SCNN: Shape Constrained Neural Network for Membrane Structure Segmentation via Joint Pooling Operation

Anonymous ICCV submission

Paper ID ****

Abstract

Accurate segmentation of biomembrane structure is a crucial step to obtain morphological statistics in biomedical analysis. However in many scenarios, prior shape knowledge about biomedical objects is available, which is especially useful for segmenting dense and regular objects. In this paper, we introduce a new approach, named shape constrained neural network(SCNN), by incorporating prior shape knowledge about plausible components into neural network. Our SCNN is a multi-task learning framework that simultaneously predicts coarse segmentation map and parameterized contour expressions map as auxiliary. The auxiliary information are utilized to optimize the contour shape of each object, especially for regular objects. Moreover, a novel joint pooling operation is developed as a trainable layer to selectively express some predictions, which is both beneficial to multi-outputs. The whole process is efficient and optimized end-to-end. Experiments on synaptic vesicle segmentation and scene glands segmentation demonstrate the effectiveness of our approach. And we expect our method to empower more works to incorporate various shape constraints into segmentation tasks through deep convolutional neural networks. Code is made publicly available at https://github.com/mboboUSTC/SCNN.git.

1. Introduction

Recent advances in biomedical image analysis have assisted many pathologists and biologists to facilitate their researches [18], [15], [5], [10], [12]. Among these researches, a significant application is to obtain the accurate segmentation of specific membrane structure in a biomedical image, such as lumenal glands, synaptic vesicles and cells. Especially, the morphological shape and spatial distribution of synaptic vesicles is helpful to study the neural activity in different brain regions, while [copy] morphological statistics of lumenal glands has been routinely used to assess the malignancy degree of adenocarcinomas.[copy] Conventionally, these crucial steps are performed by human expert, which are time-consuming and suffer from subjective factors. [copy]Therefore automatic segmentation methods are highly demanded to improve the efficiency as well as reliability and reduce the workload on experts[copy].

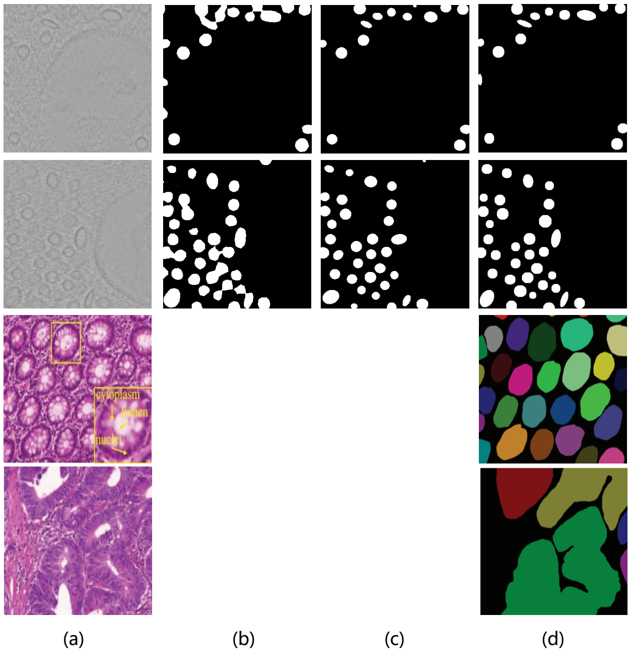

Figure 1. Examples illustrating deficiency of existing methods for biomedical segmentation. First two rows are dense synaptic vesicle with regular shape, and bottom two rows are benign and malignant gland. (a) biomedical image; (b) results from DCAN; (c) results from SCNN by incorporating prior shape constraint; (d) annotations by experts.

