Segmenting Neuronal Cells in Microscopic Images Using Cascade Mask R-CNN

Fenwei Guo[*]

School of Computer and Information Technology, Beijing Jiao tong University, Beijing 100044, China

[*]Corresponding author: 19722090@bjtu.edu.cn

Keywords: Instance segmentation, Neuronal cell, deep learning, Mask R-CNN, Cascade R-CNN.

Abstract: Delineating precisely the locations of individual neuronal cells in microscopic images is of great significance for the treatment of neurological disorders and neurodegenerative diseases. However, conventional methods for the instance segmentation of neuronal cells suffer from the limited accuracy, lack of automation, and time intensive processes. To address this challenge, we propose an R-CNN-based deep learning model for the segmentation of neuronal cells with a promising performance in this paper. The architecture of our model is the Cascade Mask R-CNN, which is a combination of the Mask R-CNN and Cascade R-CNN. In this model, a ResNeXt + FPN backbone with standard convolution and fully connected heads is utilized for the mask prediction, where the ResNeXt part of backbone is ResNeXt-152-32x8d. The model is pretrained based on the LIVEcell dataset, and subsequently trained using the dataset provided by Sartorius in a Kaggle competition. By a boost from the pseudo-label technique, our model can achieve a mAP@.5:.95 score 0.338 on the private test set. Such a score locates at 36/1505 (top 3%) in the leaderboard of Sartorius - Cell Instance Segmentation competition, and can get a silver medal in this Kaggle competition. Our results could help the researchers measure the effects of neurological disorders more easily, and potentially accelerate the discovery and development of new drugs for the treatment of neurodegenerative diseases.

1. Introduction

Neurological disorders, which affect as many as one billion people worldwide, can lead to a range of symptoms and are a leading cause of death and disability across the globe [1]. As the review of neuronal cells via light microscopy is both accessible and non-invasive, the instance segmentation of neuronal cells in microscopic images plays a crucial role for the treatment of neurological disorders [2]. Therefore, effective methods to detect and delineate the locations of neuronal cells could help the researchers measure the effects of neurological disorders more easily. However, conventional segmentation methods have limited accuracy for neuronal cells and are usually time-intensive, leading to a great demand for an automated and valid approach to segment neuronal cells in microscopic images.

On the other hand, computer vision techniques are undergoing a rapid development since machine learning has witnessed an unprecedented revolution in recent years [3-10]. In particular, deep convolutional neural networks (CNNs) are proven to have an incredible ability to automatically accomplish complicated tasks regarding medical images. A wide array of applications of CNNs have been reported in the fields of healthcare and medical-image processing, such as the diagnosis of skin cancer [11], identification of cardiovascular risk [12], and detection of pneumonia [13]. Therefore, CNNs are highly qualified to achieve the instance segmentation of neuronal cells with a promising performance.

For that purpose, two representative CNN models based on the region-based convolutional neural network (R-CNN) are selected in this work: Mask R-CNN [8] and Cascade R-CNN [9]. The model architecture we use is the Cascade Mask R-CNN, which is a combination of these two representative models. In particular, a ResNeXt + FPN backbone with standard convolution and fully connected heads is utilized for the mask prediction, where the ResNeXt part of backbone is ResNeXt-152-32x8d [4, 6]. By using two datasets provided by Sartorius, this model reaches an mAP@.5:.95 score 0.338 in the Sartorius - Cell Instance Segmentation competition [14]. Such score ranks 36/1505 (top 3%) in the Kaggle leaderboard [15], and can get a silver competition medal.

The rest of the paper is organized as follows. Two datasets provided by Sartorius, including the LIVEcell dataset and the dataset in the Kaggle competition, are briefly introduced in Section 2. The mechanisms of ResNeXt, FPN, Mask R-CNN, and Cascade R-CNN are summarized in Section 3. In the following Section 4, the workflow of our model and corresponding training schedule are given in detail. The results and model performance are shown in Section 5. Finally, we draw a conclusion in Section 6.

