DEEP MARKOV CLUSTERING FOR PANOPTIC SEGMENTATION

[1]*Minxiang Ye,* [1]*Yifei Zhang,* [1]*Shiqiang Zhu,* [1]*Anhuan Xie,* [12]*Dan Zhang*

[1] Research Center for Intelligent Robot, Zhejiang Lab, Hangzhou 311121, China
[2] Department of Mechanical Engineering, York University, Toronto M3J 1P3, Canada

ABSTRACT

Panoptic segmentation is a challenging scene understanding task that unifies semantic segmentation and instance segmentation. Namely, each pixel of an image is assigned a semantic label and an instance id. Existing works have elaborated end-to-end panoptic segmentation networks and made great progress in non-proposal-based methods. In this work, we adopt a box-free strategy and incorporate a graph-based clustering method to merge repetitive kernel weights for object instances. An alternative graph-based clustering algorithm like Markov clustering performs effective random walks for unsupervised clustering without pre-defined cluster numbers. Our proposed deep Markov clustering scheme provides an efficient alternative to guarantee instance-aware label prediction in both training and inference stages. On the COCO dataset, our method achieves promising accuracy (PQ=42.1), which is comparable with state-of-the-art methods.

Index Terms— Panoptic Segmentation, Graph Clustering, Scene Understanding

1. INTRODUCTION

Visual scene understanding has been the focus of research in computer vision, which is the fundamental premise of artificial intelligence. Perceptual systems are required to holistically recognize scene information from visual feature representation. The main components of the scene can be further categorized into stuff category and object instance. The pixel-wise recognition of stuff category is addressed as semantic segmentation [1]–[3], in which multiple objects of the same class are assigned to one label, whereas pixel-level object recognition, also known as object segmentation [4]–[7], gives a unique label to every instance of objects in the image. These two segmentation tasks form the basis of scene parsing. In recent years, panoptic segmentation [8] has been proposed to unify semantic segmentation and instance segmentation. Namely, panoptic segmentation handles both countable entity (e.g. person and car) as well as uncountable amorphous region of identical texture or material (e.g. sky and road). However, the complexity of joint task challenges the accuracy and robustness of segmentation prediction. Some works employed a separated strategy to tackle this problem, which decompose panoptic segmentation into instance prediction and semantic segmentation for stuff, such as [8]–[11]. Recent methods [12], [13] attempted to elaborate end-to-end unified framework that simultaneously extract the feature representation of stuff category and object instance.

Most previous works address panoptic segmentation by using a NMS-like post-processing procedure for non-overlapping instance segmentation. Some box-based methods such as [9], [10], [14] take bounding box proposals to guide the prediction of instance labels. However, box-based approaches involve substantial computational cost. The choice of bounding-boxes need to be elaborated to meet the needs [15]. Methods such as [11], [16], [17] proposed box-free strategy for a more efficient network architecture. Furthermore, [13] proposed to replace NMS with a designed kernel fusion operation for overlaps removal in the post-processing stage. With this fusion strategy, the kernel weights with the same identity from multiple stages can be merged before final instance generation, leading to an improved performance.

This paper provides an alternative method for kernel fusion and conventional NMS algorithm without restrictions of scenarios and requirements, which merges repetitive kernel weights with graph-based Markov clustering. Previous work attempted to incorporate the clustering constraint into the learning process for instance segmentation [18] and deep graph clustering [19], [20]. Different from other clustering such as K-means, deep Markov clustering [21] does not require the number of clusters to be known in advance. Alternatively, using convolution neural network as feature extraction tools, our deep Markov clustering achieves better clustering quality by introducing supervised graph construction as to build an optimal initial transition matrix. In this work, we adopt the kernel generation strategy in [13] and employ a DMC scheme to cluster the kernel weights with the same identity. The experimental results demonstrate the superiority of the proposed method.