[copy]However, there exists several challenges in these tasks.[copy] First, biomedical images are usually noisy and low contrast, because of deficient imaging technique as shown in Figure 1 (left column). Second, due to compact and dense arrangement of majority membrane structures, it

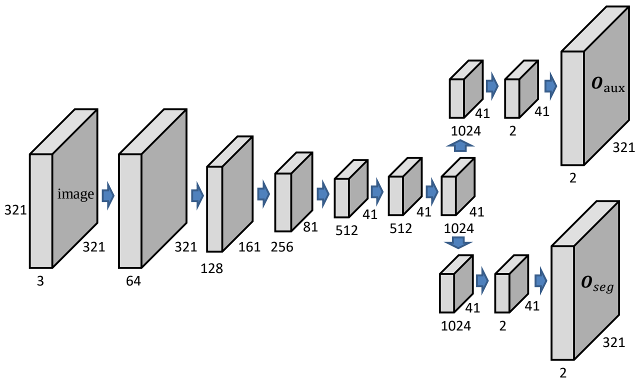

Figure 2. The illustration of our multi-task network architecture.

is hard to separate objects individually, which is known as touching problem. Third, as most deep neural architecture of biomedical images are based on fully convolutional networks [11], they inevitably suffer from poor localized object boundaries caused by large receptive fields and many pooling layers.

Recently, deep neural networks have demonstrated excellent performance in biomedical image segmentation with the use of fully convolutional networks [11], [8], [15], [16], [4], [10], [18]. However, [copy]due to the employment of max-pooling and downsampling, the output of these networks tend to have poorly localized object boundaries[copy]. For this reason, [copy][15] proposed the U-net that designed a U-shaped deep convolutional network for biomedical image segmentation.[copy] [copy]It uses skip connections between the contracting and expanding paths to directly propagate context information to higher resolution layers to preserve details.[copy] Later, a UNet variant, DeepVentricle [10], has been used for cardiac segmentation, which used same padding instead of valid padding. Further improvements have been shown in DCAN [2], which [copy]investigates the complementary information of objects and contours under a multi-task learning framework.[copy] Specially, [weak copy]DCAN simultaneously segment the object and separate the clustered objects into individual ones with the help of their contours[weak copy]. Although these methods achieved promising results in their segmentation task, they may fail to achieving satisfying performance in denser, smaller objects with regular shape as shown in Figure 1. Exactly, segmenting synaptic vesicles in our task raises higher demand on localizing contour for each vesicles.

In this paper, we propose the first shape constraint neural network (SCNN) to segment dense objects by inherently incorporating prior shape knowledge into the network. Similar to [2], we formulate the network as a multi-task learning framework by simultaneously predicting a segmentation map and an auxiliary map. Instead of predicting contour probability map, as used in [2], [6], [1], our SCNN learn a parameterized description of contour shape as auxiliary map, which emphasizes more on overall shape of object. The complementary information in parameterized contour description can not only separate objects into individual ones, but also optimize the contours shape.. However since there existing some seriously deformable objects, contours shape can't be parameterized uniformly and accurately. To this end, we select a best representative shape as constrain and only modify segmentation predictions in ambiguous region, where usually contains contours, using the predicted parameterized contour description.

However, predicting parameterized contours description over the whole map is a much tougher task than contour probability map. Therefore, we proposed a novel joint max pooling (JMP) to only predict the contour description in center region of objects and fill the rest region with them. Furthermore, JMP is designed as a trainable layer, of which the back propagation is benefit to both segmentation and parameterized contours description.

The contribution of this paper are: 1) effectively incorporating shape constraint into deep neural networks for biomedical image segmentation, 2) joint max pooling for benefiting both multi-task outputs, 3) achieving better performance on diverse biomedical segmentation tasks, 4) in experiments, we show that our method can be extended to scent detection task, which obtains the state of art performance

2. Proposed Method

A complete pipeline of Shape Constrained Neural Network (SCNN) is illustrated in Figure 1. The framework is trained end-to-end and consists of three key components: 1) a multi-task neural network based on FCN, 2) proposed joint max pooling and 3) optimizing segmentation result with parameterized contour description.

2.1. Multi-task FCN

Due to essential ambiguity in touching regions caused by segmentation map, complementary information is needed to separate clustered objects into individuals ones. In this section, we proposed a multi-task learning framework that simultaneously explore segmentation map $\mathbf{O}_{seg}$ and auxiliary map $\mathbf{O}_{aux}$ to provide complementary information. A detailed description of network architecture is shown in Figure 1.

