CDNet: Centripetal Direction Network for Nuclear Instance Segmentation

Hongliang He[1,2]   Zhongyi Huang[1]   Yao Ding[3]   Guoli Song[2]   Lin Wang[1,2]

Qian Ren[1,2]   Pengxu Wei[4]   Zhiqiang Gao[2]   Jie Chen[1,2*]

[1]School of Electronic and Computer Engineering, Peking University, Shenzhen, China
[2]Peng Cheng Laboratory, Shenzhen, China
[3]University of Chinese Academy of Sciences, Beijing, China
[4]Sun Yat-sen University, Guangzhou, China

{hehl,hzhongyi,wangl1212,renq2019}@pku.edu.cn, dingyao16@mails.ucas.ac.cn,

{songgl,gaozhq,chenj}@pcl.ac.cn, weipx3@mail.sysu.edu.cn

Abstract

Nuclear instance segmentation is a challenging task due to a large number of touching and overlapping nuclei in pathological images. Existing methods cannot effectively recognize the accurate boundary owing to neglecting the relationship between pixels (e.g., direction information). In this paper, we propose a novel Centripetal Direction Network (CDNet) for nuclear instance segmentation. Specifically, we define centripetal direction feature as a class of adjacent directions pointing to the nuclear center to represent the spatial relationship between pixels within the nucleus. These direction features are then used to construct a direction difference map to represent the similarity within instances and the differences between instances. Finally, we propose a direction-guided refinement module, which acts as a plug-and-play module to effectively integrate auxiliary tasks and aggregate the features of different branches. Experiments on MoNuSeg and CPM17 datasets show that CDNet is significantly better than the other methods and achieves the state-of-the-art performance. The code is available at https://github.com/honglianghe/CDNet.

1. Introduction

Nuclear instance segmentation techniques enable accurate quantitative characterizations of nuclear size and shape (*e.g*., circularity and aspect ratio), which are key components of the study of cancerous tissues [9]. However, a whole-slide image contains tens of thousands of nuclei of various types (as shown in Fig. 1(a)~(b)), and nuclei display a great deal of inter- and intra-instance variability be-

---

*Corresponding authors
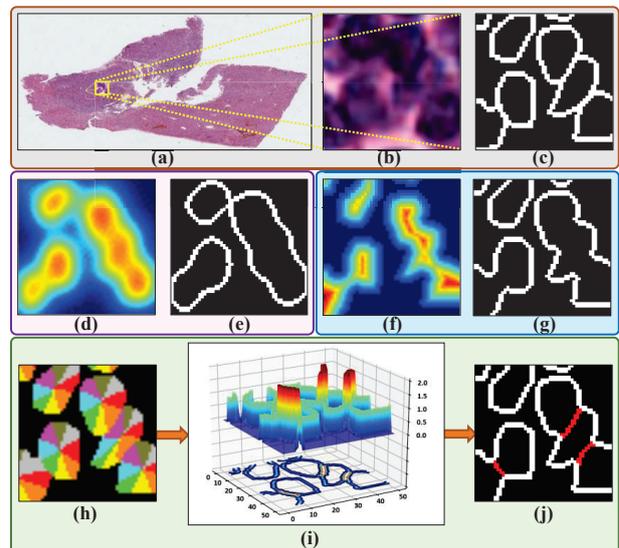
Figure 1. The schematic illustration of nuclear instance segmentation with different methods. (a) is a whole-slide image; (b) is a local region; (c) is the ground-truth boundary corresponding to (b); (d) and (f) denote the probability maps based on two kinds of existing methods, and (e) and (g) reflect the limitations of these existing methods on overlapping boundaries; (h)~(j) are obtained from our CDNet, in which (h) is the centripetal direction feature; (i) is the direction difference map, and (j) is the corresponding boundary map. The red line in (j) indicates that CDNet effectively identifies the boundaries of overlapping regions.

cause of their appearances, surroundings by organs, disease types, and even digital scanner brands. In particular, tumour nuclei tend to be present in clusters and lead to clustered overlapping instances, which also provides challenge for accurate segmentation of nuclear instances.

To tackle the above challenges of nuclear instance seg-

mentation task, different kinds of deep learning methods have been proposed by building on convolutional neural networks (CNN). In this paper, we divide these methods into three categories: (1) using CNN for object detection and then segmenting the objects within each bounding box [12, 5]; (2) using CNN for semantic segmentation and predicting instance boundary to separate different instances [9, 8]; (3) adding complex post-processing, such as watersheds [21], conditional random field [10]. Among them, the second kind draws most of attention due to the recent development of semantic segmentation techniques, and Fig. 1(d) and (f) show the probability maps of two typical methods of this kind: boundary-based model [1, 17] and distance-based model [15]. It can be seen that both boundary-based and distance-based model lack the ability to distinguish pixels belonging to two nuclei apart that touch or overlap with each other in Fig. 1(e) and (g).

In this study, we propose a centripetal direction network (CDNet) for nuclear instance segmentation. To characterize the spatial relationship between pixels within a nucleus, we define centripetal direction feature as a class of adjacent directions pointing to the nuclear center (*i.e.* Fig. 1(d)). Based on the centripetal direction feature, we propose direction difference map (DDM) and direction-guided refinement module (DGM) to improve the segmentation performance especially for overlapping instances. In particular, DDM represents intra-instance similarity and inter-instance difference (*i.e.* Fig. 1(e)), which reflects the direction difference between adjacent pixels for nuclear boundary identification. DGM is proposed as a plug-and-play module to effectively integrate auxiliary tasks and aggregate the features of three branches: point branch, direction branch, and mask branch. The optimization of these branches are combined under the guidance of centripetal direction feature.

