Frontiers in Computing and Intelligent Systems
ISSN: 2832-6024 | Vol. 1, No. 1, 2022

DisCnet: Pixel segmentation based on discrete features

Zhihui Li[1, *], Xiaoshuo Jia[2]

[1] School of Computer Science, Guangdong University of Science and Technology, Dongguan 523079, Guangdong, China
[2] Electrical Engineering Department, University of Colorado, Boulder, CO 80309, USA

[*] **Corresponding author**: Zhihui Li.

Abstract. In the process of pixel-level semantic segmentation tasks, traditional image processing algorithms will suffer from the working mechanism of convolutional layers and pooling layers, resulting in the loss of some features, which now leads to inaccurate semantic segmentation accuracy. For such problems, we design a discrete pooling layer by analyzing the distribution and statistical properties of discrete data. Compared with the traditional pooling layer, the discrete pooling layer can not only preserve the spatial information of features, avoid the loss of features, but also can efficiently improve the accurate segmentation of instance images. Then, based on the discrete pooling layer, we design DisCnet in combination with the convolutional layer. Finally, DisCnet is compared with some state-of-the-art algorithms under the Cityscapes dataset. Experiments demonstrate that DisCnet achieves excellent results in both accuracy and speed.

Keywords: Semantic segmentation, Image processing, Pooling layer, Discrete feature.

1. Introduction

In image processing, pixel-level semantic segmentation [1-3] is an extremely important and complex task. The CNN algorithm [4-7] has achieved excellent results in image classification, segmentation, tracking and other aspects of Kaggle and AI Challenger competitions using the characteristics of multi-parameters. FCN uses deconvolution for upsampling to make the extracted features more detailed. U-Net uses the network symmetric structure to fuse high-dimensional features and low-dimensional features to weight edge features. In CPFNet, the dilated convolution is proposed, which can expand the field of view of the convolutional layer to extract more feature information, and then combine the inception module to achieve context-based feature fusion, and achieve superior results in medical datasets. STDC is based on FPN for the fusion of multiple scales, so its performance is superior to the CPFnet algorithm. BiseNetV2 adopts a bilateral segmentation structure on the basis of STDC, namely Detail Branch and Semantic Branch. Detail Branch obtains more low-level feature information by expanding the channel, and Semantic Branch expands the receptive field through a lightweight convolution layer to obtain high-level feature information. At the same time, the problem of structural redundancy is also solved. Although the CNN-based algorithm has a high accuracy, it is common that the extracted features lose a lot of spatial information due to the pooling layer. Eventually, the semantic segmentation network structure redundancy, large amount of computation, segmentation errors and other problems appear.

Here we discrete the statistical properties and distribution properties of the data to the data distribution of image edge features, and design the discrete network DisCnet. DisCnet extracts the features with larger discreteness in the image by analyzing the distribution characteristics of discrete data, that is, edge features. After DisCnet extracts the edge features of the image, the edge features are regressed to locate the edge contour of the target, and then achieve accurate semantic segmentation. Under the Cityscapes data set, DisCnet and SOTA algorithm are compared. The experimental results show that DisCnet has certain advantages in terms of accuracy and speed.

2. Method

The input feature passes through the sliding window to obtain n feature maps Pi (i=1, 2..., n) with the same dimension h*w and different eigenvalues. Through formula 1, we can obtain the correlation coefficient between the feature maps Pm and Pm+1, expressing the correlation between the two feature maps. The eigenvalues ri, j of the last n positions forms a new feature map r, which ensures the correlation between the eigenvalues.

$$r_{(i,j)} = \frac{\sum(P_m|(i,j) - \overline{P}_m|(i,j))(P - \overline{P}_{m+1})}{\sqrt{(\sum(P_m|(i,j) - \overline{P}_m|(i,j))^2)(\sum(P_{m+1} - \overline{P}_{m+1})^2)}} \quad (1)$$

$$\mathrm{n} = \left\lfloor \frac{H - h + 2 * p}{s} + 1 \right\rfloor * \left\lfloor \frac{W - w + 2 * p}{s} + 1 \right\rfloor \quad (2)$$

$$\mathrm{R} = r * P \quad (3)$$

Pm|(i,j) is the feature map with point (i,j) as the upper left corner, $\overline{P}_m|(i,j)$ is the mean of the feature map Pm. Pm+1 is the next feature map of the Pm. Then, the feature map r is dot-multiplied with the P, and finally the feature map R pooled by the Rel layer is obtained.

Table 1. The network structure parameters are shown in the following table

Layer	Kernel/stride	Parameters
Conv1	128*11*11/4	15488
Dis pool	3*3/1	-
Conv3/5/7/9/11/13	8*1*3/2 8*3*1/2	3072
Rel pool	3*3/2	-
Conv4/8/12	16*3*3/2	1152
Conv2/6/10	64*11*11/4	61952

The input data is multiplied by the discrete coefficient obtained by the Dis pooling layer. The purpose is to calibrate the spatial position of the edge feature through the discrete coefficient. The result obtained here is then added to the input data. The purpose It enhances the information of edge features on the one hand, and preserves the correlation between features on the other hand. Therefore, the point multiplication is to calibrate the spatial position of the edge features, and the

purpose of addition is to strengthen the information of the edge features and preserve the correlation between the features. The DisCnet structure refers to the residual effect of Resnet, which effectively extracts edge features on the one hand, and makes the model more lightweight on the other hand.

3. Experiments

3.1. Dataset

Cityscapes contains a total of 5000 fine images, of which 2975 are training images, 500 validation images and 1525 testing images. In addition, the dataset contains 20k roughly annotated images.