In this network, the feature learning part is shared and based on the publicly available DeepLab model [7], which introduces zeros into the filters to enlarge its Field-Of-View. It is initialized from VGG-16 ImageNet pretrained model. Subsequently, feature maps output by last shared conv layer are fed into two individual branches. In each branch, successive two conv layers, respectively with kernel size of

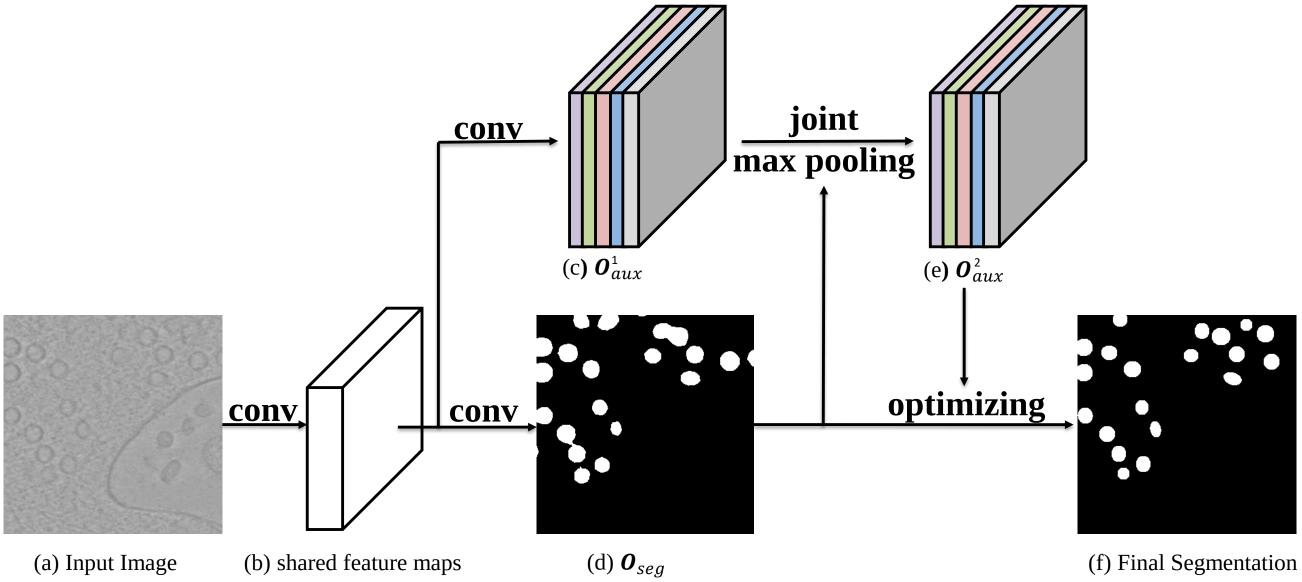

Figure 3. Overview of our proposed scnet. Given an image (a), multi-task neural network simultaneously predict a coarse segmentation (d) and parameterized contour description (d) using shared feature maps (b). Then a joint max pooling is applied to pool (c) with (d) and output new parameterized contour description (e). Finally, segmentation (f) is obtained by optimizing coarse segmentation (d) with the parameterized contour description (e).

3×3 and 1×1, are applied to input map, and then an upsampling layer restore their resolution to the input image's. During training, the parameters of shared network are jointly optimized, [copy] while the parameters of two individual branches are updated independently.

Instead of directly predicting contour probabilities [2], [18], we choose parameterized contour description as our complementary information, which emphasizes more on the overall shape. Especially, we use five parameters: θ^*, x_c^*, y_c^*, a^*, b^* to depict an ellipse shape. θ^* is the rotated angle of major axis. x_c^*, y_c^* are the ellipse center coordinates. And a^*, b^* are respectively major and minor axis. Different definitions of parameters means different shape prior knowledge of object contour. Similar to [13], the predicted contour description on (x, y) is expressed by $\mathbf{O}_{aux}(x, y) = [\theta, dx_c, dy_c, a, b]$, where

$$\begin{aligned} \theta &= \theta^* \\ dx_c &= (x - x_c^*)/width \\ dy_c &= (y - y_c^*)/height \\ a &= a^*/width \\ b &= b^*/height \end{aligned} \tag{1}$$

[θ^*, x_c^*, y_c^*, a^*, b^*] are parameters depicting the true ellipse shape of the object to be segmented. $width$ and $height$ are image size.