Our contributions are summarized as follows:

- We propose a centripetal direction network (CDNet) for nuclear instance segmentation task, which uses centripetal direction feature to effectively deal with the boundary ambiguity problem of touching and overlapping nuclei.

- We propose a direction difference map (DDM) to reflect the spatial difference between adjacent pixels for nuclear boundary identification, and a direction-guided refinement module (DGM) to aggregate features of different branches.

- Our proposed CDNet is compared with several recent methods on two public nuclei segmentation datasets, and achieves the state-of-the-art performance.

2. Related work

With the development of CNN, deep learning methods have been proposed to deal with instance segmenta-

tion tasks. Detection-based instance segmentation methods [7, 14, 2] and semantic-to-instance segmentation methods [19, 25, 24] are two mainstream methods. Since nuclear instance segmentation is an intensive and small object segmentation task, the performance of detection-based instance segmentation methods are limited by detection performance [6]. Therefore, the semantic-to-instance segmentation methods have dominated the nuclear instance segmentation, and these methods are divided into the following three types.

Model structure. In recent work, Raza et al. [18] proposed Micro-Net which learns image features with multiple input resolutions and bypasses the maximum pooling operation through additional layers. Wollmann et al. [23] proposed GRUU-Net that integrates convolutional neural networks and gated recurrent neural networks on multiple image scales. Qu et al. [17] proposed a full-resolution CNN and replaced down sampling with dilated convolution, thereby retaining complete feature information to improve localization accuracy. In addition, Gehlot et al. [4] proposed an encoder-decoder-based convolutional neural network with nested-feature cascade (EDNFC-Net) which reuses features and preserves contextual information. Xiang et al. [24] proposed a Bi-directional O-shape network (BiO-Net) that reuses the building blocks in a recurrent manner to improve its segmentation capabilities.

Segmentation strategy. In order to strengthen the division of the nuclear boundary, Kumar et al. [9] described the nuclear task as a three-class task instead of two-class. Taking the three-class prediction as an intermediate step, Kang et al. [8] designed a two-stage learning framework by stacking two U-Nets. The first stage is used to estimate nuclei and their coarse boundaries, and the second stage is used to output the final fine-grained segmentation map. In addition, Naylor et al. [15] tried to address the problem of segmenting touching nuclei by formulating the segmentation problem as a regression task of distance map. Instead of detecting bounding boxes, Schmidt et al. [20] detected the star convex polygons to locate the nuclei. These ideas provide new solutions for segmentation of overlapping and clustered nuclei.

Auxiliary task learning. Chen et al. [1] first proposed a deep contour-aware network (DCAN). It expresses their organizational structure segmentation and contour detection as a multi-task learning framework to simultaneously infer the information of the object and contour. Oda et al. [16] proposed a boundary enhanced segmentation network (BESNet) which adds an extra decoding path based on the U-Net structure for boundary supervision. In addition, Liu et al. [12] proposed a multi-task model with semantic segmentation branch and instance branch. In order to aggregate the features between different optimization tasks, Zhou et al. [26] proposed a contour-aware informative aggregation
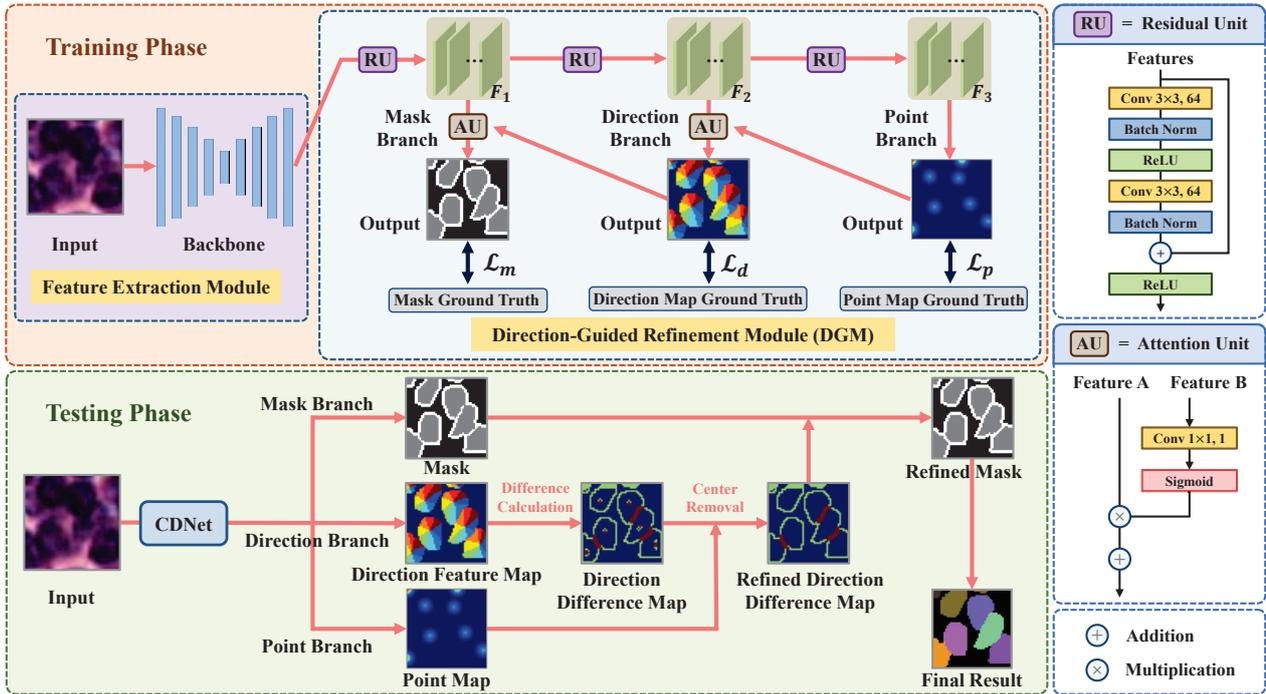
Figure 2. The flowchart of the proposed CDNet. It is classified into two parts: training phase and testing phase. There are two main modules in the training phase: feature extraction module and direction-guided refinement module. In the testing phase, the prediction results of the point detection are used to assist the direction branch to generate refined direction difference map (DDM), and then the refined DDM is used to enhance the boundary prediction of the mask branch.

network (CIA-Net) which adds a multi-level information aggregation module between the two decoder paths. Furthermore, Graham et al. [6] proposed Hover-Net that uses the rich information encoded by the vertical and horizontal distances from pixels to the nuclear center.

