Journal of Artificial Intelligence Research, vol. 1, no.1, pp. 36-40, OCTOBER 2024

Channel Reorganization for Few-Shot Segmentation

Weiyang Liu[a]

[a] *School of Media Science, Northeast Normal University, Changchun, China*

ARTICLE INFO

Keywords:
few-shot segmentation
graph convolution network
channel reorganization

ABSTRACT

Few-shot segmentation methods are often modeled as two branch convolutional neural networks, namely support branch and query branch. Existing methods often rely too much on support images. They ignore the power of the query image and fail to fully learn the information of the query image. In addition, we all know that convolution extracts information features by fusing spatial and channel features in local receptive fields. However, most of the existing methods extract information by fusing spatial features, ignoring the role of channel features in information extraction. To address the issues, we propose a new semantic segmentation module based on channel reorganization graph convolution network (CRGCN). First, we construct the graph structure according to the channel features, then screen the beneficial structures based on motif, and finally use GCN to recombine the channel features. This can sufficiently mine the potential relationship between the features of the query image. Experiments on $PASCAL-5^i$ and FSS-1000 datasets show that our proposed method is superior to the baseline and state-of-the-art method.

1. Introduction

Nowadays, there are many problems in computer vision task, such as high cost and long time consuming of data collection, which makes it difficult to extend the current method to the unseen class. In order to solve the data problem, people began to try few-shot learning. The target of semantic segmentation based on few-shot learning is to segment new class targets by using a small number of labeled images.

Existing few-shot segmentation methods are often modeled as two branch convolutional neural networks, namely support branch and query branch [12, 17, 20]. The support branch is used to extract the segmentation prior of the support image, and the query branch is used to extract the features of the query image in the embedded space. According to the segmentation prior of the support branch, the similarity relationship between the two is constructed to realize the segmentation of the query image. Based on this infrastructure, various methods solve problems by proposing different modules. PFENet proposed feature enrichment modules to overcome the problem of spatial inconsistent [15]; CyCTR proposed a cyclic consistency transformer, which makes full use of foreground and background information and removes the interference of harmful information in supporting images [22]; PCCNet proposed a pyramid co-attention compare network to alleviate the problem of differences in the appearance of objects in query images [21]; SAGNN proposed a scale-aware graph neural network to capture cross scale relationships and overcome object changes (e.g., appearance, scale and location) [18]; ASR proposed semantic span module and semantic filtering module to solve the problem of semantic similarity [7].

Through analysis, we find that the above methods often rely too much on supporting images. They either fully learn the foreground and background information of supporting images, or reduce the differences between supporting query images by means. However, they ignore the power of the query image itself and fail to fully learn the information of the query image. In addition, convolution is widely used in the above methods. We all know that convolution extracts information features by fusing spatial and channel features in local receptive fields. However, most of the above methods extract information by fusing spatial features, ignoring the role of channel features in information extraction.

To solve the above two problems, a new semantic segmentation module based on channel reorganization graph convolution network is proposed to reorganize the channel features and fully mine the query image information. In this module, each channel is regarded as a node, and the distance based method is applied to construct the graph, and the discrete channel features are modeled as channel feature graphs. Then calculate the number of motifs in the graph, and select the most number of motifs as the structural features, which can suppress the structures with adverse effects. Finally, GCN is used to reorganize the relationship between channels, enhance the interdependence between channels, and effectively aggregate the features in different channels. The use of GCN can mine the potential relationship between channel features, capture the similarity between the abstract features of similar objects in the image, and make the query information more fully utilized.

We conducted extensive performance evaluation experiments on $PASCAL-5^i$ and FSS-1000 datasets. A large number of experimental results show that the mIoU value of this method is higher than the baseline and some existing few-shot semantic segmentation models. In addition, we also conducted a series of ablation experiments to prove the advantages of the channel reorganization module and the use of GCN.