2. Data Description

In this work, we use two datasets to train the Cascade Mask R-CNN model. These datasets are both provided by Sartorius, which is a famous international pharmaceutical and laboratory equipment supplier in Germany. One is the dataset in the Sartorius - Cell Instance Segmentation competition, the other is the LIVEcell dataset [16].

The competition dataset has 606 images in the training set and roughly 240 images in the test set (3 samples is provided publicly, others are hidden in the Kaggle backend). In addition, 1972 unlabeled images are offered in this dataset. This dataset consists of three different kinds of neuronal cells, i.e., cort, astro, and shsy5y, where representative examples and corresponding masks are demonstrated in Figure 1. On the other hand, 9 kinds of neuronal cells are in the LIVEcell dataset, with 4184 images in the training set and 1664 images in the test set. All images in both datasets have the same pixel size 704 × 520, with masks stored as run length encoded pixels.

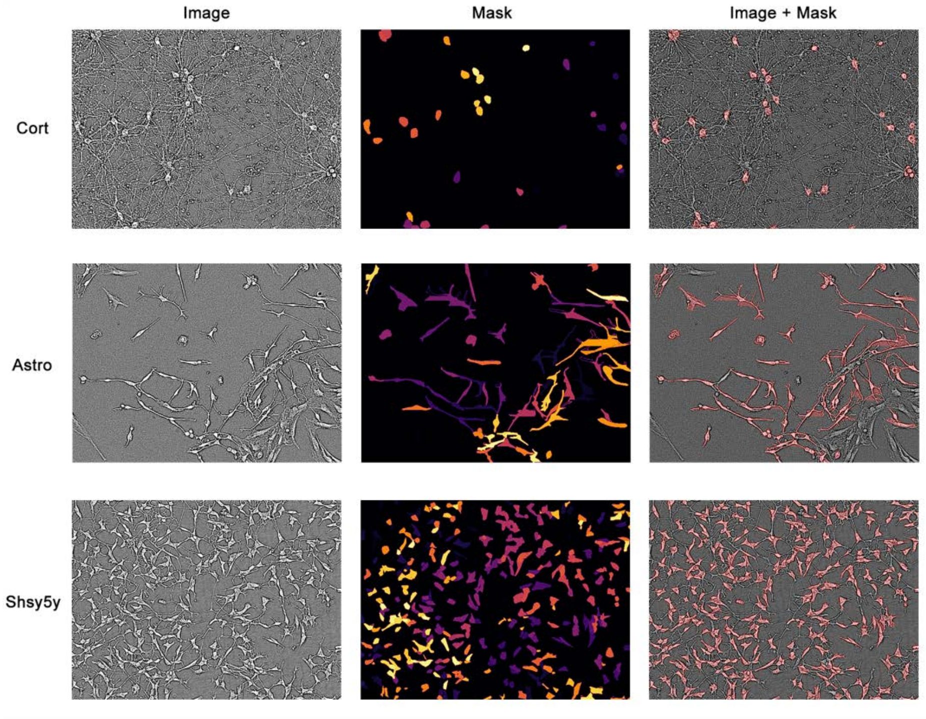

Figure 1: Representative images and corresponding masks of three different kinds of neuronal cells, cort, astro, and shsy5y, in the competition dataset.

Moreover, the evaluation metric in this Kaggle competition is the mean average precision at different intersection over union (IoU) thresholds [14]. The IoU of a predicted mask and corresponding ground truth is given by:

$$\mathrm{IoU} = \frac{\mathrm{Pred} \cap \mathrm{Target}}{\mathrm{Pred} \cup \mathrm{Target}} \tag{1}$$

In particular, the competition metric sweeps over a range of IoU thresholds from 0.5 to 0.95 with a step size of 0.05, at each point calculating an average precision value. For a given threshold value *t*, the precision value P_t is written as:

$$\mathrm{P}_t = \frac{\mathrm{TP}(t)}{\mathrm{TP}(t)+\mathrm{FP}(t)+\mathrm{FN}(t)} \tag{2}$$

