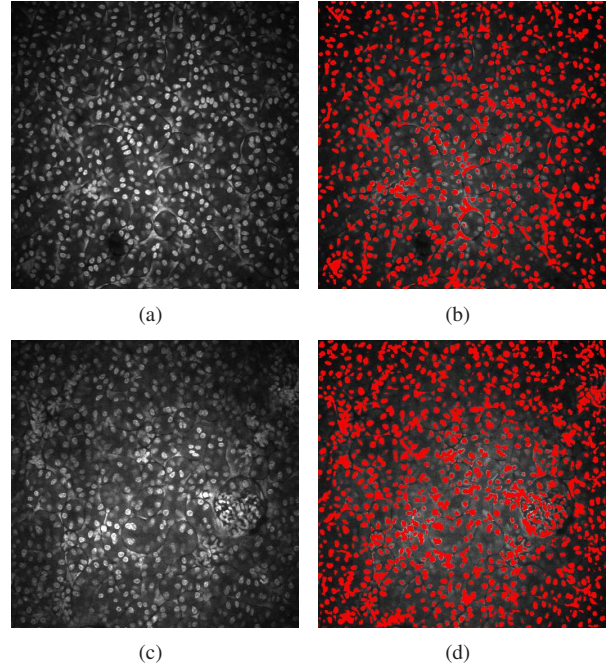

Figure 7. Original images and their segmented images of Data-I using the proposed method in different depth (a) $I^{orig}_{z_{67}}$, (b) $I^{seg}_{z_{67}}$, (c) $I^{orig}_{z_{230}}$, (d) $I^{seg}_{z_{230}}$

Our method was compared to other segmentation methods used in microscopy images including 3D active surface [7], 3D active surface with inhomogeneity correction [9], 3D Squassh [10, 11], and 2D$^+$ convolutional neural network (2D$^+$ CNN) [29]. In order to evaluate our method, we used three 3D ground truth sub-volumes of Data-I, $I^{gt}_{(241:272,241:272,31:62)}$, $I^{gt}_{(241:272,241:272,131:162)}$, $I^{gt}_{(241:272,241:272,231:262)}$ in different depth with size of $32 \times 32 \times 32$. Here, the ground truth volumes for evaluation are manually generated from a real microscopy data set. Figure 8, Figure 9, Figure 10 are the 3D visualization of various segmentation method results on different sub-volumes,

helped by Voxx [31].

It was observed that 3D active surfaces had poor results as sub-volumes are acquired deeper into tissue because voxel intensities on nuclei gets dimmer and more blurred. This issue was resolved by counting inhomogeneity correction, yet it has no ability to distinguish between nuclei and other subcellular structures. Squassh also failed to distinguish nuclei and other structures. Although 2D$^+$ CNN produced good results, discontinuity may be observed between planes because it has not utilized all three dimensional information. However, 3D CNN can segment in ellipsoidal shape which is close to the shape of nuclei. Note that the segmentation results are generated in a short running time (2∼3 minutes) using NVIDIA's GeForce GTX Titan X without any manually generated ground truth volumes to train the network.

(a)   (b)   (c)
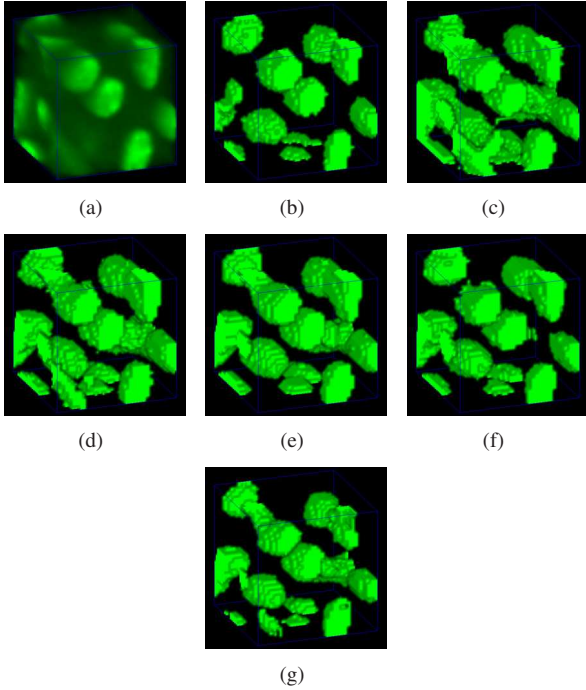
(d)   (e)   (f)

(g)