The above methods have improved the segmentation performance, but the correlation between pixels and the boundary feature representation of overlapping regions need to be considered again, because it leads to some bottlenecks in segmenting overlapping and clustered nuclei in histopathology images. In this study, we propose a CDNet to address the above bottlenecks. Details will be described in Sec. 3.

3. Methodology

In this paper, we propose a centripetal direction network (CDNet) for nuclear instance segmentation. The flowchart is shown in Fig. 2. The goal of the network is to learn the direction features of pixels pointing to the corresponding instance center. We construct the direction difference map according to the centripetal direction feature to distinguish the different instances of the overlapping region. To clearly explain our CDNet, we first describe the direction difference map in Sec. 3.1, and then introduce the direction-guided refinement network in Sec. 3.2.

3.1. Direction Difference Map

In order to separate different instances of overlapping regions, we first construct direction feature map based on the generated block-wise centripetal direction, and then transform the direction feature to generate DDM which can represent the context information and highlight the boundaries of instances.

3.1.1 Direction feature map

In order to enable the model to learn the centripetal direction feature for each pixel in the input image, we design a direction feature map as a direction label. Fig.3 illustrates the process of generating the centripetal direction ground truth from annotated mask. The mathematical details and technical aspects of the direction feature are described as follows.

First of all, we convert an annotated mask into a pixel-to-boundary distance map M to locate center points. We define an image $\mathcal{X} = \mathcal{P} \cup \mathcal{B}$, where $\mathcal{P}$ and $\mathcal{B}$ denote the set of foreground pixels and background pixels, respectively. Further, we define $\mathcal{P} = \{\mathcal{P}_1, \mathcal{P}_2, ..., \mathcal{P}_N\}$, where $\mathcal{P}_k$ denotes the set of pixels belonging to the kth instance in $\mathcal{X}$ and N is the number of instances in $\mathcal{X}$. Then, the pixel-to-boundary distance map $M = \{M_p\}$ (Fig. 3(c)) is calcu-
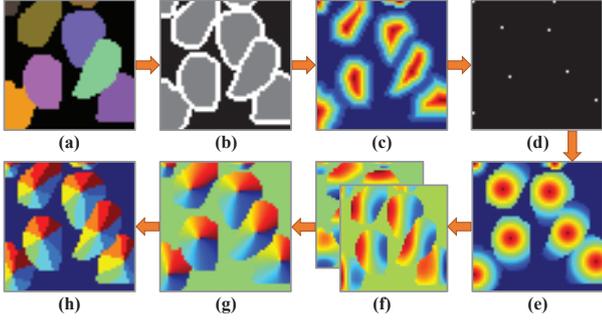
Figure 3. The process of generating centripetal direction ground truth (GT). (a) is the original ground truth mask; (b) is the three-class ground truth mask (black, gray and white regions denote the background, instance inside and boundary, respectively); (c) is the pixel-to-boundary distance map, M; (d) is the center point map; (e) is the pixel-to-center distance map, U; (f) is the gradient map, G; (g) is the angle map, A; (h) is the centripetal direction GT, D^C.

lated, where M_p denotes the pixel-to-boundary distance of pixel $\mathbf{p} = (p_x, p_y)$, and is defined as Eq. 1.

$$M_{\mathbf{p}} = \begin{cases} 0, & if \ \mathbf{p} \in \mathcal{B} \\ \sqrt{(p_x - b_x)^2 + (p_y - b_y)^2}, & if \ \mathbf{p} \in \mathcal{P} \end{cases}, \quad (1)$$

where $\mathbf{b} = (b_x, b_y)$ denotes the nearest background pixel to $\mathbf{p}$.

In the second step, we construct the pixel-to-center distance map. Specifically, we define the center axis of the kth instance as:

$$\mathcal{T}_k = \left\{ \mathbf{p} = (p_x, p_y) \mid M_p = max(M^k), \mathbf{p} \in \mathcal{P}_k \right\}, \quad (2)$$

where M^k denotes the pixel-to-boundary distance map of the kth instance. We sort $\mathcal{T}_k$ and select the medium value $\tilde{\mathcal{T}}_k$ in $\mathcal{T}_k$ as the center pixel z_k of the instance $\mathcal{P}_k$. Further, the pixel-to-center distance map U is defined as: for pixels in an instance, the value is calculated by the distance between them and their corresponding instance center. In order to avoid the effect of nuclear size on distance, we normalize the pixel-to-center distance to 0-1. For pixels in the background, the value is set to 0 (Fig. 3(e)).

In the third step, we calculate the corresponding direction gradient through the pixel-to-center distance map U, and finally obtain the the centripetal direction ground truth according to the gradient map. By utilizing the convolution operation, we convert U to the gradient map G, consisting of the horizontal gradient G^x and the vertical gradient G^y, which is used to quantify relative angle (Fig. 3(g)) and compute the centripetal direction feature (Fig. 3(h)). Specifically, for each instance pixel $\mathbf{p} \in \mathcal{P}_k$, we compute the direction towards its center pixel z_k as $\theta = arctan(G_p^x / G_p^y)$ and discretize this angle to generate $D_{\mathbf{p}}^C = \lfloor \theta * C/360 \rfloor$, where C is the number of centripetal direction classes. For example, as shown in Fig. 3(h), these pixels within the instance have 8 directions.