Where TP, FP, and FN denote the true positive, false positive, and false negative samples respectively. Then the average precision value of a single image is the mean of precision values at each IoU threshold. We denote this value as AP@.5:.95, which is calculated as:

$$\mathrm{AP}@.5:.95 = \frac{\mathrm{P}_{0.50} + \mathrm{P}_{0.55} + \cdots + \mathrm{P}_{0.95}}{10} \tag{3}$$

After that, the overall metric mAP@.5:.95 in the competition leaderboard is the mean of AP@.5:.95 taken over all images in the test set.

3. Methods

The crucial machine learning blocks in the architecture of Cascade Mask R-CNN are the ResNeXt [4], FPN (feature pyramid network) [6], Mask R-CNN [8] and Cascade R-CNN [9]. Hence, the mechanisms of these models will be briefly summarized in this section.

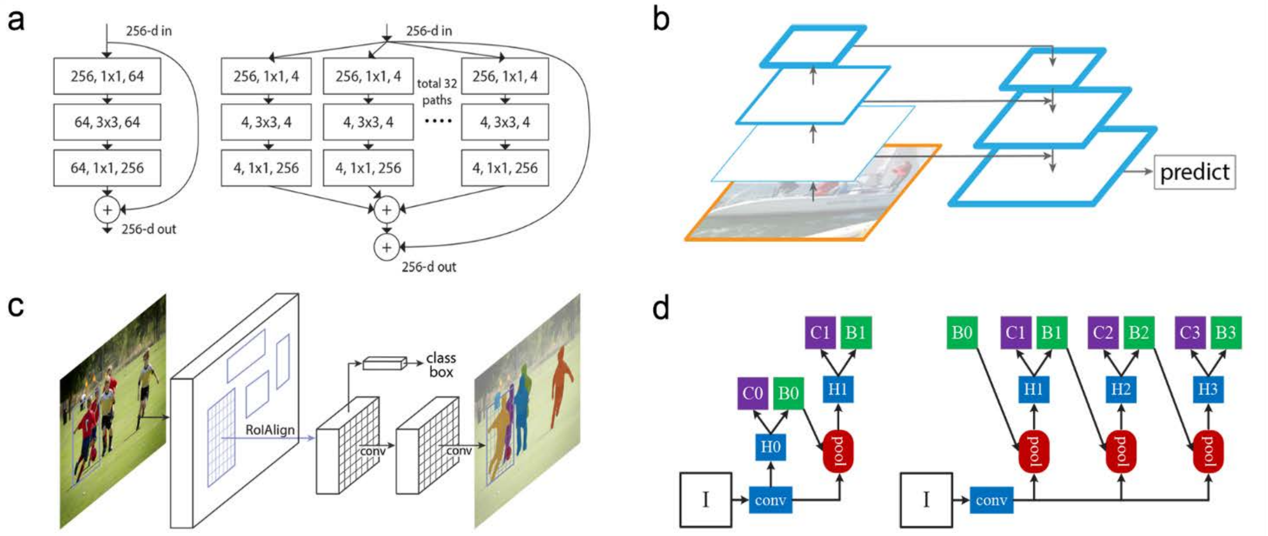

Figure 2: (a) Comparison between the structures of ResNet and ResNeXt [4]. (b) Schematic diagram of the FPN architecture [6]. (c) Workflow of Mask R-CNN [8]. (d) The difference of structures in the second stage between Faster R-CNN and Cascade R-CNN [9].

3.1 ResNeXt and FPN

The mechanism of ResNeXt is depicted in Figure 2a [4], which is an improved version of the well-known CNN backbone ResNet [3]. As shown in the right of Figure 2a, aggregated transformations are adapted to the second convolution layer of each bottleneck block in ResNeXt, where the number of transformations is called as cardinality. The backbone we choose in this work is the ResNeXt-152-32x8d, which is a ResNeXt network with layer depth = 152, cardinality = 32, and input channel dimension = 8.