The contributions of this paper are listed as follows:

DOI: https://doi.org/10.70891/JAIR.2024.100025
ISSN of JAIR: 3078-5529

- This paper proposes a few-shot semantic segmentation method based on channel reorganization graph convolution network. The graph convolution network is used to fully mine the potential relationship between channel features and make full use of query features.
- In the field of few-shot semantic segmentation, as far as we know, we are the first person to use channel features to build graph structure and use graph convolution network, which provides a new idea for using graph learning to solve the problem of few-shot semantic segmentation.

2. Related Works

2.1. Few-shot Semantic Segmentation.

Few-shot semantic segmentation (FSS) requires a few labeled samples to mark the new image with dense pixels. As the first research in this field, OSLSM proposed a two-branch method that uses the weight parameters learned by conditional branch to segmentation branch for query image segmentation [12]. Inspired by the prototype network [13], PL introduced prototype learning into few-shot semantic segmentation, and predicts by comparing the similarity between the prototype and pixels of each class [2]. Then, some methods based on two-branch architecture began to study how to build prototypes in different ways to achieve better segmentation effect. Representative methods include SGOne [23], PANet [17], FWB [9], CANet [20], PPNet [8], PMM [19] and ASGNet [6]. However, the prototypes constructed by these methods are usually limited and have low generalization, which may lead to the loss of useful support information and damage the segmentation of query images.

Recently, people began to try new methods to improve the segmentation effect of few-shot semantic segmentation. PFENet established the support-query relationship with the help of prior knowledge of high-level feature generation, and proposed a top-down feature enrichment module for feature fusion [15]. Both DAN [16] and PCCNet [21] use the attention mechanism. The difference is that DAN introduced the democratized graph attention mechanism to activate more pixels on the foreground object. PCCNet introduced the pyramid co-attention model, which uses attention to pick up similar features from the opposite to make the corresponding features closer. CyCTR used two improved transformers [22]. On the one hand, it aggregates the query features separately, on the other hand, it cross aligns the foreground and background to aggregate the support image features onto the query image features. SAGNN constructed a scale-aware graph neural network based on different scale features after feature fusion [18]. Through message passing on the graph, SAGNN captures cross-scale relations and overcomes object variations.

2.2. Graph Convolution Network.

Graph convolution network (GCN) applies the convolution operation to the data of graph structure, and collects the information of nodes and their neighbors, so as to achieve the purpose of extracting graph spatial features [5]. Recently, graph convolution network has been widely used in the field of semantic segmentation. GraphNet [11] and WSGCN [10] are applications of graph convolution in weakly-supervised semantic segmentation. GraphNet uses GCN to propagate the initial pixel-level labels converted from bounding box labels into integral pseudo labels. WSGCN learns a two-layer GCN for each training image through back-propagation Laplace operator and entropy regularization loss to solve the lack of regularization of pseudo tags when using CAM [24] based on image level category tags. 3D-MPA applied GCN to 3D point cloud semantic segmentation, represents the proposal as nodes in the graph, and uses GCN to mine the high-order interaction between proposals [3]. Similarly, PGCNet used patches of different classes generated by point clouds as nodes to construct dynamic graph convolution and capture semantically similar structures, even though they may be far away in the original input graph [14].

3. Method

3.1. Overview

We propose the deep network called Channel Reorganization Graph Convolution Network (CRGCN). CRGCN is divided into four sections. These are: feature extraction backbone, Relation Reference Module (RRM), Channel Reorganization Module (CRM) and Multi-scale Interact Module (MIM). CRM is novel module proposed in this paper. CRM takes each channel as a node and applies a method based on distance to build a graph. Then, the structures with adverse impact are inhibited by screening. Finally, we use GCN to reorganize the relationship between channels to effectively aggregate the features in different channels. RRM calculates the relationship between the support image and the query image to provide a reference for subsequent segmentation. Based on multi-scale, MIM overcomes the problem of spatial inconsistency by adaptively enriching query features with support feature. In the following subsection, we will describe these modules in detail.

3.2. Channel Reorganization Graph Convolution Network

3.2.1. Relation Reference Module

The research of CANet shows that the middle-layer features prompt the object parts shared by unseen classes, such as color, position, edge and so on [20]. High-layer features are related to object classes. However, it will have an adverse impact on the final segmentation effect. But the semantic information provided by high-layer features is very meaningful. PFENet founded a method to solve this problem [15]. It transformed high-layer features into a prior guide which shows the probability that pixels belong to the target classes. Following this, we introduce the relation reference module, which uses high-layer features to establish the relationship between support features and query features.