We utilize the generated centripetal direction ground truth map D^C as direction supervisor, by the loss function in terms of both cross-entropy loss and dice loss:

$$\begin{aligned} \mathcal{L}_d &= \mathcal{L}_{ce} + w\mathcal{L}_{dice} \\ &= -\frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{C} (y_{ij} log(\hat{y}_{ij})) \\ &+ (1 - \frac{1}{C} \sum_{j=1}^{C} \frac{2 \sum_{i=1}^{N} y_{ij} \hat{y}_{ij}}{\sum_{i=1}^{N} y_{ij} + \sum_{i=1}^{N} \hat{y}_{ij}}), \end{aligned} \quad (3)$$

where $\mathcal{L}_{ce}$ denotes the cross-entropy loss; $\mathcal{L}_{dice}$ denotes the dice loss; N is the number of pixels; y_{ij} is the ground-truth of the input x_i; and $\hat{y}_{ij}$ is the probability that the input x_i is predicted to be class j. w is a weight coefficient, which is taken as 1 in the experiments.

The centripetal direction feature is block-wise, which has better ability to resist noise than pixel-wise feature and to represent local information than instance-wise feature. Moreover, if multiple instances are adjacent or have overlapping regions, there exist drastic directional changes between pixels in the boundary or overlapping regions, facilitating to segment instances.

3.1.2 The generation of direction difference map

To directly strengthen the difference between boundaries, we further propose the direction difference map, which is transformed via the direction feature.

As mentioned above, we convert the direction feature map (Fig. 4(a)) of each pixel predicted by the network into the gradient map G. Furthermore, we use the cosine distance function to measure the directional similarity between two pixels, $dis(\mathbf{p}^i, \mathbf{p}^j) = \frac{G_{\mathbf{p}^i} G_{\mathbf{p}^j}}{||G_{\mathbf{p}^i}|| \times ||G_{\mathbf{p}^j}||}$.

Subsequently, the direction difference map, denoted as D, is computed by the directional similarity. $D_{\mathbf{p}^i}$ of the pixel $\mathbf{p}^i$ in D, is calculated as follows:

$$D_{\mathbf{p}^i} = \begin{cases} 1 - \min_{\mathbf{p}^j \in \mathcal{V}_{\mathbf{p}^i}} \{dis(\mathbf{p}^i, \mathbf{p}^j)\}, & if \ \mathbf{p}^i \in \mathcal{P} \\ 0, & if \ \mathbf{p}^i \in \mathcal{B} \end{cases}, \quad (4)$$

where $\mathcal{V}_{\mathbf{p}^i}$ denotes the set of pixels surrounding $\mathbf{p}^i$.

According to the definition in Eq. 4, pixels within the same instance except that adjacent to instance centers have small values in D, while pixels in the overlapping regions belonging to different instances or in the boundary have large values in D. In other words, the direction difference map describes one characteristic for pixels at the boundary or the overlapping regions. In this way, the pixels in overlapping regions are distinguished according to direction difference map. How to utilize the direction difference map in our CDNet to strengthen the ability of segmentation is introduced in the Section 3.2.

(a) Direction feature map **(b) Direction difference map**
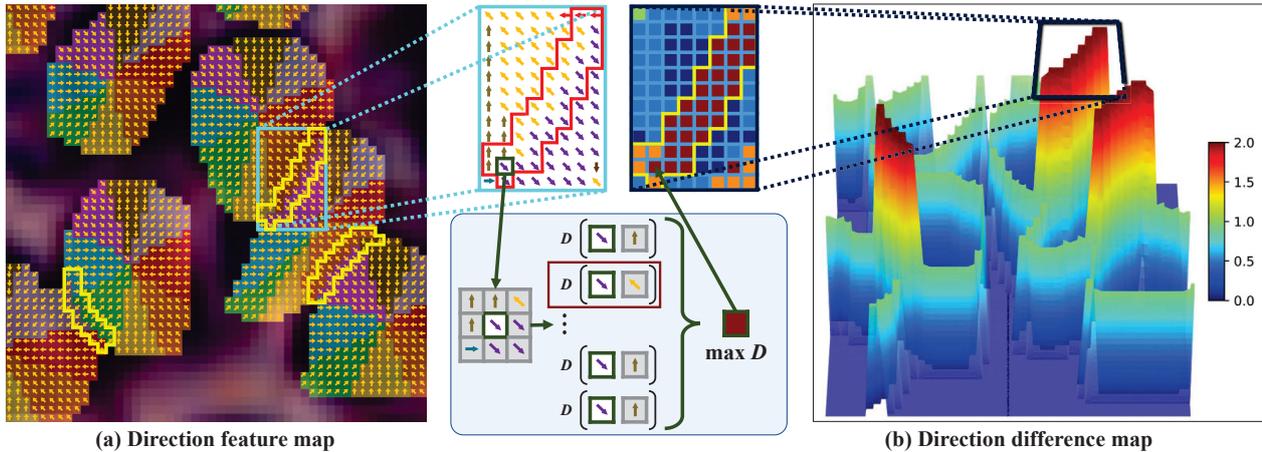
Figure 4. Illustration of Direction feature map and Direction difference map. In the centripetal direction map, each instance has k directions ($k = 8$ in (a)); Each pixel calculates the maximum direction difference between the pixel and its adjacent pixels, which is depicted by the direction difference map. Note: in order to clearly show the direction difference near the boundary, we temporarily ignore the direction difference near the instance center in (b).