2. DEEP MARKOV CLUSTER LEARNING

2.1. Preliminaries

Given a set of observations $\mathbf{X} = \{x_1, \ldots, x_N\}$, where $x_i \in \mathcal{R}^n$, $i = 1, \ldots N$, our task is to assign an unique id

$\mathbf{X} = \{x_1, \ldots, x_N\}$ to the observations within the same cluster. Our proposed clustering scheme is based on the following graph signal processing tools. Specifically, one can define a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathcal{A}\}$ with: (1) a set of vertices $\mathcal{V}$ associated with the observations $\mathbf{X}$, and (2) a set of edges $\mathcal{E}$ that correspond to the levels of similarities among the vertices $\mathcal{V}$. Each entry $a_{i,j}$ of the *adjacency* matrix $\mathcal{A}$ of the graph can be binary (same as the binary edge matrix $\mathcal{E}$) or defined as an edge weights matrix by assigning weights $\mathcal{W}$ on edges $\mathcal{E}$. $\mathbf{D}$ is a degree matrix with entries $d_{i,i} = \sum_{j=1}^{N} a_{i,j}$, and $d_{i,j} = 0$ for $i \neq j$.

A typical approach to assign weights to the edges $\mathcal{E}$ is using the Gaussian kernel function. With a Gaussian kernel, the non-negative weight $w_{i,j} \in \mathcal{W}$ assigned to each edge $e_{i,j}$ in $\mathcal{E}$ for vertices i and j is given by:

$$w_{i,j} = \begin{cases} \exp\left\{-(\mathbf{f}_i - \mathbf{f}_j)^\top \mathbf{I}(\mathbf{f}_i - \mathbf{f}_j)\right\}, & \text{if } i \neq j \\ 0, \quad \text{o.w.} \end{cases} \tag{1}$$

Here, $\mathbf{f}_i$ denotes the features of signal i (*e.g.*, locations and pixel intensities), $\mathbf{I}$ denotes an identity matrix. One can replace $\mathbf{I}$ with a symmetric, positive-definite matrix via Mahalanobis distance [22] or replace the entire weighting kernel with different conventional functions, which is out of the scope of this paper.

Graph-based clustering algorithms are typically performed on a similarity graph via weighting kernels that quantify the similarity among connected nodes. In such a similarity graph, study like [21] make an assumption that more edges between nodes can be observed within the same natural cluster than different clusters.

Given this assumption, Markov clustering (MCL) efficiently finds the cluster structure in a graph via random walk manner [21]. Within a graph $\mathcal{G}$, MCL computes random walk using language of stochastic matrices that correspond to probability values in Markov chains. MCL simulates this process by two operators called expansion and inflation. Given a Markov matrix $\mathcal{T}$, the expansion operator mathematically makes neighbors reachable by taking e-th power of the matrix as $\mathbf{T}^e$.On the contrary, the inflation operator raises each element of the column to its non-negative power $\mathcal{T}^{\circ r}$ and then perform column-wise normalization which can be summarized as $\mathcal{T}_j = \mathcal{T}_j / \sum_i^N \mathcal{T}_{i,j}^r$. A typical Markov matrix $\mathcal{T}$ can be obtained by column-wise normalization on adjacency matrix $\mathcal{A}$. By repeating these two operators, the relations between communities are strengthened and weakened respectively. The convergence of MCL are reached if there's no significant values changes on $\mathcal{T}$ and the resulting matrix $\mathcal{C}$. Note that adding self-loops to graph $\mathcal{G}$ can lead to better convergences. The cluster ids $L_i, \ldots, L_N$ can be obtained by indexing the attractors and extract the nodes connected to them, where attractors are non-zero elements of $\mathcal{C}$ diagonal.

Our proposed deep Markov clustering (DMC) scheme is based on these preliminaries. In the following, we describe our DMC scheme for the panoptic segmentation task.

2.2. Proposed Scheme

Motivated by CNNs ability to extract representative features, we formulate deep Markov clustering learning as two stages: (1) *graph learning* - find deep metric function to generate the underlying graph to reflect the node-to-node relations. (2) *Markov clustering learning* - optimize the trainable hyperparameters MCL operators to strengthen and weaken the node-to-node relations for better clustering results.