Figure 8. 3D visualization of $I_{(241:272,241:272,31:62)}$ of Data-I using Voxx [31] (a) original volume (b) 3D ground truth volume, (c) 3D active surfaces from [7], (d) 3D active surfaces with inhomogeneity correction from [9], (e) 3D Squassh from [10, 11], (f) 2D$^+$ CNN from [29], (g) proposed method

All segmentations were evaluated using 3D ground truth volumes based on the accuracy, Type-I error, and Type-II error metrics. Here, accuracy $= \frac{n_{TP}+n_{TN}}{n_{total}}$, Type-I error $= \frac{n_{FP}}{n_{total}}$, Type-II error $= \frac{n_{FN}}{n_{total}}$, where n_{TP}, n_{TN}, n_{FP}, n_{FN}, n_{total} are defined to be the number of true-positives (voxels correctly segmented as nuclei), true-negatives (voxels correctly segmented as background), false-positives (voxels wrongly segmented as nuclei), false-negatives (voxels wrongly segmented as background), and the total number of voxels in

(a)   (b)   (c)
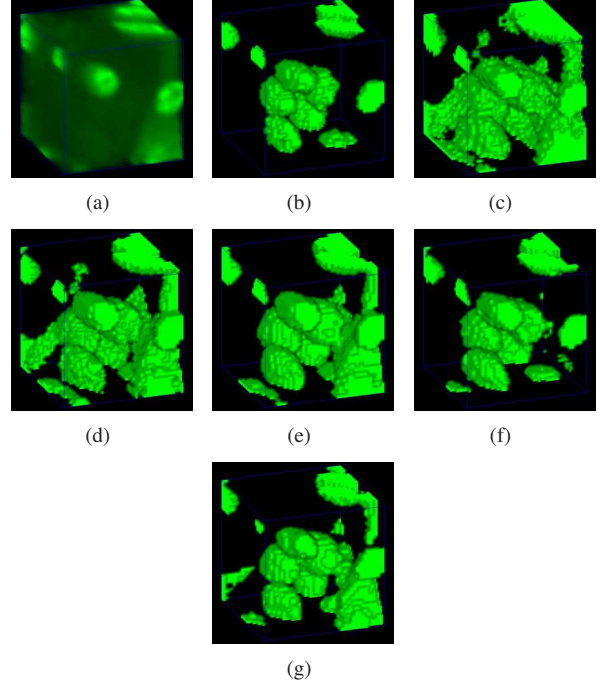
(d)   (e)   (f)

(g)

Figure 9. 3D visualization of $I_{(241:272,241:272,131:162)}$ of Data-I using Voxx [31] (a) original volume (b) 3D ground truth volume, (c) 3D active surfaces from [7], (d) 3D active surfaces with inhomogeneity correction from [9], (e) 3D Squassh from [10, 11], (f) 2D$^+$ CNN from [29], (g) proposed method

a volume, respectively. Table 1, Table 2, Table 3 shows the accuracy for various segmentation methods and the proposed method on different sub-volumes.

As shown in the 3D visualization from Figure 8, Figure 9, Figure 10, and accuracy test from Table 1, Table 2, Table 3, our 3D CNN achieved similar results "without" any ground truth volumes from Data-I during training. Training with synthetic volumes can be extremely helpful for automatic segmentation due to the difficulty of manually generating ground truth volumes in biomedical data sets.

Table 1. Accuracy, Type-I and Type-II errors for known methods and our method on $I_{(241:272,241:272,31:62)}$ of Data-I

	Acc.	Type-I	Type-II
Method [7]	84.09%	15.68%	0.23%
Method [9]	87.36%	12.44%	0.20%
Method [10, 11]	90.14%	9.07%	0.79%
Method [29]	94.25%	5.18%	0.57%
Proposed Method	92.20%	5.38%	2.42%