3.2. Centripetal direction network

In our proposed CDNet, its main structure consists of two components: feature extraction module and direction-guided refinement module. In the feature extraction module, CNN is used to extract feature information and its backbone networks is dynamically adjusted according to different tasks. The flowchart of the proposed CDNet is provided in Fig. 2. The training and inference of the centripetal direction network are described in detail as follows.

3.2.1 Training of CDNet

Based on the centripetal direction feature, we propose a plug-and-play direction-guided refinement module to improve the original segmentation performance. As shown in Fig. 2, the module contains three branches: mask branch, direction branch and point branch, which correspond to mask segmentation task, direction segmentation task and center point detection task, respectively. The ground truth of three tasks can be obtained in the process of generating centripetal direction feature, so there is no need for additional artificial tagging. This module integrates the advantages of different tasks and strengthens the guidance of centripetal direction feature in instance segmentation.

The center point detection task is introduced to assist the network training, in which the point branch is used to learn the center position of instances. However, there is an optimization gap between the center point detection task and the mask segmentation task, because center point detection task allows model to tend to learn more information about central area and ignore the overall features of the instance, while the mask segmentation task expects to learn the overall features and boundary features of instance. Therefore,

if these two tasks are combined directly, the network learning is confused, which leads to the decline of the prediction performance.

To this end, a direction segmentation task is introduced to bridge the optimization gap between the center point detection task and the mask segmentation task, where direction features are predicted and used to characterize the context information about instance center, boundary, and the correlation between internal pixels. Therefore, we put the direction branch in the middle of the mask branch and the point branch.

Specifically, as shown in Fig. 2, the feature map obtained from feature extraction module passes through the first residual unit (RU) to obtain the feature map F_1 of shape $H \times W \times C$. The feature map F_1 will enters two branches. The first branch passes F_1 to the second RU, and the second branch uses F_1 to generate 3-class segmentation map; in the same way, the feature map F_2 obtained by the second RU also enters two branches. The first branch passes the feature map to the third RU to obtain the feature map F_3, and the second branch is used to generate the direction feature. Point map prediction results are obtained by F_3 through corresponding convolution operation. Among them, the RU is composed of 3×3 convolutions + Batch Normalization + ReLU and shortcut connection.

In order to effectively aggregate the features of different branches, we construct a reverse transmission path, which is the attention unit (AU) in Fig. 2. In this unit, feature B is used as the attention of feature A, i.e., using feature B to guide feature A to generate output with different spatial attention. As shown in Fig. 2, the calculation process in the attention unit is: feature B undergoes a 1×1 convolution to convert the channels into one, and spatial attention is obtained through $sigmoid$ function and then is combined with

feature A through multiplication and addition.

In the process of training, direction-guided refinement module enhances the original network to improve segmentation performance. The total loss function is:

$$\mathcal{L}_{total} = w_1\mathcal{L}_m + w_2\mathcal{L}_d + w_3\mathcal{L}_p \quad (5)$$

where $\mathcal{L}_m$ is the loss of mask branch; $\mathcal{L}_d$ is the loss of direction branch, and $\mathcal{L}_p$ is the loss of point branch. w_i is a set of weight coefficients, all of which are set to 1 in the experiments. We use cross-entropy loss and dice loss in the mask branch and direction branch, and use mean square error (MSE) loss in the point branch.

3.2.2 Inference of CDNet

In the process of testing, DDM is further used as an important approach to enhance the segmentation performance of the proposed CDNet. In details, the 3-class mask map, direction feature and point map are obtained by the mask branch, direction branch and point branch respectively (Fig. 2). In the direction branch, we calculate the corresponding DDM according to Eq. 4). In DDM, there is a large direction difference between the boundary pixels and the pixels near the instance center. In order to eliminate the influence of pixels near the instance center in the refining process, we use the point map obtained by the point branch to assist in eliminating the large direction difference near the instance center. Specifically, we eliminate the value of the central region in the DDM, which refers to the area where the value in the corresponding point map is greater than the median value. Moreover, refined DDM is used as a weight map to enhance the prediction probability of the boundary class in the original 3-class mask segmentation task. The boundary reinforcement function is shown in Eq. 6 below.

$$p_i^{bre} = (p_i^b + \frac{D_{p_i}}{2}) \times (1 + D_{p_i}) \quad (6)$$

where p_i^b represents the prediction probability of the boundary class for the pixel p_i in original 3-class segmentation map, and D_{p_i} represents the direction difference value of the pixel p_i in refined DDM.

4. Experiments

4.1. Datasets and evaluation metrics

Datasets. The datasets for nuclei segmentation are relatively small, because the manual annotation requires labor-intensive work and involvement of pathologists whose time is limited and expensive. In this paper, we evaluate our proposed CDNet on two widely-used nuclei segmentation datasets. (1) **MoNuSeg [9]:** The multi-organ nuclei segmentation dataset (MoNuSeg) is one of the largest repository of hand annotated nuclei. It contains 30 H&E stained histopathology images of size 1000×1000, from 7 different organs with a total of 21,623 individual annotated nuclei. To make a fair comparison, we use the same training and testing sets as described in the supplementary materials of [9], and those images were also used for the MoNuSeg grand challenge in 2017. According to [9], we divide 30 images into three sets: 12 images for training, 4 images for validation, and 14 images for testing. (2) **CPM17 [22] :** It is from the computational precision medicine digital pathology challenge [22]. It contains 64 H&E stained histopathology images with 7,570 annotated nuclear boundaries. The dataset is split according to the original challenge [22] with 32 images in both training and testing sets.