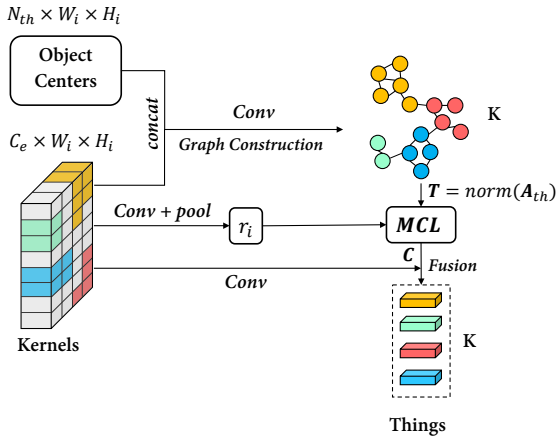


Fig. 1: The block diagram of the proposed DMC scheme. Object centers represent the position of identical object instance. Kernels denote the kernel weights for things and stuff. Following [13], the object centers are extracted by keep only top-K object centers with scores larger a preset threshold. The embeddings used in the graph generator network fuses features both from kernels and the extracted object centers

The block diagram of the proposed DMC scheme is presented in Fig. 1. Our overall scheme consists of three subnetworks: (1) graph generator network - learn a deep metric function to construct an undirected and weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$ by minimizing clustering loss 2. (2) inflation network - learn an optimal power factor r in the MCL's inflation operator to maximize the modularity 3 which is widely used as an index to evaluate the accuracy of clustering results [23]. (3) embedding network - extract the embeddings for each instance, following the work [13].

The cluster loss for graph generator network computes the difference of guided clusters and the resulting transition matrix after operating the overall DMC scheme as:

$$Loss_{\mathcal{C}} = \frac{1}{\sum_{i,j}^{M,M} b_{i,j} m_{i,j}} \sum_{i=1,j=1}^{M,M} b_{i,j} \left| c_{i,j} - m_{i,j} \right| \tag{2}$$

where $m_{i,j}$ is zero if i-th node and j-th node corresponds two different instances, otherwise is one. Note that $b_{i,j}$ is one except that 50% random i, j pairs are manually set to zero if $m_{i,j}$ is one which generating uncertainty for better position learning over the instance segmentation task.

To effectively learn the resulting power factor from inflation network, we maximum the widely used metric *modularity* [24] to improve the clustering performance by:

$$\mathbf{Q} = \frac{1}{2u}\sum_{i\neq j}(c_{i,j} - \frac{d_{i,i}d_{j,j}}{2u})\delta(L_i, L_j) \tag{3}$$

u is the total number of edges within the graph $\mathcal{G}$. $\delta(\cdot)$ is the Kronecker function which outputs one if variables $\cdot$ are equal, and zero otherwise. Therefore, higher clustering quality corresponds to higher modularity $\mathbf{Q}$, where the value of $\mathbf{Q}$ lies in the range $[-1, 1]$.

The adjacency matrix $\mathcal{A}$ of the graph $\mathcal{G}$ is constructed using the metric function Eq. 1. Given the concatenation of object centers and kernels following the work [13], $\mathbf{f}$ is output of the graph generator network. We adopt the Sigmoid activation on $\mathbf{f}$ and scale the outputs by 100.0. For the power factor of inflation operator, the output of the inflation network is bounded in range of $[1, 3]$. The above constraints are kept to ensure the mathematical stability for Markov clustering during the overall training process.

Furthermore, panoptic segmentation, as a joint segmentation task, needs to distinguish object instance and stuff category (i.e. things and stuff) in a unified workflow. We adopt the definition of *object centers* in [8], which represents the position of each object instance. Similar to [13], kernels in Fig. 1 denotes the kernel weights for things and stuff generated by stacks of convolutions. With the proposed DMC scheme, kernels with the same identity are optimally clustered to a single embedding for things, leading to an instance-aware prediction in the following stages. Notably, DMC also allows for flexible integration with other network architectures to remove repetitive instances.