Our method can successfully segment nuclei from different rat kidney data (Data-II, Data-III, Data-IV). Since the size of nuclei in Data-II, Data-III, and Data-IV is smaller than the size of nuclei in Data-I, we used **a**, length of semiaxes of a synthetic nucleus, randomly generated between 2 and 3. Figure 11 shows the segmented images on different
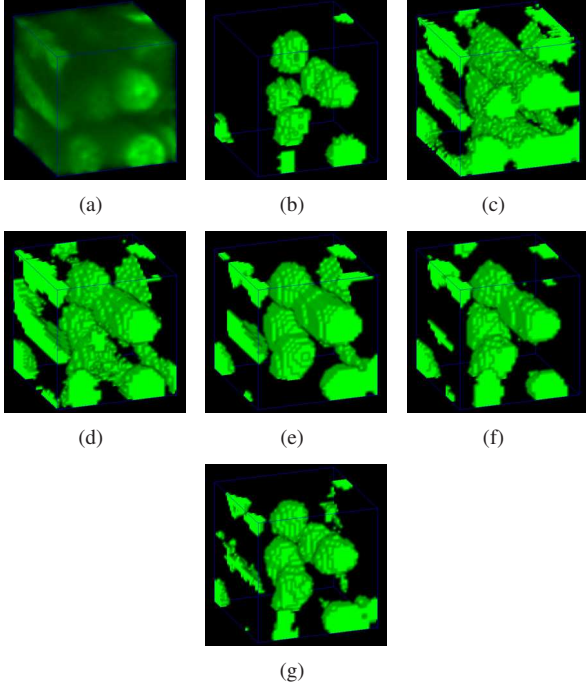
Figure 10. 3D visualization of $I_{(241:272,241:272,231:262)}$ of Data-I using Voxx [31] (a) original volume (b) 3D ground truth volume, (c) 3D active surfaces from [7], (d) 3D active surfaces with inhomogeneity correction from [9], (e) 3D Squassh from [10, 11], (f) 2D$^+$ CNN from [29], (g) proposed method

Table 2. Accuracy, Type-I and Type-II errors for known methods and our method on $I_{(241:272,241:272,131:162)}$ of Data-I

	Acc.	Type-I	Type-II
Method [7]	79.25%	20.71%	0.04%
Method [9]	86.78%	13.12%	0.10%
Method [10, 11]	88.26%	11.67%	0.07%
Method [29]	95.24%	4.18%	0.58%
Proposed Method	92.32%	6.81%	0.87%

Table 3. Accuracy, Type-I and Type-II errors for known methods and our method on $I_{(241:272,241:272,231:262)}$ of Data-I

	Acc.	Type-I	Type-II
Method [7]	76.44%	23.55%	0.01%
Method [9]	83.47%	16.53%	0.00%
Method [10, 11]	87.29%	12.61%	0.10%
Method [29]	93.21%	6.61%	0.18%
Proposed Method	94.26%	5.19%	0.55%

data sets.

4. Conclusions

In this paper we presented a nuclei segmentation method that uses a fully 3D convolutional neural network. The training volumes were generated using synthetic data. The experimental results show that our method can accurately
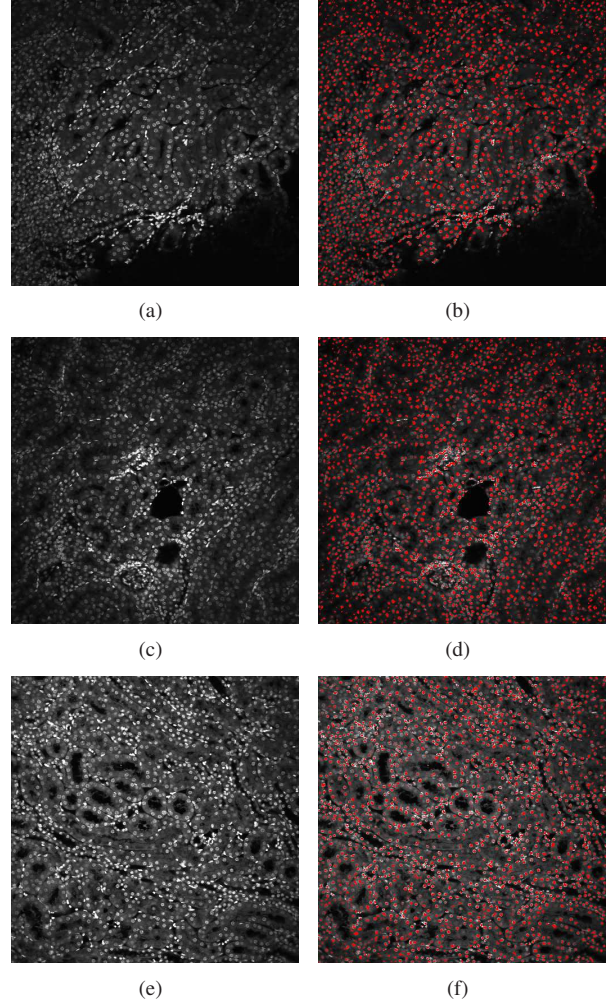