Evaluation Metrics. To measure the overall segmentation performance of the proposed CDNet, we use four evaluation metrics as presented in [9]: F1-score (F1), average Dice coefficient (Dice), average Hausdorff distance (HausD), and the Aggregated Jaccard Index (AJI).

4.2. Implementation details

We first preprocess the training images by dividing each image into 16 patches with a size of 300×300, and obtain 192 training images for MoNuSeg and 512 training images for CPM17. For training, we randomly augment our data by standard data augmentation techniques such as color transformation, horizontal flip, rotation, elastic transformation, and cropping. Finally, we resize the image patch to a size of 256×256 before inputting the network. For segmentation supervision, we use 3-class masks: inside, boundary, and background. For point supervision, we use the heatmap obtained by Gaussian kernel convolution of the point map as the ground truth.

We use PyTorch to implement CDNet on NVIDIA 2080Ti with CUDA 10.1. RAdam [13] is used as the optimizer. The initial learning rate is set to 0.0005, and the training epoch is set to 300. In inference stage, we remove small objects with area less than 20 pixels to avoid unnecessary foreground caused by incorrect pixel prediction.

4.3. Ablation studies

To evaluate the effectiveness of key components used in our proposed CDNet, we use U-Net as the baseline model and perform a series of comparison with its variants on MoNuSeg and CPM17 datasets.

Effectiveness of direction feature. The goal is to clarify the effect of direction feature on nuclear instance segmentation. On the basis of 3-class mask supervision, we tune the number of direction classes such that direction category supervision changes in the compared models. The comparison results on MoNuSeg and CPM17 datasets are summarized in Table 1, where k indicates the number of direction classes in direction feature. Specifically, $k = 0$

Baseline	MoNuSeg				CPM17			
	F1	Dice	HausD	AJI	F1	Dice	HausD	AJI
+ Direction (k=0)	0.8608	0.8184	7.3329	0.5910	0.8688	0.8473	6.6222	0.6648
+ Direction (k=4)	0.8679	0.8223	7.0086	0.6091	0.9161	0.8710	**5.6532**	0.7097
+ Direction (k=8)	**0.8717**	**0.8229**	**6.9326**	**0.6095**	**0.9167**	**0.8717**	5.6672	**0.7101**
+ Direction (k=16)	0.8649	0.8209	7.1466	0.6003	0.9119	0.8641	5.7856	0.7047

Table 1. **Comparison results of baseline with respect to different number of direction classes** k**.** The baseline refers to U-Net without direction category supervision (*i.e.*, k = 0).

Mask	Direction	Point	Interaction	MoNuSeg				CPM17			
				F1	Dice	HausD	AJI	F1	Dice	HausD	AJI
√				0.8608	0.8184	7.3329	0.5910	0.8688	0.8473	6.6222	0.6648
√	√			0.8717	0.8229	6.9326	0.6095	0.9167	0.8717	5.6672	0.7101
√	√	√		**0.8740**	0.8241	6.8576	0.6153	**0.9186**	0.8748	5.5377	0.7173
√	√	√	√	0.8708	**0.8277**	**6.6925**	**0.6196**	0.9167	**0.8771**	**5.4968**	**0.7232**

Table 2. **Performance comparison between baseline and the model with direction-guided refinement module (DGM).** √ indicates adding the task or strategy of DGM. Mask: mask supervision; Direction: direction segmentation task; Point: center point detection task; Interaction: interaction of different supervision branches in DGM.

refers to the baseline U-Net, only with 3-class supervision. It can be seen from Table 1 that all the models with direction category supervision ($k = 4, 8, 16$) achieve better performance compared to the baseline without direction category supervision ($k = 0$). Specifically, when the number of direction class is set to 8, the performance on MoNuSeg dataset is increased by 1.09%, 0.45%, 0.40, 1.85% in F1, Dice, HausD, and AJI, respectively, and on CPM17 dataset, the performance is increased by 4.79% F1, 2.44% Dice, 0.96 HausD, and 4.53% AJI. Overall, the results demonstrate that the direction feature helps the nuclear instance segmentation.

Effectiveness of direction-guided refinement module (DGM). Table 2 shows the comparison between the baseline and the model with DGM. First, we investigate whether adding the center point detection task has a positive impact on segmentation performance. From the results in the second row and the third row of Table 2, it is observed that there is an improvement of 0.58% AJI and 0.72% AJI on MoNuSeg dataset and CPM17 dataset respectively, when adding the center point detection task. It indicates that the strategy of introducing point supervision in DGM helps the mask and direction branches to learn more effective features. Furthermore, in order to validate the effectiveness of the interaction of different supervision branches in DGM, we compare the performance of DGM with and without interactive operation. Table 2 shows that using interactive operation in DGM leads to an improvement of 0.43% AJI and 0.59% AJI on MoNuSeg and CPM17 datasets. Therefore, the interaction between different task branches promotes mutual learning and improve the segmentation performance of the proposed CDNet.

Effectiveness of direction difference map (DDM). We
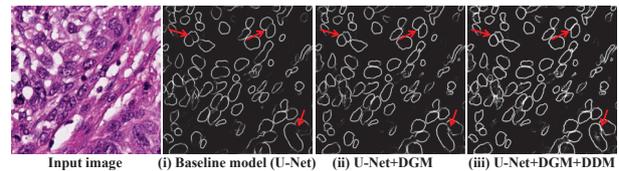
Figure 5. **Qualitative comparison of predicted boundary results.** There is a significant improvement in the boundaries indicated by the red arrows.

obtain DDM by calculating the difference of direction feature between a pixel and its surrounding pixels. In our method, DDM is used in the inference stage to enhance the original boundary probability. In order to verify its effectiveness, we compare the segmentation performance of our method with and without DDM for post-processing. The results in Table 3 show that DDM has a significant contribution to improve segmentation performance in our CDNet. We further perform a qualitative evaluation of DDM on boundary prediction. As shown in Fig. 5, we compare the predicted boundary results of (i) ordinary U-Net, (ii) U-Net with DGM module, and (iii) U-Net with DGM and DDM. The results clearly show that the effectiveness of DGM and DDM on boundary prediction. Specifically, DDM enables a refinement of the results predicted by U-Net with DGM, which helps to enhance the segmentation masks and separate touching objects.