3. EXPERIMENTS AND DISCUSSION

In this section, we report our experimental results for panoptic segmentation task. The experiments are conducted on widely used COCO dataset [25] in Detectron2 framework [26]. The COCO dataset contains 118K, 5K images for training, validation described by 80 thing classes and 53 stuff classes. We conduct analytical visualization on graph spectral domain to reveal the effect of our proposed DMC scheme. Besides, we compare the proposed DMC scheme against the state-of-the-art box-based approaches such as PanopticFPN [9] as well as box-free approaches including SOLOv2 [7] and PanopticFCN [13]. Discussions and comparison with previous studies on COCO dataset [25] are presented in the following.

3.1. Experimental Settings

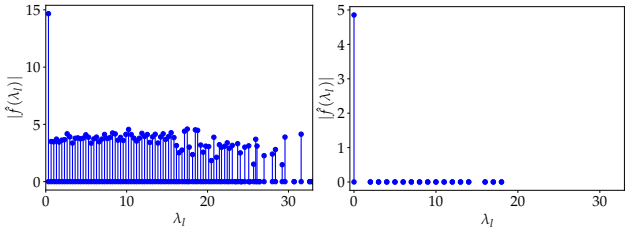
Inspired by the work of [13], we implement our method based on PanopticFCN network with PyTorch. To evaluate the performance of the proposed DMC scheme for panoptic segmentation, we adopt the evaluation metrics introduced by [8]. We employ panoptic quality (PQ) metric to measure the performance for things and stuff in a unified manner. Assuming g is the ground truth segment and p is the predicted segment, for each class, PQ can be defined as:

$$PQ = \frac{\sum_{(p,g)\in TP} IoU(p, g))}{|TP| + \frac{1}{2}|FP| + \frac{1}{2}|FN|}.$$

where TP (true positives) denotes the matched pairs of segments, FP (false positives) denotes the unmatched predicted segments, and FN (false negatives) is the unmatched ground truth segments. IoU means the intersection over union. Besides, we train the network for 90K iterations with a learning rate 0.01 using dual Tesla V100S GPU acceleration. We set the maximum iteration step of the Markov Clustering to 20. The model is trained using SGD with weight decay $1e^{-4}$ and momentum 0.9.

3.2. Ablation study

In this section, we perform an ablation study by removing inflation network components during training to reveal effects for the resulting clustering quality. We employ modularity and Graph Fourier Transform (GFT) to represent the clustering quality and the smoothness of the resulting transition matrix $\mathcal{C}$. As in [27], we visualize the magnitude of the GFT coefficients in Fig. 2 across the entire dataset using PanopticFCN and DMC w/o inflation network. Note that we use a fixed inflation value of 1.6 when inflation network is not used.



(a) without inflation $\mathbf{Q} = 0.10$ (b) with inflation $\mathbf{Q} = 0.23$

Fig. 2: The magnitude $|\hat{f}(\lambda_l)|$ of the Graph Fourier Transform Coefficients against Eigen values λ_l on Clustering.

In accordance with [27], the magnitude of GFT coefficients decay rapidly for a smooth signal (in our case, indicates better clusters that with less nodes in incorrect clusters). We can observe the magnitude of GFT coefficients is decaying rapidly along spectral frequencies using inflation network. In addition, higher modularity value indicates that our proposed inflation network improves the clustering quality in advances.