The objective function follows the multi-task loss in Faster R-CNN [14]. Our loss function for an image is defined as:

$$\begin{aligned} L(\mathbf{O}_{seg}, \mathbf{O}_{aux}) = L_{seg}(\mathbf{O}_{seg}, \mathbf{O}_{seg}^*) + \\ \lambda L_{aux}(\mathbf{O}_{aux}, \mathbf{O}_{aux}^* \mathbf{O}_{seg}) \end{aligned} \tag{2}$$

where $L_{aux}(\mathbf{O}_{aux}, \mathbf{O}_{aux}^* \mathbf{O}_{seg}^*)$ only compute loss on regions with positive $\mathbf{O}_{seg}^*$ and λ is a balancing weight.

2.2. Joint Max Pooling

In this section, we proposed a novel joint max pooling (JMP) to further improve both the accuracy of segmentation and parameterized contour predictions. Different from conventional max pooling, our JMP takes two inputs and pooling one with the other one.

In a conventional pooling operation, the pooling operation can expressed by:

$$y_{\mu,\nu} = \sum_{i,j} x_{i,j} s_{i,j} \qquad x_{i,j} \in \overline{\mathbf{X}}, s_{i,j} \in \mathbf{S} \tag{3}$$

where $\overline{\mathbf{X}}$ is the pooling window sliding on input map, and $\mathbf{S}$ is a weight matrix with the same size of $\overline{\mathbf{X}}$. Specifically for max pooling, $\mathbf{S}$ in Eq. 3 is a binary matrix, whose elements are all zero except for the position which has a maximum x in $\overline{\mathbf{X}}$. When Eq. 3 is an average pooling, all elements in $\mathbf{S}$ are identical averaging coefficient. Intuitively, $\mathbf{S}$ acts like an "indictor" determining which information in $\overline{\mathbf{X}}$ should be propagated to next layer. Based on this observation, we proposed a split version of pooling by obtaining $\mathbf{S}$ from an

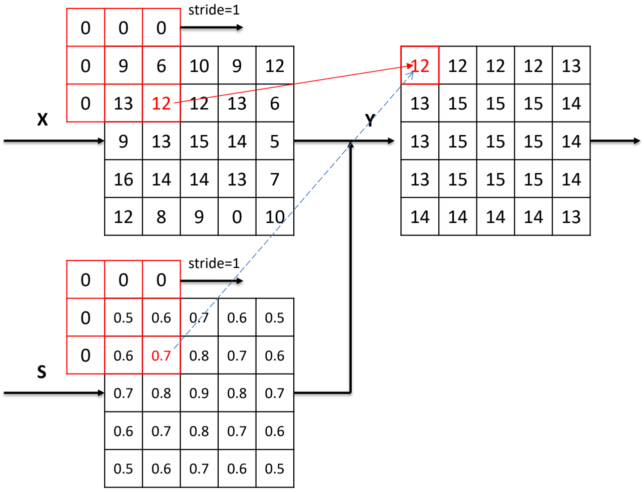

Figure 4. An example of joint max pooling. Two windows of same size synchronously slide on **X** and **S**. Top window will propagate the element, of which the position corresponding to bottom window has a maximum value, to next layer.

independent input, instead of $\overline{\mathbf{X}}$. In practice, two windows with same size synchronously slide on two independent input. One of the window is denoted as indicating window acting as the "indictor", while the other one is denoted as pooling windows, of which the useful information should be propagated to next layer. In other word, elements in indicating window determine the pooling strategy in pooling windows. A simple example is illustrated in Figure 2.2.