4.4. Comparison with state-of-the-arts

In this section, we provide quantitative and qualitative comparisons between our method CDNet and other nuclei segmentation methods. As shown in Table 4, our CDNet achieves the best performance for nuclei segmentation task. Specifically, our proposed CDNet achieves 0.8316
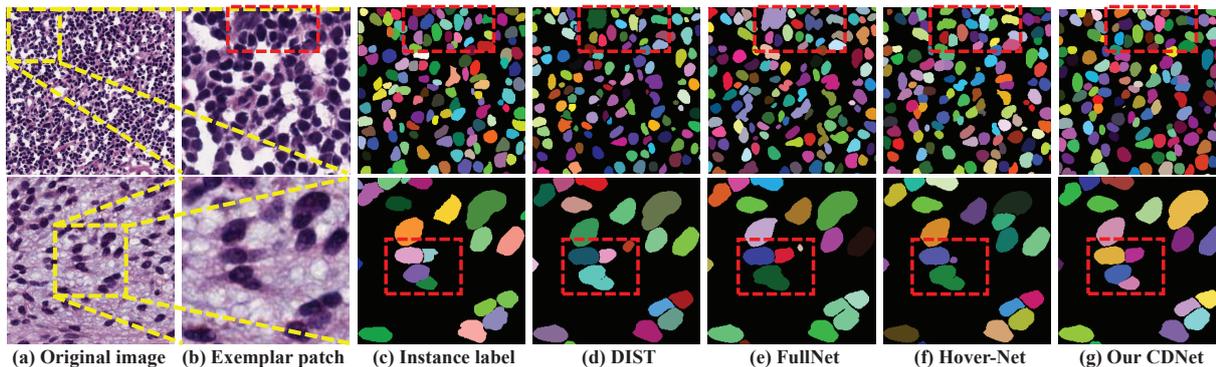
Figure 6. **Visualization of segmentation results on MoNuSeg (top) and CPM17 (bottom) datasets.** (a) Original image; (b) Example patch; (c) Ground Truth; The prediction results of (d) DIST [15]; (e) FullNet [17]; (f) Hover-Net [6]; (g) our proposed CDNet. Different colors indicate different instances in the images. The red rectangles are drawn for clear comparison.

Methods	MoNuSeg				CPM17			
	F1	Dice	HausD	AJI	F1	Dice	HausD	AJI
Baseline	0.8608	0.8184	7.3329	0.5910	0.8688	0.8473	6.6222	0.6648
w/o DDM	**0.8708**	0.8277	6.6925	0.6196	0.9167	0.8771	5.4968	0.7232
$w/$ DDM	0.8705	**0.8316**	**6.4754**	**0.6331**	**0.9237**	**0.8801**	**5.3495**	**0.7326**

Table 3. **Performance comparison between our CDNet with and without direction difference map (DDM) during the inference stage.** Baseline is the benchmark model (U-Net).

Method	MoNuSeg		CPM17	
	Dice	AJI	Dice	AJI
U-Net [19]	0.8362	0.5971	0.8473	0.6648
Mask-RCNN [7]	0.7600	0.5460	0.8500	0.6840
DCAN [1]	0.7920	0.5250	0.8280	0.5610
Micro-Net [18]	0.7970	0.5600	0.8570	0.6680
DIST [15]	0.7890	0.5590	0.8260	0.6160
CIA-Net [26]	0.8180	0.6200	0.8416	0.6648
FullNet [17]	0.8027	0.6039	0.8306	0.6609
Hover-Net [6]	0.8260	0.6180	0.8690	0.7050
BRP-Net [3]	-	**0.6422**	0.8770	0.7310
PFF-Net [11]	0.8091	0.6107	-	-
Our CDNet	**0.8316**	0.6331	**0.8801**	**0.7326**

Table 4. Comparison with SOTAs on MoNuSeg and CPM17.

Dice and 0.6331 AJI on MoNuSeg dataset and 0.8801 Dice and 0.7326 AJI on CPM17 dataset, respectively. The performance of CDNet is obviously improved compared with that of Hover-Net [6] and recently published PFFNet [11]. Note that although BRP-Net surpasses our CDNet in terms of AJI on MoNuSeg, it is a complex two-stage model which requires more computational power than our method.

We further carry out a qualitative visualization analysis on MoNuSeg and CPM17 datasets. As shown in Fig. 6, compared to all the other methods, our CDNet still achieves better results for nuclear instance segmentation. Specifically, it can be observed from the red rectangles in Fig. 6 that our method effectively distinguishes nuclear pix-

els from the background and segment clustered instances. Therefore, quantitative and qualitative results both demonstrate the effectiveness of our CDNet for nuclear instance segmentation.

5. Conclusion

In this paper, we propose a centripetal direction network (CDNet), a simple and effective network designed to address the challenge of touching and overlapping nuclear segmentation. To effectively separate touching and overlapping nuclei instances, we define centripetal direction feature to represent the spatial relationship between pixels in an object. Based on the centripetal direction feature, we further design a direction difference map to measure the direction difference between adjacent pixels for accurate nuclear boundary identification. Finally, by combining the direction feature map, the direction-guided refinement module is used as a plug-and-play module to refine the segmentation masks. The experimental results on two nuclei datasets demonstrate the effectiveness of our proposed CDNet.