Figure 11. Nuclei segmentation on different rat kidney data (a) $I_{z_9}^{orig}$ of Data-II, (b) $I_{z_9}^{seg}$ of Data-II, (c) $I_{z_{17}}^{orig}$ of Data-III, (d) $I_{z_{17}}^{seg}$ of Data-III, (e) $I_{z_6}^{orig}$ of Data-IV, (f) $I_{z_6}^{seg}$ of Data-IV

segment nuclei from fluorescence microscopy images. Our results was compared to multiple segmentation methods using 3D groundtruth of real data. Our method achieved similar performance without using real data groundtruth in training. False detection rate of our method is higher than the best results, since non-nuclei structures were not simulated in our synthetic data. In the future, we plan to improve our synthetic data generating process to create more realistic synthetic volumes by adding negative samples.

5. Acknowledgments

University and was collected at the Indiana Center for Biological Microscopy. Data-II, Data-III, and Data-IV were provided by Tarek Ashkar of the Indiana University School of Medicine.

Address all correspondence to Edward Delp, ace@ecn.purdue.edu

References

[1] C. Vonesch, F. Aguet, J. Vonesch, and M. Unser, "The colored revolution of bioimaging," *IEEE Signal Processing Magazine*, vol. 23, no. 3, pp. 20–31, May 2006.

[2] K. W. Dunn, R. M. Sandoval, K. J. Kelly, P. C. Dagher, G. A. Tanner, S. J. Atkinson, R. L. Bacallao, and B. A. Molitoris, "Functional studies of the kidney of living animals using multicolor two-photon microscopy," *American Journal of Physiology-Cell Physiology*, vol. 283, no. 3, pp. C905–C916, September 2002.

[3] F. Helmchen and W. Denk, "Deep tissue two-photon microscopy," *Nature Methods*, vol. 2, no. 12, pp. 932–940, December 2005.

[4] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, January 1988.

[5] R. Delgado-Gonzalo, V. Uhlmann, D. Schmitter, and M. Unser, "Snakes on a plane: A perfect snap for bioimage analysis," *IEEE Signal Processing Magazine*, vol. 32, no. 1, pp. 41–48, January 2015.

[6] B. Li and S. T. Acton, "Automatic active model initialization via Poisson inverse gradient," *IEEE Transactions on Image Processing*, vol. 17, no. 8, pp. 1406–1420, August 2008.

[7] K. Lorenz, P. Salama, K. Dunn, and E. Delp, "Three dimensional segmentation of fluorescence microscopy images using active surfaces," *Proceedings of the IEEE International Conference on Image Processing*, pp. 1153–1157, September 2013, Melbourne, Australia.

[8] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Transactions on Image Processing*, vol. 10, no. 2, pp. 266–277, February 2001.

[9] S. Lee, P. Salama, K. Dunn, and E. Delp, "Segmentation of fluorescence microscopy images using three dimensional active contours with inhomogeneity correction," *Proceedings of the IEEE International Symposium on Biomedical Imaging*, pp. 709–713, April 2017, Melbourne, Australia.

[10] G. Paul, J. Cardinale, and I. F. Sbalzarini, "Coupling image restoration and segmentation: A generalized linear model/Bregman perspective," *International Journal of Computer Vision*, vol. 104, no. 1, pp. 69–93, March 2013.

[11] A. Rizk, G. Paul, P. Incardona, M. Bugarski, M. Mansouri, A. Niemann, U. Ziegler, P. Berger, and I. F. Sbalzarini, "Segmentation and quantification of subcellular structures in fluorescence microscopy images using Squassh," *Nature Protocols*, vol. 9, no. 3, pp. 586–596, February 2014.

[12] A. Dufour, V. Shinin, S. Tajbakhsh, N. Guillen-Aghion, J. C. Olivo-Marin, and C. Zimmer, "Segmenting and tracking fluorescent cells in dynamic 3-D microscopy with coupled active surfaces," *IEEE Transactions on Image Processing*, vol. 14, no. 9, pp. 1396–1410, September 2005.

[13] O. Dzyubachyk, W. A. van Cappellen, J. Essers, W. J. Niessen, and E. Meijering, "Advanced level-set-based cell tracking in time-lapse fluorescence microscopy," *IEEE Transactions on Image Processing*, vol. 29, no. 3, pp. 852–867, March 2010.

[14] S. Arslan, T. Ersahin, R. Cetin-Atalay, and C. Gunduz-Demir, "Attributed relational graphs for cell nucleus segmentation in fluorescence microscopy images," *IEEE Transactions on Medical Imaging*, vol. 32, no. 6, pp. 1121–1131, June 2013.