In this section, X_q and X_s represent the high-level features of query images and supporting images extracted from the backbone. Specifically, to generate relation reference R_q, we first calculate the relation similarity map for support/query features through cosine similarity $cos(v_q, v_s)$. After, we determine the closest support pixel for each v_q by taking the maximum similarity among all support pixels as relation value $r_q \in R$ and R'_q. Then we reshape $R'_q \in R^{hw \times 1}$ into $R'_q \in R^{h \times w}$. Last min-max normalization is used to normalize all relation value R'_q to relation reference R_q.

3.2.2. Channel Reorganization Module

SENet mentioned that convolutional is to fuse spatial information and channel-wise information into information combination in local receptive fields [4]. It is clear that channel-wise information is of great significance in feature fusion. So in this paper, we mainly study feature fusion through channel relationship. Recently studies have shown that explicitly embedding learning can improve the performance of networks. According to this, we model discrete channel features into channel feature graph. The relationship between channel features is displayed learning by using the topology of the graph. Our goal is to reorganize channel features and establish the interdependence between channels. To achieve this, we propose the Channel Reorganization Module to emphasize the beneficial structure and suppress the useless features.

Specifically, the middle-layer query features $X_q \in R^{C \times H \times W}$ extracted from the feature extraction backbone are reshaped to $R^{C \times HW}$. Then we transform pixel-wise features into channel-wise features X_g. Next, in order to model the discrete channel nodes into topological graph structure, we use the K-nearest neighbor graph to establish the neighbor relationship between nodes and obtain the adjacency matrix A. So we get the connected graph $G = (A, X_g)$.

Following is to calculate the feature information of the graph. It mainly includes attribute information and structure information of nodes. Firstly, we calculate the Node Motif Degree of M_{31}, M_{32}, M_{41}, M_{42} and M_{43} for each node. The attribute information of each node in the graph is obtained. And we set it into the node attribute feature tensor X_{aft} of the graph. In order to obtain the structural information, we slice the node attribute feature tensor, select the three motifs with the highest Node Motif Degree, and integrate them into the structural feature tensor X_{sft}. The structure information not only strengthens the common structures in the graph structure, but also emphasizes the beneficial channel features and suppresses the useless channel features.

According to the above process, there are great differences between attribute feature tensor and structural feature tensor, which is easy to cause the problem of uneven dimension in subsequent calculation. In addition, in order to capture the similarity between abstract features of similar object parts in the image, we consider using Graph Convolution Neural (GCN). After GCN, we get the attribute embedding tensor and structure embedding tensor. Finally, we multiply the two tensors element-wise to obtain the final reorganized channel features X_{crm}. After this step, we need to project the new features $X_{crm} \in N \times D$ back into the original coordinate space $\tilde{X}_q$. We reshape $X_q \in R^{C \times H \times W}$ to $X_q \in R^{C \times HW}$. Then multiply the reorganized channel features $X_{crm} \in R^{N \times D}$ and the reshaped query features $X_q \in R^{C \times HW}$ to obtain $\tilde{X}_q \in R^{C \times HW}$. Last, we reshape the $\tilde{X}_q$ to $\tilde{X}_q \in R^{C \times H \times W}$ as the new query features.

3.2.3. Multi-scale Interact Module

The size, position, and posture of the query target will be very different from the support object, resulting in spatial inconsistency. To alleviate this problem, we use a multi-scale feature interact module like PFENet. Horizontally, MIM interacts the reorganized query features, support features and relation reference. Vertically, MIM integrates more refined features into the original features from top to bottom. Finally, MIM collects the features of different scales as new query features.

To begin this module, we concatenate the reorganized query features, support features and relation reference to generate the merge features $X_m \in R^{C \times H \times W}$. Then, we use the interpolation function to adjust X_m to n different sizes and get $X'_m = [X^1_m, X^2_m, \dots, X^n_m]$.