References

[1] Hao Chen, Xiaojuan Qi, Lequan Yu, Qi Dou, Jing Qin, and Pheng-Ann Heng. Dcan: Deep contour-aware networks for object instance segmentation from histology images. *Medical image analysis*, 36:135–146, 2017.

[2] Liang-Chieh Chen, Alexander Hermans, George Papandreou, Florian Schroff, Peng Wang, and Hartwig Adam. Masklab: Instance segmentation by refining object detection with semantic and direction features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4013–4022, 2018.

[3] Shengcong Chen, Changxing Ding, and Dacheng Tao. Boundary-assisted region proposal networks for nucleus segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 279–288. Springer, 2020.

[4] Shiv Gehlot, Anubha Gupta, and Ritu Gupta. Ednfc-net: Convolutional neural network with nested feature concatenation for nuclei-instance segmentation. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1389–1393. IEEE, 2020.

[5] Xuan Gong, Shuyan Chen, Baochang Zhang, and David Doermann. Style consistent image generation for nuclei instance segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3994–4003, 2021.

[6] Simon Graham, Quoc Dang Vu, Shan E Ahmed Raza, Ayesha Azam, Yee Wah Tsang, Jin Tae Kwak, and Nasir Rajpoot. Hover-net: Simultaneous segmentation and classification of nuclei in multi-tissue histology images. *Medical Image Analysis*, 58:101563, 2019.

[7] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[8] Qingbo Kang, Qicheng Lao, and Thomas Fevens. Nuclei segmentation in histopathological images using two-stage learning. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 703–711. Springer, 2019.

[9] Neeraj Kumar, Ruchika Verma, Sanuj Sharma, Surabhi Bhargava, Abhishek Vahadane, and Amit Sethi. A dataset and a technique for generalized nuclear segmentation for computational pathology. *IEEE transactions on medical imaging*, 36(7):1550–1560, 2017.

[10] John Lafferty, Andrew McCallum, and Fernando CN Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. 2001.

[11] Dongnan Liu, Donghao Zhang, Yang Song, Heng Huang, and Weidong Cai. Panoptic feature fusion net: A novel instance segmentation paradigm for biomedical and biological images. *IEEE Transactions on Image Processing*, 30:2045–2059, 2021.

[12] Dongnan Liu, Donghao Zhang, Yang Song, Chaoyi Zhang, Fan Zhang, Lauren O'Donnell, and Weidong Cai. Nuclei segmentation via a deep panoptic model with semantic feature fusion. In *IJCAI*, pages 861–868, 2019.

[13] Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. On the variance of the adaptive learning rate and beyond. *arXiv preprint arXiv:1908.03265*, 2019.

[14] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8759–8768, 2018.

[15] Peter Naylor, Marick Laé, Fabien Reyal, and Thomas Walter. Segmentation of nuclei in histopathology images by deep regression of the distance map. *IEEE transactions on medical imaging*, 38(2):448–459, 2018.

[16] Hirohisa Oda, Holger R Roth, Kosuke Chiba, Jure Sokolić, Takayuki Kitasaka, Masahiro Oda, Akinari Hinoki, Hiroo Uchida, Julia A Schnabel, and Kensaku Mori. Besnet: boundary-enhanced segmentation of cells in histopathological images. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 228–236. Springer, 2018.

[17] Hui Qu, Zhennan Yan, Gregory M Riedlinger, Subhajyoti De, and Dimitris N Metaxas. Improving nuclei/gland instance segmentation in histopathology images by full resolution neural network and spatial constrained loss. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 378–386. Springer, 2019.

[18] Shan E Ahmed Raza, Linda Cheung, Muhammad Shaban, Simon Graham, David Epstein, Stella Pelengaris, Michael Khan, and Nasir M Rajpoot. Micro-net: A unified model for segmentation of various objects in microscopy images. *Medical image analysis*, 52:160–173, 2019.

[19] Olaf Ronneberger, Philipp Fischer, and Thomas 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.

[20] Uwe Schmidt, Martin Weigert, Coleman Broaddus, and Gene Myers. Cell detection with star-convex polygons. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 265–273. Springer, 2018.

[21] Luc Vincent and Pierre Soille. Watersheds in digital spaces: an efficient algorithm based on immersion simulations. *IEEE Computer Architecture Letters*, 13(06):583–598, 1991.

[22] Quoc Dang Vu, Simon Graham, Tahsin Kurc, Minh Nguyen Nhat To, Muhammad Shaban, Talha Qaiser, Navid Alemi Koohbanani, Syed Ali Khurram, Jayashree Kalpathy-Cramer, Tianhao Zhao, et al. Methods for segmentation and classification of digital microscopy tissue images. *Frontiers in bioengineering and biotechnology*, 7:53, 2019.

[23] Thomas Wollmann, Manuel Gunkel, Inn Chung, Holger Erfle, Karsten Rippe, and Karl Rohr. Gruu-net: Integrated convolutional and gated recurrent neural network for cell segmentation. *Medical image analysis*, 56:68–79, 2019.

[24] Tiange Xiang, Chaoyi Zhang, Dongnan Liu, Yang Song, Heng Huang, and Weidong Cai. Bio-net: Learning recurrent bi-directional connections for encoder-decoder architecture.

In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 74–84. Springer, 2020.

[25] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017.

[26] Yanning Zhou, Omer Fahri Onder, Qi Dou, Efstratios Tsougenis, Hao Chen, and Pheng-Ann Heng. Cia-net: Robust nuclei instance segmentation with contour-aware information aggregation. In *International Conference on Information Processing in Medical Imaging*, pages 682–693. Springer, 2019.