[15] J. Cardinale, G. Paul, and I. F. Sbalzarini, "Discrete region competition for unknown numbers of connected regions," *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp. 3531–3545, August 2012.

[16] N. Gadgil, P. Salama, K. Dunn, and E. Delp, "Nuclei segmentation of fluorescence microscopy images based on midpoint analysis and marked point process," *Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation*, pp. 37–40, March 2016, Santa Fe, NM.

[17] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural networks*, vol. 61, pp. 85–117, 2015.

[18] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. van der Laak, B. van Ginneken, and C. I. Sanchez, "A survey on deep learning in medical image analysis," *arXiv preprint arXiv:1702.05747*, February 2017.

[19] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, November 1998.

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Proceedings of the Neural Information Processing Systems*, pp. 1097–1105, December 2012, Lake Tahoe, NV.

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

[22] D. Ciresan, A. Giusti, L. M. Gambardella, and J. Schmidhuber, "Deep neural networks segment neuronal membranes in electron microscopy images," *Proceedings of the Neural Information Processing Systems*, pp. 1–9, December 2012, Lake Tahoe, NV.

[23] B. Dong, L. Shao, M. D. Costa, O. Bandmann, and A. F. Frangi, "Deep learning for automatic cell detection in widefield microscopy zebrafish images," *Proceedings of the IEEE International Symposium on Biomedical Imaging*, pp. 772–776, April 2015, Brooklyn, NY.

[24] M. Kolesnik and A. Fexa, "Multi-dimensional color histograms for segmentation of wounds in images," *Proceedings of the International Conference Image Analysis and*

Recognition, pp. 1014–1022, September 2005, Toronto, Canada.

[25] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," *Proceedings of the Medical Image Computing and Computer-Assisted Intervention*, pp. 231–241, October 2015, Munich, Germany.

[26] F. Xing, Y. Xie, and L. Yang, "An automatic learning-based framework for robust nucleus segmentation," *IEEE Transactions on Medical Imaging*, vol. 35, no. 2, pp. 550–566, February 2016.

[27] A. Prasoon, K. Petersen, C. Igel, F. Lauze, E. Dam, and M. Nielsen, "Deep feature learning for knee cartilage segmentation using a triplanar convolutional neural network," *Proceedings of the Medical Image Computing and Computer-Assisted Intervention*, pp. 246–253, September 2013, Nagoya, Japan.

[28] H. Roth, L. Lu, A. Seff, K. Cherry, J. Hoffman, S. Wang, J. Liu, E. Turkbey, and R. Summers, "A new 2.5D representation for lymph node detection using random sets of deep convolutional neural network observations," *Proceedings of the Medical Image Computing and Computer-Assisted Intervention*, pp. 520–527, September 2014, Boston, MA.

[29] C. Fu, D. Ho, S. Han, P. Salama, K. Dunn, and E. Delp, "Nuclei segmentation of fluorescence microscopy images using convolutional neural networks," *Proceedings of the IEEE International Symposium on Biomedical Imaging*, pp. 704–708, April 2017, Melbourne, Australia.

[30] O. Cicek, A. Abdulkadir, S. Lienkamp, T. Brox, and O. Ronneberger, "3D U-Net: Learning dense volumetric segmentation from sparse annotation," *Proceedings of the Medical Image Computing and Computer-Assisted Intervention*, pp. 424–432, October 2016, Athens, Greece.

[31] J. L. Clendenon, C. L. Phillips, R. M. Sandoval, S. Fang, and K. W. Dunn, "Voxx: a PC-based, near real-time volume rendering system for biological microscopy," *American Journal of Physiology-Cell Physiology*, vol. 282, no. 1, pp. C213–C218, January 2002.

[32] P. Sarder and A. Nehorai, "Deconvolution methods for 3-D fluorescence microscopy images," *IEEE Signal Processing Magazine*, vol. 23, no. 3, pp. 32–45, May 2006.

[33] F. Luisier, T. Blu, and M. Unser, "Image denoising in mixed Poisson-Gaussian noise," *IEEE Transactions on Image Processing*, vol. 20, no. 3, pp. 696–708, March 2011.

[34] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, March 2015.

[35] R. Collobert, K. Kavukcuoglu, and C. Farabet, "Torch7: A matlab-like environment for machine learning," *Proceedings of the BigLearn workshop at the Neural Information Processing Systems*, pp. 1–6, December 2011, Granada, Spain.