As shown in Table 1, we can observe that the proposed method outperforms the most baseline networks, while runs in 0.6x baseline inference time [13]. To make a fair comparison, our method for 1x case achieves 42.1% PQ, 47.6% PQ^{th}

Method	Backbone	box-free	PQ	SQ	RQ	PQ^{th}	SQ^{th}	RQ^{th}	PQ^{st}	RQ^{st}	RQ^{st}
PanopticFPN [9]	Res50-FPN	x	39.0	-	-	45.9	-	-	28.7	-	-
CIAE [28]	Res50-FPN	x	40.2	-	-	45.3	-	-	32.3	-	-
UPSNet [10]	Res50-FPN	x	42.5	78.0	52.5	48.6	79.4	59.6	33.4	75.9	41.7
DeeperLab [16]	Xception-71	✓	33.8	-	-	-	-	-	-	-	-
Panoptic-DeepLab [11]	Res50	✓	35.1	-	-	-	-	-	-	-	-
SOLO V2 [7]	Res50-FPN	✓	42.1	-	-	49.6	-	-	30.7	-	-
PanopticFCN-1x [13]	Res50-FPN	✓	41.3	-	-	46.9	-	-	32.9	-	-
PanopticFCN-2x	Res50-FPN	✓	43.2	-	-	48.8	-	-	34.7	-	-
PanopticFCN-3x	Res50-FPN	✓	43.6	80.6	52.6	49.3	82.6	58.9	35.0	77.6	42.9
Ours-1x	Res50-FPN	✓	42.1	80.4	51.0	47.6	82.3	57.2	33.8	77.5	41.6

Table 1: Comparison with other baseline networks on the COCO val set.1x, 2x, 3x denote the 90K, 180K and 270K training iterations in Detectron2 [26], respectively.

and 41.6% PQ^{st}, which brings 0.8%, 0.7% and 0.9% improvements compared with PanopticFCN-1x. We also compare DMC method with other operations of removing repetitive predictions. As presented in Table 2, our method outperforms Matrix NMS and kernel fusion, demonstrating the leading performance and huge potential of DMC scheme. Table 3 shows the same effectiveness of the proposed method w/o class-wise.

Method	nms	kernel fusion	DMC	PQ	PQ^{th}	PQ^{st}	AP
PanopticFCN [13]	x	x	x	38.7	42.6	32.9	25.4
	✓	x	x	38.7	42.6	32.9	27.8
	x	✓	x	41.3	46.9	32.9	32.1
Ours	x	x	✓	42.1	47.6	33.8	32.2

Table 2: Comparison with different operations of removing repetitive predictions. nms and kernel fusion denote the operation in [7] and [13], respectively. DMC is the proposed method.

Method	class-aware	PQ	PQ^{th}	PQ^{st}	AP
PanopticFCN [13]	×	41.2	46.7	32.9	30.9
	✓	41.3	46.9	32.9	32.1
Ours	x	42.1	47.6	33.8	32.2
	✓	42.1	47.6	33.8	32.2

Table 3: Comparison with PanopticFCN w/o class-wise on the COCO val set. The cosine similarity threshold for PanopticFCN is set to 0.9. Class-aware means merging kernel weights with the same predicted class.

4. CONCLUSIONS

We have proposed an effective graph-based Markov clustering algorithm for panoptic segmentation task. Our method applied the DMC scheme to merge repetitive kernel weights for object instance during training and inference stage. The evaluations on the COCO dataset demonstrate its superiority of instance-awareness for things and advanced performance compared with other baseline methods. Furthermore, the proposed method can be flexibly integrated into other panoptic and instance segmentation networks.

5. ACKNOWLEDGEMENT

This work is supported by the Leading Innovation and Entrepreneurship Team of Zhejiang Province of China (Grant No.2018R01006) and Ten Thousand Talents Program of Zhejiang Province (Grant No. 2019R51010).

6. REFERENCES

References

[1] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE Computer Society, 2015.

[2] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018. DOI: 10.1109/TPAMI.2017.2699184.

[3] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, *Rethinking atrous convolution for semantic image segmentation*, 2017. arXiv: 1706.05587 [cs.CV].

[4] S. Ren, K. He, R. Girshick, and J. Sun, *Faster r-cnn: Towards real-time object detection with region proposal networks*, 2016. arXiv: 1506.01497 [cs.CV].

[5] K. He, G. Gkioxari, P. Dollár, and R. Girshick, *Mask r-cnn*, 2018. arXiv: 1703.06870 [cs.CV].