Feature maps extracted by the CNN backbone is processed by the FPN for the purpose of instance segmentation [6]. As demonstrated in Figure 2b, the architecture of FPN is a top-down structure with lateral connections for building high-level semantic feature maps at all scales. It has the inherent multi-scale, pyramidal hierarchy of deep convolutional networks to construct feature pyramids with marginal extra cost. Such architecture allows the model to accomplish the segmentation tasks with efficient cost of computing resources.

3.2 Mask R-CNN and Cascade R-CNN

Mask R-CNN [8] is a kind of region-based convolutional neural networks and is built on the top of Faster R-CNN [7]. It is designed for image segmentation and was the state-of-the-art segmentation model in the past years. Based on the structure of Faster R-CNN which has two outputs for a class label and a bounding-box offset, Mask R-CNN introduces a third output that predicts the object mask (Figure 2c). It is realized by adding only a small overhead to Faster R-CNN, thereby making Mask R-CNN efficient and simple to train. In particular, Mask R-CNN undergoes a two-stage procedure when doing the segmentation tasks. The model proposes multiple objects using the region proposal network (RPN) in the first stage, and outputs the predictions, including class, box offset, and mask for each region of interest pooling (RoI) in the second stage.

Unlike Mask R-CNN which is originally designed for image segmentation, Cascade R-CNN is a deep learning model dealing with object detection problems [9]. This architecture aims to address the degrading performance with increased IoU thresholds due to overfitting during training. Considering the evaluation metric (mAP@.5:.95) in this competition, this model can be highly qualified for the segmentation task in this work. As shown in Figure 2d, different from Faster R-CNN using only one head in the second stage, different heads are used at different stages in Cascade R-CNN. Each of heads is designed for one specific IoU threshold from small to large. In particular, the cascaded regression in this model is a resampling procedure, which can provide good positive samples to the next stage. The architecture of Cascade R-CNN can also be applied to FPN, making this model able to deal with segmentation tasks when combining with the Mask R-CNN [10].

4. Workflow and Training SCHEDULE

Schematic diagram of the Cascade Mask R-CNN model in this work is exhibited in Figure 3a. It can be found that a ResNeXt + FPN backbone is utilized in the first stage to extract feature maps and region proposals, where the ResNeXt part of backbone is ResNeXt-152-32x8d. In the second stage, a segmentation branch is added to each cascade stage to allow the Cascade R-CNN output mask predictions. The code of this model is implemented by Detectron2, which is a library of state-of-the-art detection and segmentation algorithms and is provided by Facebook AI Research [17].

The training schedule of this model is summarized in Figure 3b and briefly introduced here. A pre-training process is firstly conducted using the LIVEcell dataset to obtain appropriate pretrained model weights. In particular, despite that there are 9 kinds of neuronal cells in this dataset, we regard all neuronal cells as one kind and make the model deal with a 1-class segmentation task. All data in the training set are used for training (without evaluation set) in the pre-training process. The training is realized by 8 × RTX 3090 Nvidia GPU, and lasts 5000 iterations with a cosine annealing schedule of learning rate [18]. Here, we set the max (initial) learning rate $\eta_{\max} = 0.005$ and minimum learning rate $\eta_{\min} = 10^{-5}$. The period of this scheduler $T_{\max}$ = 200. When applying this scheduler, the learning rate η_t at each iteration is expressed as:

$$\eta_t = \eta_{\min} + \frac{1}{2}\left(\eta_{\max} - \eta_{\min}\right)\left[1 + \cos\left(\frac{T_{\mathrm{cur}}}{T_{\max}}\pi\right)\right]. \tag{4}$$

Where T_{cur} is the number of current iteration. The weight of the final iteration is chosen as the pretrained weight for the next process.