However for max pooling, **S** is hard to be directly learnt to be binary. Therefore we add a threshold function by:

$$y_{\mu,\nu} = \sum_{i,j} x_{i,j} G(s_{i,j}) \quad x_{i,j} \in \overline{\mathbf{X}}, s_{i,j} \in \mathbf{S}$$
$$G_{\mathbf{S}}(s_{i,j},) = \begin{cases} 1 & if\ s_{i,j} >= max(\mathbf{S}) \\ 0 & else \end{cases} \tag{4}$$

From Figure 2.2, it should be noted that most elements in **X** have been substituted with the element of which the position corresponding to maximum in **S**. Therefore in SCNN, if $\mathbf{O}_{seg}$ is denoted as **S** and $\mathbf{O}_{aux}$ is denoted as **X**, only the contour description in $\mathbf{O}_{aux}$ with a local maximum objectness in $\mathbf{O}_{seg}$ can be retained by JMP. Moreover, the discarded elements in $\mathbf{O}_{aux}$ will be replaced by a nearest retained contour description when pooling stride is set to be 1 And positions with local maximum objectness usually correspond to the center region of objects. Especially, the kernel size and iterations of JMP determine how far a retained contour expression can be spread. JMP guarantees the accuracy and consistency of contour description in ambiguous region of an object.

One important contribution of our JMP is that the residual error can be correctly back propagated to its inputs. This makes it a trainable layer in any network architecture and our SCNN become a fully trainable system. Defining $L_{\mathbf{X}}$ as the residual error on **X** , the back propagation for $x_{i,j}$ can be expressed by:

$$\frac{\partial L_{\mathbf{X}}}{\partial x_{i,j}} = \frac{1}{m} \sum_{y_{\mu,\nu} \in \mathbf{U}} \frac{\partial L_{\mathbf{X}}}{\partial y_{\mu,\nu}} G_{\mathbf{S}_{u,v}}(s_{i,j}) \tag{5}$$

where **U** is output set $\{y_{\mu.\nu}\}$ associated with $x_{i,j}$ and m is the size of **U**. $S_{u,v}$ is the corresponding pooling window centered on u, v of **X**. Different from conventional max pooling, Eq. 5 converge the gradients on the positions with local maximum $s_{i,j}$ which usually are the centers of object. Implementing Eq. 5 to our SCNN can make it only focus on predicting accurate parameterized contour description on the center area of objects, instead of the whole region.

Defining $L_{\mathbf{S}}$ as the residual error on **S**. We assume that $s_{i,j}$ not only influences the following $y_{i,j}$ but also feeds a subsequent layer in Figure 1, thus also receiving gradient contributions $\frac{\partial L_{\mathbf{S}}}{\partial s_{i,j}}$ from that layer during back-propagation. The back propagation for $s_{i,j}$ is formulated by.

$$\begin{aligned} \frac{\partial L_{\mathbf{S}}}{\partial s_{i,j}} &= \frac{\partial L_{\mathbf{S}}}{\partial s_{i,j}} + \frac{1}{m} \sum_{y_{\mu,\nu} \in \mathbf{U}} \frac{\partial L_{\mathbf{X}}}{\partial y_{\mu,\nu}} \frac{\partial y_{\mu,\nu}}{\partial s_{i,j}} \\ &= \frac{\partial L_{\mathbf{S}}}{\partial s_{i,j}} + \frac{1}{m} \sum_{y_{\mu,\nu} \in \mathbf{U}} \frac{\partial L_{\mathbf{X}}}{\partial y_{\mu,\nu}} x_{i,j} \frac{\partial G_{\mathbf{S}_{u,v}}(s_{i,j})}{\partial s_{i,j}} \\ &= \frac{\partial L_{\mathbf{S}}}{\partial s_{i,j}} + \sum_{y_{\mu,\nu} \in \mathbf{U}} \frac{1}{m} \frac{\partial L_{\mathbf{X}}}{\partial y_{\mu,\nu}} x_{i,j} \delta_{\mathbf{S}_{u,v}}(s_{i,j}) \end{aligned} \tag{6}$$

where $\delta_{\mathbf{S}_{u,v}}(s_{i,j})$ is the derived function of $G_{\mathbf{S}}(s_{i,j})$, which has an infinite response when $s_{i,j} = max(\mathbf{S}_{u,v})$. In order to normally back propagate, $\frac{\partial L_{\mathbf{S}}}{\partial s_{i,j}}$ is approximated by:

$$\frac{\partial L_{\mathbf{S}}}{\partial s_{i,j}} = \frac{\partial L_{\mathbf{S}}}{\partial s_{i,j}} (1 + \frac{1}{m} \sum_{y_{\mu,\nu} \in \mathbf{U}} \lambda \widetilde{\delta}_{\mathbf{S}_{u,v}}(s_{i,j}))$$
$$\widetilde{\delta}_{\mathbf{S}_{u,v}}(s_{i,j}) = \begin{cases} 1 & if\ s_{i,j} = max(\mathbf{S}) \\ 0 & else \end{cases} \tag{7}$$

Intuitively, Eq. 7 add a loss weight on gradients of local maximum $s_{i,j}$ with the control λ (set according to iterations of JMP) to avoid false detection as much as possible.

2.3. Fusion for Final Segmentation

With the predicted probability maps of objectness $\mathbf{O}_{seg}$ and parameterized contour description $\mathbf{O}_{aux}$ from SCNN, the final segmentation $m(i, j)$ can be obtained by fusing them together:

$$m(i,j) = \begin{cases} 1 & if\ \mathbf{O}_{seg}(i,j) > \tau_2 \\ 0 & if\ \mathbf{O}_{seg}(i,j) < \tau_1 \\ f(\mathbf{O}_{aux}(i,j)) & else \end{cases} \tag{8}$$

where τ_2 and τ_1 are the thresholds (set empirically) to control the degree of object contour modification by $\mathbf{O}_{aux}(i,j)$. $f(\mathbf{O}_{aux}(i,j))$ is a function judging whether a position is within a shape by its coordinate (i,j) and shape description $\mathbf{O}_{aux}(i,j)$. For example in our task, we define an ellipse by $\mathbf{O}_{aux}(i,j) = [\theta, x_c, y_c, a, b]$, therefore the function is expressed by:

$$
\begin{aligned}
f(\mathbf{O}_{aux}(i,j)) &= \begin{cases} 1 & if\ \frac{dx^2}{a^2} + \frac{dy^2}{b^2} < 1 \\ 0 & else \end{cases} \\
dx &= cos(\theta)(i - x_c) + sin(\theta)(j - y_c) \\
dy &= -sin(\theta)(i - x_c) + cos(\theta)(j - y_c)
\end{aligned}
\tag{9}
$$

Our fusion strategy can appropriately utilize prior shape knowledge to optimize segmented object. It can not only separate objects into individual ones, but also optimize most regular object while don't loss generalization to deformable objects. The SCNN can be easily extended to other shape constraint, if only the shape can be parameterized.

3. Experiments

We first present results on two diverse biomedical image segmentation problems to demonstrate our superiority over existing methods. The applications including synaptic vesicle segmentation and gland segmentation. Additionally, our SCNN is extended to scene text segmentation task to prove its generic applicability.

3.1. Synaptic vesicle segmentation

Dataset Synaptic vesicle is a good example to evaluate our method. The images were acquired by . They are much noisy and contain both vesicle and ambiguous membrane structure, such as cell nucleus. The dataset is composed of 8787 images with ground truth annotations provided by biologists. [copy]To increase the robustness and reduce overfitting, we utilized the strategy of data augmentation to enlarge the training dataset. The augmentation transformations include translation, rotation and flipping.[copy] For cross validation experiments, The dataset are divided into two parts. The first five out of six images are prepared to train our model and the rest of them are used to test its performance. The validation processing has been repeated several times and the average performance will be reports.

Implementation details [copy]Our framework was evaluated on the open-source deep learning library Caffe [9][copy]. The network resize the image to 321×321 as input and output objectness score and parameterized contour maps. [copy]A three-step training process is employed. We first train the segmentation branch independently and then we jointly fine-tune the multi-task network without joint max pooling. [copy]Specifically, we employ exactly the same setting as [7] in the first stage.[copy] In the second stage, the balancing weight λ is set to be 5 and a small learning rate of 10^{-8} is employed for fine-tuning in the second stage. In the final stage, the joint max pooling is attached to the tail of network and the whole model is fine-tuned again to further improve the performance. In order to make better use of parameterized contour information, the joint max pooling is iterated twice with the pooling size 11×11 and stride 1. Finally in fusion step as most vesicle contours shape are regular, τ_2 and τ_1 are set to be 0.9 and 0.2 for a strong modification by parameterized contour constraint.