Next, we interact between different scales to achieve feature fusion. Concatenating X^i_m and X^{i-1}_m. Then using a 1×1 convolutional layer for them. The purpose of this is to transfer the rough features of the previous scale to the current scale. For this, two 3×3 convolution F_α are used to complete the feature fusion of each scale. At the same time, in order to better fuse all features, we use skip connections [1] at 1×1 convolutional and F_α.

After obtaining the refined feature map of n scales $X^i_{q,m}, i \in \{1, 2, \dots, n\}$, we use the interpolations function to change them to the same size $X^i_{q,m} \in R^{C \times H \times W}$ and then concatenate them together. Finally, a new fused query feature $X_{q,f} \in R^{C \times H \times W}$ is obtained by 1×1 convolution.

In addition, for better supervised learning, we output the refined features of each scale and directly predict them. This process is realized by a 3×3 convolution and a 1×1 convolution.

Finally, we use a classification header to segment the fused query features to obtain the predicted query image mask M'_q.

3.3. Training Loss

We use the cross entropy loss function as the loss function of CRGCN. In the training process, the loss mainly includes three aspects. The first and main is the loss L_{main} between the predicted query image mask and the ground-truth query image mask. The second comes from the multi-scale fusion module. We mentioned that for the i-th scale, its features will also be predicted, and its predicted mask and the ground-truth mask will produce a loss L_i, $i \in \{1, 2, \dots, n\}$. Average them and get the multiscale loss $L_{ms} = \frac{1}{n} \sum_{i=1}^{n} L_i$. So the total loss is the sum of the two loss as:

$$L = L_{main} + L_{ms} \tag{1}$$

Table 1
Mean-IoU performance of 1-way 1-shot and 5-shot segmentation on $PASCAL-5^i$. The best is marked in bold.

Method	1-shot					5-shot				
	Fold-0	Fold-1	Fold-2	Fold-3	Mean	Fold-0	Fold-1	Fold-2	Fold-3	Mean
CANet	52.5	65.9	51.3	51.9	55.4	55.5	67.8	51.9	53.2	57.1
PGNet	56.0	66.9	50.6	50.4	56.0	57.7	68.7	52.9	54.6	58.5
PFENet	**61.7**	69.5	**55.4**	56.3	60.8	63.1	70.7	**55.8**	57.9	61.9
PMMs	52.0	67.5	51.5	49.8	55.2	55.0	68.2	52.9	51.1	56.8
BriNet	56.5	67.2	51.6	53.0	57.1	-	-	-	-	-
Basline	60.2	69.5	54.5	54.5	59.7	62.6	70.7	54.8	57.9	61.5
Ours	61.6	**70.1**	54.7	**57.4**	**61.0**	**64.3**	**71.2**	55.2	**60.9**	**62.9**

Table 2
Mean-IoU performance of 1-way 1-shot and 5-shot segmentation on FSS-1000. The best is marked in bold.

Method	Backbone	1-shot	5-shot
OSLSM	VGG16	70.3	73.0
GANet	VGG16	71.9	74.3
FSS	VGG16	73.5	80.1
DoG-LSTM	VGG16	**80.8**	**83.4**
Baseline	ResNet50	73.7	74.5
Ours	ResNet50	74.8	75.7

4. Experimental Results

4.1. Experimental Settings

4.1.1. Datasets

In order to evaluate our proposed method, we experimented on $PASCAL - 5^i$ datasets and $COCO - 20^i$ datasets, which are standard datasets commonly used in FSS. $PASCAL-5^i$ includes PASCAL VOC 2012 and the annotations of SDS datasets. It contains 20 classes, which are evenly divided into 4 folds and 5 classes for each fold. Following the previous work, we used cross-validation to evaluate CRGCN. We selected 5 classes of 1 fold as the test set, and the remaining 15 classes of 3 folds as the training set. FSS-1000 contains 1000 classes of pictures with mask marks, including 520 training classes, 240 verification classes and 240 test classes.