[6] X. Wang, T. Kong, C. Shen, Y. Jiang, and L. Li, "SOLO: Segmenting objects by locations," in *Proc. Eur. Conf. Computer Vision (ECCV)*, 2020.

[7] X. Wang, R. Zhang, T. Kong, L. Li, and C. Shen, "Solov2: Dynamic and fast instance segmentation," *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[8] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollar, "Panoptic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[9] A. Kirillov, R. Girshick, K. He, and P. Dollar, "Panoptic feature pyramid networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[10] Y. Xiong, R. Liao, Z. Hengshuang, H. Rui, M. Bai, E. Yumer, and R. Urtasun, "Upsnet: A unified panoptic segmentation network," 2019.

[11] B. Cheng, M. D. Collins, Y. Zhu, T. Liu, T. S. Huang, H. Adam, and L.-C. Chen, "Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation," in *CVPR*, 2020.

[12] W. Zhang, J. Pang, K. Chen, and C. C. Loy, *K-net: Towards unified image segmentation*, 2021. arXiv: 2106.14855 [cs.CV].

[13] Y. Li, H. Zhao, X. Qi, L. Wang, Z. Li, J. Sun, and J. Jia, "Fully convolutional networks for panoptic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[14] Y. Li, X. Chen, Z. Zhu, L. Xie, G. Huang, D. Du, and X. Wang, "Attention-guided unified network for panoptic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[15] U. Bonde, P. F. Alcantarilla, and S. Leutenegger, *Towards bounding-box free panoptic segmentation*, 2020. arXiv: 2002.07705 [cs.CV].

[16] T.-J. Yang, M. D. Collins, Y. Zhu, J.-J. Hwang, T. Liu, X. Zhang, V. Sze, G. Papandreou, and L.-C. Chen, "Deeperlab: Single-shot image parser," *ArXiv*, vol. abs/1902.05093, 2019.

[17] N. Gao, Y. Shan, Y. Wang, X. Zhao, Y. Yu, M. Yang, and K. Huang, "Ssap: Single-shot instance segmentation with affinity pyramid," *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 642–651, 2019.

[18] J. Cao and H. Yan, "Instance segmentation with the number of clusters incorporated in embedding learning," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 1800–1804. DOI: 10.1109/ICASSP39728.2021.9414312.

[19] R. Liu, M. Chen, Q. Wang, and X. Li, "Robust rank constrained sparse learning: A graph-based method for clustering," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020.

[20] X. Zhang, J. Mu, H. Liu, and X. Zhang, "Graphnet: Graph clustering with deep neural networks," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 3800–3804. DOI: 10.1109/ICASSP39728.2021.9413809.

[21] A. J. Enright, S. Van Dongen, and C. A. Ouzounis, "An efficient algorithm for large-scale detection of protein families," *Nucleic Acids Research*, Apr. 2002.

[22] P. C. Mahalanobis, "On the generalized distance in statistics," *Proceedings of the National Institute of Sciences of India*, pp. 49–55, 1936.

[23] H. Akama, M. Miyake, and J. Jung, "How to take advantage of the limitations with markov clustering? the foundations of branching markov clustering (bmcl)," 2008.

[24] F. D. Malliaros and M. Vazirgiannis, "Clustering and community detection in directed networks: A survey," *Physics reports*, 2013.

[25] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Computer Vision – ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds., Cham: Springer International Publishing, 2014, pp. 740–755.

[26] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, *Detectron2*, https://github.com/facebookresearch/detectron2, 2019.

[27] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE signal processing magazine*, vol. 30, no. 3, pp. 83–98, 2013.

[28] N. Gao, Y. Shan, X. Zhao, and K. Huang, "Learning category- and instance-aware pixel embedding for fast panoptic segmentation," *IEEE Transactions on Image Processing*, vol. 30, pp. 6013–6023, 2021, ISSN: 1941-0042. DOI: 10.1109/tip.2021.3090522.