After that, we utilize the competition dataset to train our model. 80% data is used for training and the other 20% is used for evaluation. The model in this process is trained to accomplish a 3-class (cort, astro, and shsy5y) segmentation task, where the training method is the same with that in the pre-training process. Considering that the amount of unlabeled data is much larger than labeled data in the competition dataset, pseudo-label technique [19] is applied as a semi-supervised approach to improve the model performance. In detail, we use the trained model to generate predicted masks on these unlabeled microscope images. Next,

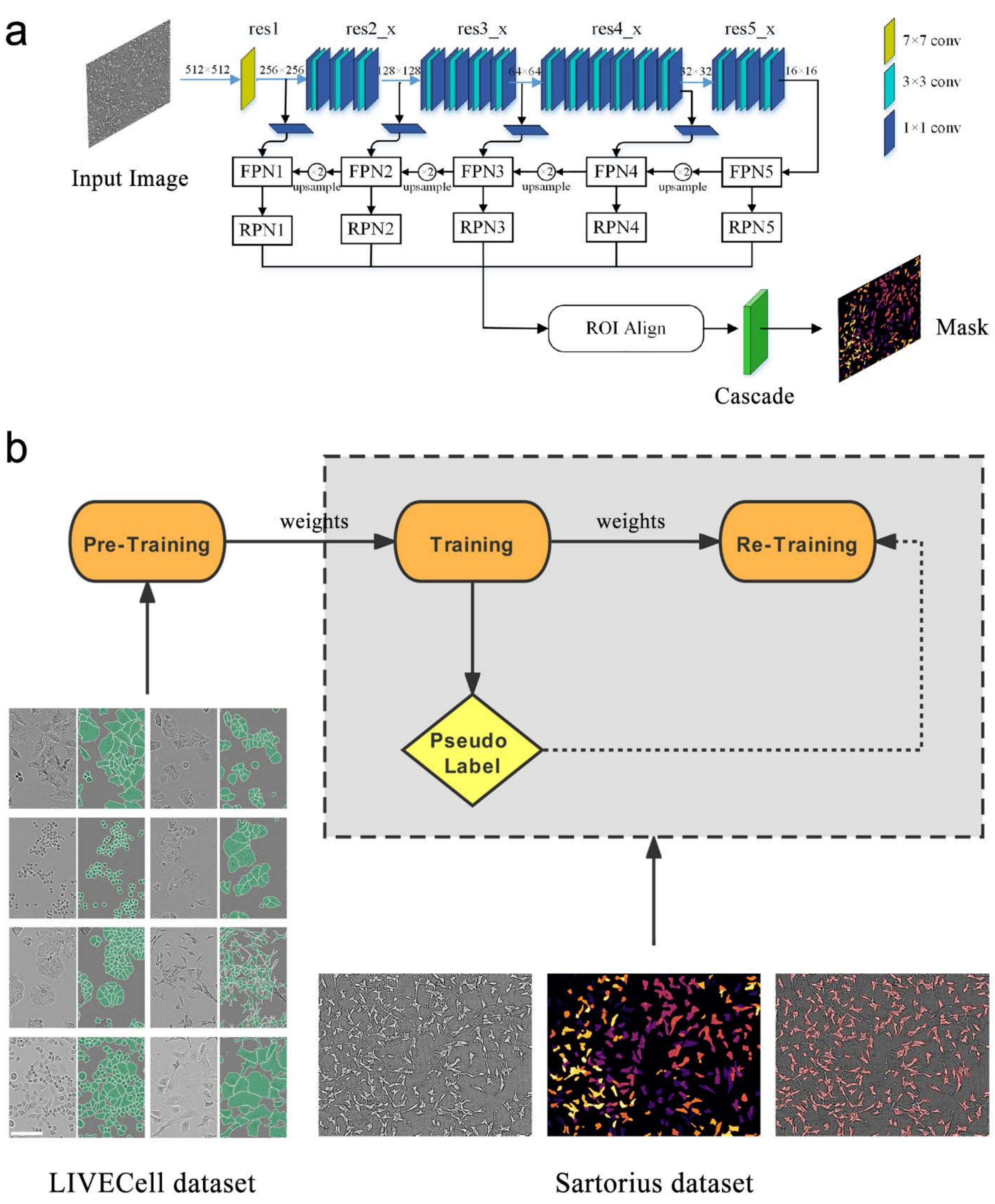

Figure 3: (a) Schematic diagram of the Cascade Mask R-CNN model in this work, using a ResNeXt + FPN backbone in the first stage and cascade blocks in the second stage. (b) The training