Evaluation setup The evaluation criteria in our experiments includes F1 score and pixel intersection-overunion (IOU) averaged across different classes. The F1 score considers the performance of object detection, while IOU consider the segmentation performance, respectively.

For detection, the metric F1 score is defined as:

$$
F1 = \frac{2PR}{P+R}, P = \frac{N_{tp}}{N_{tp}+N_{fp}}, R = \frac{N_{tp}}{N_{tp}+N_{fn}} \tag{10}
$$

where M_{tp},M_{fp} and M_{fn} are respectively the number of true positives, false positives and false negatives. Especially, a segmented object that intersects with at least 50 with the ground truth is regarded as a true positive, otherwise it is regarded as a false positive. If a ground truth object has no corresponding segmented object that intersects more than 50, it is regarded as a false negative.

For segmentation, the metric IOU is defined as:

$$
IOU = \frac{1}{N_s} \sum_{i=1}^{N_s} \frac{G_i \bigcap S_i}{G_i \bigcup S_i} \tag{11}
$$

where N_i denotes the number of segmented classes. G_i denotes the ground truth of i-th class. S_i denotes the segmented map of i-th class.

Qualitative evaluation on vesicle segmentation Figure **??** shows some segmentation results of testing data. In order to verify the effectiveness of parameterized contour information, we compared the performance of u-net [15] relying only on the prediction of vesicle objects and DCAN [2] using contour probability information. From segmentation results we can see that without any complementary information, there exists many touching vesicle objects that cannot be separated by u-net. This situation usually occurs in regions with ambiguous context between two vesicle objects. And the contours of many regular objects are very coarse, as the case shown in the second row of Figure **??**. Although DCAN is capable of separating the touching vesicles into individuals in the third row in Figre **??**, the obtained contours of regular vesicle are still coarse. In comparison, our SCNN using the parameterized contour information can not only separate those touching vesicles clearly, but also obtain a much smooth and regular shape for each vesicle contour. This demonstrates the superiority of our SCNN in segmenting densely arranged objects with regular shape contour by

Method	F1	IOU
DeepLab	0.8404	0.8495
U-net	0.8404	0.8495
DCAN	0.8404	0.8495
SCNNv1	0.8404	0.8495
SCNNv2	0.8404	0.8495

Table 1. The detection and segmentation results of different methods in our synaptic vesicle segmentation dataset.

exploring the complementary information in parameterized contour description.

Quantitative evaluation We used metrics of F1 score and IOU to evaluate our method. We compare SCNN with the state of the art segmentation methods based on Deeplab [7], U-net [15] and DCAN [3], which are commonly used in biomedical image process. Their results on our synaptic vesicle dataset are shown in 3.1. And we further implement two version of SCNN. The first SCNNv1 is the implementation of multi-task neural network without joint max pooling, and SCNNv2 is the complete form. Their results are also presented in Table 3.1 to prove effectiveness of our joint max pooling.

Qualitatively, the performance of SCNN surpassed all the other methods by a large margin on vesicle segmentation task, proving its effectiveness for segmenting regular biomedical objects. From Table **??**, we can find that

3.2. Gland segmentation

Dataset In this section we present SCNN for segmenting benign and malignant gland. We consider the public dataset of *Gland Segmentation Challenge Contest* in MICCAI2015 [17] The training dataset is composed of 85 images, consisting of 37 benign and 48 malignant, with ground truth annotations provided by expert pathologists. Especially, there is a huge variation of glandular morphology in malignant case, which can prove the generalization of our SCNN to irregular objects. The same data augmentation in vesicle segmentation is implemented for a better performance.