4.1.2. Implementation Details

In order to comprehensively demonstrate the effectiveness of our proposed method and fairly compare it with other methods, we use VGG-16, ResNet-50 and ResNet-101, which the three backbone networks commonly used in FSS, to conduct main experiments on $PASCAL - 5^i$ datasets and FSS-1000 datasets. According to the previous work, all backbone networks are initialized with the model pre trained on ImageNet, and the parameters are fixed during the CRGCN model training. We use the data enhancement strategies of random horizontal flip, random clipping and random rotation to enhance the image, and adjust the size of all input images to 473 * 473. After passing through the backbone network, the size of the feature map is 60 * 60, that is, H = W = 60. Before evaluation, the segmentation mask of the query image will be adjusted back to 473 * 473.

All our experiments are implemented on Dell PowerEdge R740 using Pytorch. The SGD optimizer is used to train each network layer, and the momentum and weight attenuation are set to 0.9 and 0.0001 respectively. We adopt the "poly" policy to decay the learning rate by multiplying $(1-\frac{current_{iter}}{max_{iter}})^{power}$ where power is equal to 0.9. On both datasets, our model is trained for 100 epochs, and the learning rate and batch size are 0.0025 and 4, respectively.

In addition, in channel reorganization module, due to the need to convert discrete channel nodes into graph structure and the large number of features of each node, it takes a lot of extra time. In order to reduce the training and testing time, during the experiment, we first train an epoch and save the attribute feature tensor of each image in the training set and test set in this training. And then we can not calculate the attribute feature tensor in large batch training, but directly load the saved tensor, and then continue the subsequent calculation to complete the final training. Experiments show that this method can not only greatly reduce the time consumption, but also does not affect the accuracy of segmentation.

4.1.3. Evaluation Metrics

Like the previous method, we use two indicators to evaluate the performance of our model, namely, the class mean intersection over union(mIoU). mIoU is to calculate the average value of intersection over union(IoU) of the foreground classes in the test set (5 foreground classes in $PASCAL - 5^i$). IoU is defined as $\frac{TP}{FP+TP+FN}$, where TP, FP and FN represent true positive, false positive and false negative counts, respectively. For the convenience of comparison, we calculate the mIoU of each fold and the average mIoU of four folds.

4.2. Segmentation Performance

Table 1 reports the performance of our proposed CRGNN on $PASCAL-5^i$ dataset. We take ResNet-50 as a backbone network. In this table, we can clearly observe that our method not only outperforms the baseline, but also outperforms other methods under 1-shot setting and 5-shot setting. For 1-shot segmentation, compared with baseline and other methods, mIoU increased by 1.3% and 0.2% respectively. For 5-shot segmentation, mIoU increased by 1.4% and 1.0% respectively. Table 2 reports the performance of CRGCN on the FSS-1000 dataset,it can be seen that our methods are also better than baseline. Altogether, these results convincingly demonstrate that our method of reconstructing channel features with graph convolution network is effective in improving segmentation performance.

5. Conclusion

In this paper, we have proposed a new few-shot semantic segmentation method (CRGCN) based on channel reconfiguration graph convolution network, which is mainly composed of RRM, CRM and MIM. The main innovation of our method lies in CRM. In CRM, the graph structure is constructed according to the channel features, then the beneficial structure is filtered based on motif, and finally the channel characteristics are recombined using GCN. This can fully mine the potential relationship between query image elements. A large number of experiments have proved the effectiveness of our method.

Acknowledgement

The authors declare that there is no funding and no conflict of interest.

References

[1] Chen, L., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L., 2018. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. IEEE Transactions on Pattern Analysis and Machine Intelligence 40, 834–848.

[2] Dong, N., Xing, E.P., 2018. Few-shot semantic segmentation with prototype learning, in: Proceedings of the 29th British Machine Vision Conference.

[3] Engelmann, F., Bokeloh, M., Fathi, A., Leibe, B., Niessner, M., 2020. 3D-MPA: Multi-proposal aggregation for 3D semantic instance segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.

[4] Hu, J., Shen, L., Sun, G., 2017. Squeeze-and-excitation networks. arXiv preprint arXiv:1709.01507 .