schedule of our model, which is composed of three processes, i.e., pre-training, training, and re-training, respectively. LIVEcell dataset is used in the pretraining process, while the competition dataset is used in the other two processes. Pseudo-label technique is applied in the re-training process to improve the model performance.

These unlabeled data with predicted masks are added to the training set to re-train our model, leading to the model with best performance for the instance segmentation of neuronal cells.

5. Results and Model Performance

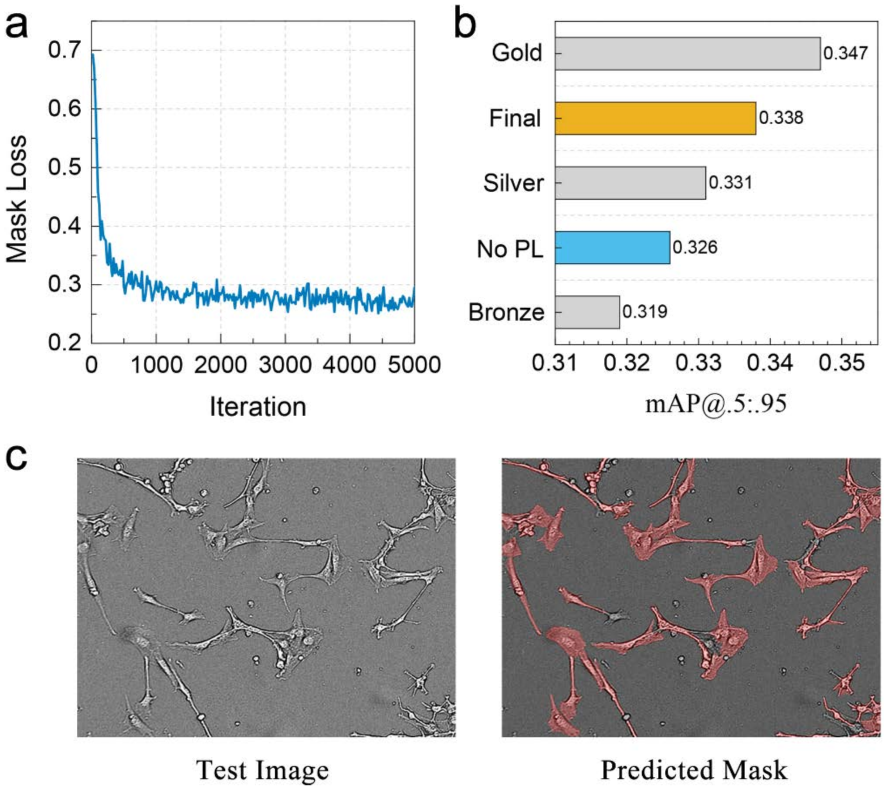

Figure 4: (a) The evolution of mask loss in the training process. (b) Performance of our model on the private test set (leaderboard score), where the benchmarks for gold, silver, and bronze medals are also given. (c) Image in the test set and corresponding predicted mask.