3.2. Comparison with SOTA

We first trim the images of Cistyscapes datasets to a size of 500*500, and set the initial learning rate to 1*e-5 and the epoch to 12000. The Cityscapes dataset has 2975 training images and 1525 testing images. The training platforms are Ryzen 7 3800X and RTX 2070. The optimization function is Adam optimizer. The loss function uses formula 4 to calculate the error between the true value and the predicted value, and uses IoU to evaluate the test results. yp and yt represent the predicted values and actual values respectively.

$$\mathrm{dice}(p,t) = 2 * |y_p \cap y_t| / (|y_p| + |y_t|) \quad (4)$$

Here we compare Resnet with some SOTA algorithms, such as Deep snake [8], Unet[9], PANet[10], FCIS[11], ESE[12-13], etc. The results for the Cityscapes datasets are shown in Table 2 below.

Table 2. Comparison results of Cityscapes datasets.

Network	DisCnet	Deep	UNet	PANet	FCIS	ESE	SegNet
AUC (%)	38.6	37.4	38.4	36.5	29.4	18.6	13.2
Fps	18.6	4.6	8.6	7.5	10.2	18.7	16.5

Judging from the accuracy results of the Cityscapes datasets, Resnet can achieve a good result. UNet and PANet can enhance the features of corresponding locations by concatenating data dimensions. Through feature splicing, the high-resolution features are enhanced by the low-resolution features, so the edge features can be accurately segmented by locating the enhanced features. However, due to the problem of feature loss in the pooling layer, the segmentation position is inaccurate. FCIS and ESE will be trained on the basis of fully convolutional network by means of encode-decode. FCIS can effectively avoid the problem of inaccurate information caused by the loss of feature information in the pooling layer, but it will also reduce the calculation speed due to the excessive number of convolutional layers. Here we extract relevant feature about the edge features of the image through the Rel layer. Resnet uses these features to enhance the edge feature and achieve feature positioning, and then achieve the effect of image segmentation, as shown in Figure 5. From the comparison results in Figure 1, it can be directly seen that Resnet can accurately segment the target edge. Unet and Deep snake cannot accurately locate the fine boundary contour, and also have the problem of inaccurate segmentation for small volume targets. It can be seen from the comparison results of segmentation renderings and accuracy that Resnet has certain advantages.

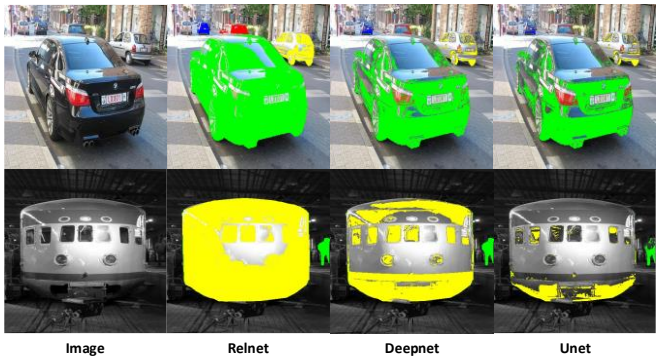

Figure 1. Image represent the original image. And Resnet, Deepnet, Unet correspond to the segmentation effect image of these algorithms respectively.

4. Conclusion

In this paper, we extract discrete features in the image by analyzing the statistical characteristics of discrete data, and design the corresponding discrete pooling layer, and then combine the residual structure to design DisCnet. Then we conduct a comprehensive comparison with some SOTA algorithms under the Cityscapes datasets, demonstrating that DisCnet performs well in both accuracy and model size.

Acknowledgements

This work is supported by Natural Science Program of Guangdong University of Science and Technology under the Grant No.GKY-2021KYQNK-3.

References

[1] Liu S, Qi L, Qin H, et al. Path aggregation network for instance segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 8759-8768.

[2] Li Y, Qi H, Dai J, et al. Fully convolutional instance-aware semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 2359-2367.

[3] Xu W, Wang H, Qi F, et al. Explicit shape encoding for real-time instance segmentation[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019: 5168-5177.

[4] Yuan Z W, Zhang J. Feature extraction and image retrieval based on AlexNet[C]//Eighth International Conference on Digital Image Processing (ICDIP 2016). SPIE, 2016, 10033: 65-69.

[5] Iandola F N, Han S, Moskewicz M W, et al. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size[J]. arXiv preprint arXiv:1602.07360, 2016.

[6] Szegedy C , Vanhoucke V , Ioffe S , et al. Rethinking the Inception Architecture for Computer Vision[J]. IEEE, 2016:2818-2826.

[7] Howard A G, Zhu M, Chen B, et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications[J]. arXiv preprint arXiv:1704.04861, 2017.

[8] Peng S, Jiang W, Pi H, et al. Deep snake for real-time instance segmentation[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 8533-8542.

[9] Zhao X, Vemulapalli R, Mansfield P A, et al. Contrastive learning for label efficient semantic segmentation[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 10623-10633.

[10] Armato S G , Roberts R Y , Mcnitt-Gray M F , et al. The Lung Image Database Consortium (LIDC) and Image Database Resource Initiative (IDRI): a completed reference database of lung nodules on CT scans.[J]. Academic Radiology, 2007, 14(12):1455-1463.

[11] Jetley S, Sapienza M, Golodetz S, et al. Straight to shapes: real-time detection of encoded shapes[C]//Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. 2017: 6550-6559.

[12] Ze Yang, Yinghao Xu, Han Xue, Zheng Zhang, Raquel Urtasun, Liwei Wang, Stephen Lin, and Han Hu. Dense reppoints: Representing visual objects with dense point sets. arXiv preprint arXiv:1912.11473, 2019.

[13] Xingyi Zhou, Dequan Wang, and Philipp Kr¨ahenb¨uhl. Objects as points. Arxiv preprint arxiv:1904.07850, 2019.