Implementation details Because there exists many irregular objects in gland images that we desire to remain more contour information obtained by object prediction, the contour modification by parameterized contour information should be relatively weaker than segmenting vesicles. By experimental verification, we find that $\tau_2 = 0.7$ and τ_1 =0.4 produce a better results. And we still use the standard elliptic parameter as the prior shape constraint for gland, as most benign glands and few malignant glands' shape are approximate ellipses. The other implementation settings and evaluation metrics follow the vesicle segmentation.

Qualitative evaluation on gland segmentation Follow previous qualitative evaluation, we presented the results of u-net and the state of art method DCAN in gland segmentation with our SCNN in Figure **??**. The first two column are the examples of benign gland images, and the rest two column are the examples of malignant images. From the results, we can observed that both SCAN and SCNN can well solve the touching problem in benign and malignant cases. However for benign case, the contours of glands obtained by SCNN are more smooth than that of DCAN. And for malignant case, since SCNN only modify the segmentation predictions on the object border, there is no obvious deterioration compared to DCAN.

Quantitative evaluation Table **??** shows the F1 score and IOU metric over the *Gland Segmentation Challenge Contest* by several commonly used biomedical segmentation methods.

3.3. Scene text detection

We further extended SCNN to scene text detection task, which

References

[1] G. Bertasius, J. Shi, and L. Torresani. Semantic segmentation with boundary neural fields. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3602–3610, 2016. 2

[2] H. Chen, X. Qi, L. Yu, and P.-A. Heng. Dcan: Deep contour-aware networks for accurate gland segmentation. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2487–2496, 2016. 2, 3, 5

[3] H. Chen, X. Qi, L. Yu, and P.-A. Heng. Dcan: Deep contour-aware networks for accurate gland segmentation. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2487–2496, 2016. 6

[4] H. Chen, C. Shen, J. Qin, D. Ni, L. Shi, J. C. Cheng, and P.-A. Heng. Automatic localization and identification of vertebrae in spine ct via a joint learning model with deep neural networks. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 515–522. Springer, 2015. 2

[5] J. Chen, L. Yang, Y. Zhang, M. Alber, and D. Z. Chen. Combining fully convolutional and recurrent neural networks for 3d biomedical image segmentation. In *Advances in Neural Information Processing Systems*, pages 3036–3044, 2016. 1

[6] L.-C. Chen, J. T. Barron, G. Papandreou, K. Murphy, and A. L. Yuille. Semantic image segmentation with task-specific edge detection using cnns and a discriminatively trained domain transform. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4545–4554, 2016. 2

[7] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*, 2014. 2, 5, 6

[8] N. Dhungel, G. Carneiro, and A. P. Bradley. Deep learning and structured prediction for the segmentation of mass in

mammograms. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 605–612. Springer, 2015. 2

[9] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 675–678. ACM, 2014. 5

[10] J. Lieman-Sifry, M. Le, F. Lau, S. Sall, and D. Golden. Fastventricle: Cardiac segmentation with enet. In *International Conference on Functional Imaging and Modeling of the Heart*, pages 127–138. Springer, 2017. 1, 2

[11] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015. 2

[12] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello. Enet: A deep neural network architecture for real-time semantic segmentation. *arXiv preprint arXiv:1606.02147*, 2016. 1

[13] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 779–788, 2016. 3

[14] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015. 3

[15] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015. 1, 2, 5, 6

[16] H. R. Roth, L. Lu, A. Farag, H.-C. Shin, J. Liu, E. B. Turkbey, and R. M. Summers. Deeporgan: Multi-level deep convolutional networks for automated pancreas segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 556–564. Springer, 2015. 2

[17] K. Sirinukunwattana, D. R. Snead, and N. M. Rajpoot. A stochastic polygons model for glandular structures in colon histology images. *IEEE transactions on medical imaging*, 34(11):2366–2378, 2015. 6

[18] Y. Xu, Y. Li, M. Liu, Y. Wang, Y. Fan, M. Lai, E. I. Chang, et al. Gland instance segmentation by deep multichannel neural networks. *arXiv preprint arXiv:1607.04889*, 2016. 1, 2, 3