To examine the model convergence of this Cascade Mask R-CNN model, we show the evolution of mask loss in the training process in Figure 4a. It can be found that the loss can finally reach a small steady value, suggesting that our model performs well on the training set. Next, the performance of our model on the private test set (leaderboard score) is given in Figure 4b, where the benchmarks for gold, silver, and bronze medals are also given. The result shows that the performance of our model without the boost of pseudo-label technique (0.326) can surpass the bronze benchmark. Moreover, after applying the pseudo-label technique, this model reaches an mAP@.5:.95 score 0.338. Such a score ranks 36/1505 (top 3%) in the leaderboard of Sartorius - Cell Instance Segmentation competition [15], and can get a silver medal in this Kaggle competition. In addition, to demonstrate the model performance more intuitively, we show the predicted mask on a sample image in the test set in Figure 4c. It can be found that most cells in this image are appropriately segmented, indicating that our Cascade Mask R-CNN model can be qualified for the instance segmentation of neuronal cells in microscopic images.

6. Conclusion

In summary, by combining Mask R-CNN and Cascade R-CNN, we use the Cascade Mask R-CNN model to successfully develop an automated deep learning approach for the instance segmentation of neuronal cells in microscopic images with a promising performance. By a boost from the pseudo-label technique, our model can achieve a mAP@.5:.95 score 0.338 on the private test set in the Sartorius - Cell Instance Segmentation competition. Such a score ranks 36/1505 (top 3%) in the leaderboard, and can get a silver medal in this Kaggle competition. Our results could help the researchers measure the effects of neurological disorders more easily, and potentially accelerate the discovery and development of new drugs for the treatment of neurodegenerative diseases.

Acknowledgments

We thank Kaggle and Sartorius for providing the LIVEcell dataset and competition dataset regarding the instance segmentation of neuronal cells.

References

[1] T. Sanders, Y. Liu, V. Buchner, and P. B. Tchounwou. Neurotoxic Effects and Biomarkers of Lead Exposure: A Review. Rev. Environ. Health 24: 15-45, 2009.

[2] J. Wu, et al. Kilohertz Two-Photon Fluorescence Microscopy Imaging of Neural Activity in vivo. Nat. Methods 17: 287-290, 2020.

[3] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. In CVPR, 2016.

[4] S. Xie, et al. Aggregated Residual Transformations for Deep Neural Networks. In CVPR, 2017.

[5] M. Tan and Q. V. Le. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In ICML, 2019.

[6] T. -Y. Lin, et al. Feature Pyramid Networks for Object Detection. arXiv: 1612.03144.

[7] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. arXiv: 1506.01497.

[8] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask R-CNN. arXiv: 1703.06870.

[9] Z. Cai and N. Vasconcelos. Cascade R-CNN: Delving into High Quality Object Detection. arXiv: 1712.00726.

[10] Z. Cai and N. Vasconcelos. Cascade R-CNN: High Quality Object Detection and Instance Segmentation. arXiv: 1906.09756.

[11] A. Esteva, et al. Dermatologist-Level Classification of Skin Cancer with Deep Neural Networks. Nature 542: 115-118, 2017.

[12] H. Chao, et al. Deep Learning Predicts Cardiovascular Disease Risks from Lung Cancer Screening Low Dose Computed Tomography. Nat. Commun. 12: 2963, 2021.

[13] R. Kundu, et al. Pneumonia Detection in Chest X-ray Images Using an Ensemble of Deep Learning Models. PLoS ONE 16: e0256630, 2021.

[14] Public available at the Kaggle platform: https://www.kaggle.com/c/sartorius-cell-instance-segmentation/

[15] Guo Fenwei: https://www.kaggle.com/guofenwei/

[16] C. Edlund, et al. LIVECell - A Large-Scale Dataset for Label-Free Live Cell Segmentation. Nat. Methods 18: 1038-1045, 2021.

[17] Detectron2: https://ai.facebook.com/tools/detectron2/

[18] I. Loshchilov and F. Hutter. SGDR: Stochastic Gradient Descent with Warm Restarts. In ICLR, 2017.

[19] D. H. Lee. Pseudo-Label: The Simple and Efficient Semi-Supervised Learning Method for Deep Neural Networks. In ICML, 2013.