[5] Kipf, T.N., Welling, M., 2017. Semi-supervised classification with graph convolutional networks, in: Proceedings of the 5th International Conference on Learning Representations, OpenReview.net, Toulon, France.

[6] Li, G., Jampani, V., Sevilla-Lara, L., Sun, D., Kim, J., Kim, J., 2021. Adaptive prototype learning and allocation for few-shot segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8334–8343.

[7] Liu, B., Ding, Y., Jiao, J., Ji, X., Ye, Q., 2021. Anti-aliasing semantic reconstruction for few-shot semantic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9747–9756.

[8] Liu, Y., Zhang, X., Zhang, S., He, X., 2020. Part-aware prototype network for few-shot semantic segmentation, in: Proceedings of the 16th European Conference on Computer Vision, Springer, Cham. pp. 142–158.

[9] Nguyen, K., Todorovic, S., 2019. Feature weighting and boosting for few-shot segmentation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 622–631.

[10] Pan, S.Y., Lu, C.Y., Lee, S.P., Peng, W.H., 2021. Weakly-supervised image semantic segmentation using graph convolutional networks, in: Proceedings of the IEEE International Conference on Multimedia and Expo, IEEE, Shenzhen, China,. pp. 1–6.

[11] Pu, M., Huang, Y., Guan, Q., Zou, Q., 2018. GraphNet: Learning image pseudo annotations for weakly-supervised semantic segmentation, in: Proceedings of the ACM Multimedia Conference on Multimedia Conference, ACM, Seoul, Republic of Korea. pp. 483–491.

[12] Shaban, A., Bansal, S., Liu, Z., Essa, I., Boots, B., 2017. One-shot learning for semantic segmentation, in: Proceedings of the 28th British Machine Vision Conference.

[13] Snell, J., Swersky, K., Zemel, R.S., 2017. Prototypical networks for few-shot learning, in: Proceedings of the Annual Conference on Neural Information Processing Systems, Long Beach, CA, USA. pp. 4077–4087.

[14] Sun, Y., Miao, Y., Chen, J., Pajarola, R., 2020. PGCNet: Patch graph convolutional network for point cloud segmentation of indoor scenes. The Visual Computer 36, 2407–2418.

[15] Tian, Z., Zhao, H., Shu, M., Yang, Z., Jia, J., 2022. Prior guided feature enrichment network for few-shot segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence 44, 1050–1065.

[16] Wang, H., Zhang, X., Hu, Y., Yang, Y., Cao, X., Zhen, X., 2020. Few-shot semantic segmentation with democratic attention networks, in: Proceedings of the 16th European Conference on Computer Vision, Springer, Glasgow, UK. pp. 730–746.

[17] Wang, K., Liew, J.H., Zou, Y., Zhou, D., Feng, J., 2019. PANet: Few-shot image semantic segmentation with prototype alignment, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 9196–9205.

[18] Xie, G.S., Liu, J., Xiong, H., Shao, L., 2021. Scale-aware graph neural network for few-shot semantic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5475–5484.

[19] Yang, B., Liu, C., Li, B., Jiao, J., Ye, Q., 2020. Prototype mixture models for few-shot semantic segmentation, in: Proceedings of the 16th European Conference on Computer Vision, Springer, Cham. pp. 763–778.

[20] Zhang, C., Lin, G., Liu, F., Yao, R., Shen, C., 2019. CANet: Class-agnostic segmentation networks with iterative refinement and attentive few-shot learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.

[21] Zhang, D., Luo, R., Chen, X., Chen, L., 2021a. Pyramid co-attention compare network for few-shot segmentation. IEEE Access 9, 137249–137259.

[22] Zhang, G., Kang, G., Yang, Y., Wei, Y., 2021b. Few-shot segmentation via cycle-consistent transformer, in: Proceedings of the Annual Conference on Neural Information Processing Systems, Virtual Event. pp. 21984–21996.

[23] Zhang, X., Wei, Y., Yang, Y., Huang, T.S., 2020. SG-One: Similarity guidance network for one-shot semantic segmentation. IEEE Transactions on Cybernetics 50, 3855–3865.

[24] Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A., 2016. Learning deep features for discriminative localization, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2921–2929.