# A Novel Adaptive Hypergraph Neural Network for Enhancing Medical Image Segmentation

Shurong Chai[1], Rahul K. JAIN[1], Shaocong Mo[2], Jiaqing Liu[1], Yulin Yang[1], Yinhao Li[1], Tomoko Tateyama[3], Lanfen Lin[2], and Yen-Wei Chen[1](✉)

[1] Graduate School of Information and Engineering, Ritsumeikan University, Osaka, Japan
chen@is.ritsumei.ac.jp
[2] College of Computer Science and Technology, Zhejiang University, Hangzhou, China
[3] Department of Intelligent Information Engineering, Fujita Health University, Fujita, Japan

**Abstract.** Medical image segmentation is crucial in the field of medical imaging, assisting healthcare professionals in analyzing images and improving diagnostic performance. Recent advancements in Transformer-based networks, which utilize self-attention mechanism, have proven their effectiveness in various medical problems, including medical imaging. However, existing self-attention mechanism in Transformers only captures pairwise correlations among image patches, neglecting non-pairwise correlations that are essential for performance enhancement. On the other hand, recently, graph-based networks have emerged to capture both pairwise and non-pairwise correlations effectively. Inspired by recent Hypergraph Neural Network (HGNN), we propose a novel hypergraph-based network for medical image segmentation. Our contribution lies in formulating novel and efficient HGNN methods for constructing Hyperedges. To effectively aggregate multiple patches with similar attributes at both feature and local levels, we introduce an improved adaptive technique leveraging the K-Nearest Neighbors (KNN) algorithm to enhance the hypergraph construction process. Additionally, we generalize the concept of Convolutional Neural Networks (CNNs) to hypergraphs. Our method achieves state-of-the-art results on two publicly available segmentation datasets, and visualization results further validate its effectiveness. Our code is released on Github: https://github.com/11yxk/AHGNN.

**Keywords:** Medical image segmentation · Graph neural network · Hypergraph Neural Network · K-NN · U-Net.

## 1 Introduction

Automatic analysis of medical images has emerged as a significant field to assist healthcare professionals in early diagnosis and treatment. The task of medical image segmentation is very important in medical imaging analysis domain. Over the past decade, CNNs have emerged as the leading frameworks due to their superior performance. Several CNN-based frameworks for different computer vision tasks

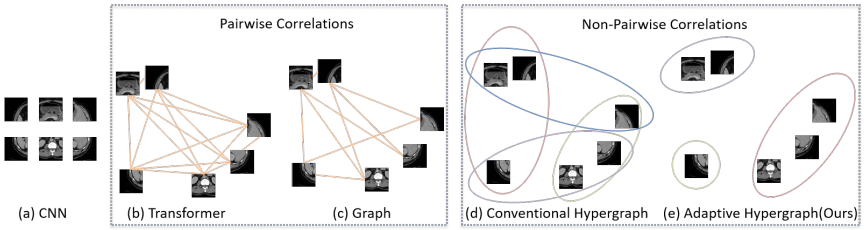


Fig. 1: The comparison between CNNs, Transformer, Graph, Hypergraph. Conventional hypergraph generated hyperedges by using fixed number of nodes while proposed scheme is constructed adaptively according to the object size.

such as segmentation, detection and classification are presented [3,9,16,22,25]. Recent advancements in Vision Transformer (ViT) [4] employing self-attention mechanism have proven effective for a variety of tasks [12]. Unlike CNNs, Transformers can efficiently capture long-range dependencies and expand receptive fields, offering significant improvements in understanding visual data. However, the design of Transformer has a weak inductive bias. Normally, Transformers necessitate a large amount of data samples for pretraining. Moreover, self-attention mechanism of Transformers tends to capture excessive redundant information, due to the use of all image patches.

More recently, Graph Convolutional Networks (GCNs) [13] have been introduced. Due to their promising performance, graph-based methods have attracted much attention. Han et al. [7] proposed the Vision Graph Neural Network (ViG), which extracts and utilizes graph-level features. The ViG model effectively captures the connections between image patches, achieving better results compared to Transformer-based models. Han et al. [8] proposed the Vision HyperGraph Neural Network (ViHGNN) and further applied the hypergraph for vision tasks. The mechanisms based on Transformers, graph, and hypergraph demonstrate a fundamental conceptual interconnectedness. A Transformer can be seen as a specific type of graph, a fully connected graph, where each node is connected to every other nodes. Similarly, a graph can be viewed as an instance of a hypergraph, in which each edge contains only two nodes, representing pairwise correlations. While hypergraph can be considered a more generalized graph structure. They utilize the concept of hyperedge, which can model not only pairwise correlations but also non-pairwise connections. This enables the hypergraph network to identify relevant correlations more effectively. This concept is resembled in Figure 1. More recently, Feng et al. introduced a Hypergraph Neural Network (HGNN) [5], which employs the K-Nearest Neighbors (KNN) algorithm to construct the hypergraph from an image. Peng et al. [19] applied HGNN at both local and global levels within a U-Net structure for MRI segmentation. Lostar et al. [17] generalized the concept of the Graph U-Net to hypergraphs to learn data embeddings more effectively, showing the potential of hypergraph structure in enhancing data representation and segmentation tasks. The construction of hyperedges is a critical aspect and currently hot research topic in this field.

Motivated by the promising results of hypergraphs, we aim to introduce the concept of hypergraphs for medical imaging problems. Our motivations can be expressed as follows: (i) The hypergraph represents a more generalized form than graph and Transformers, effectively modeling non-pairwise correlations. (ii) Semantic segmentation is the process of partitioning an image into distinct regions. This can be viewed as the process of grouping pixels that have similar characteristics. Similarly, hypergraph construction involves creating hyperedges, which also signifies a clustering mechanism for pixels with similar features. (iii) Effectively modeling both pairwise and non-pairwise connectivity necessitates the careful construction of hyperedges. Moreover, in medical imaging, one of the challenges lies in effectively segmenting objects that vary in size. Therefore, we propose a novel methodology that utilizes the degree of nodes to construct the hypergraph according to the size of objects. Conceptually, we introduce a novel approach based on node degree using KNN, offering an improved scheme to construct hypergraphs. Further, we propose to construct hypergraph at local level similar to the sliding window receptive field used in CNNs, to capture local features. These hypergraphs at the feature and local levels are concatenated to enhance the connections between nodes with similar attributes. Our approach aims to enhance medical imaging analysis by enabling a more comprehensive grouping of pixel information, leading to improved segmentation and diagnostic accuracy. Our contributions can be given as:
(1) We present a segmentation framework using the concept of hypergraph for medical image segmentation.
(2) We introduce a novel strategy for constructing hyperedges utilizing the degree of nodes, which represents their importance. The connectivity among the nodes is adaptively determined based on their degrees.
(3) Additionally, we generalize the concept of CNNs to hypergraph domain, enabling the construction of hyperedges to learn local features in a manner similar to the sliding window receptive field.
(4) Our experiment results show that our method achieves state-of-the-art performance using two publicly available medical segmentation datasets. Our visualization results further validate the effectiveness of our method.

## 2 Proposed Method

The overview of the proposed framework has been given in the Figure 2. We adopt an encoder and decoder network following the design of ScaleFormer [10]. However, ScaleFormer employs Transformers in their network, but we incorporate hypergraph modules to identify essential features more effectively. The following section provides a detailed explanation of the proposed methods.

### 2.1 Hypergraph Construction

**Hypergraph Definition.** In a graph, each edge usually connects two nodes. In contrast, a hypergraph consists of hyperedges, which can contain any number

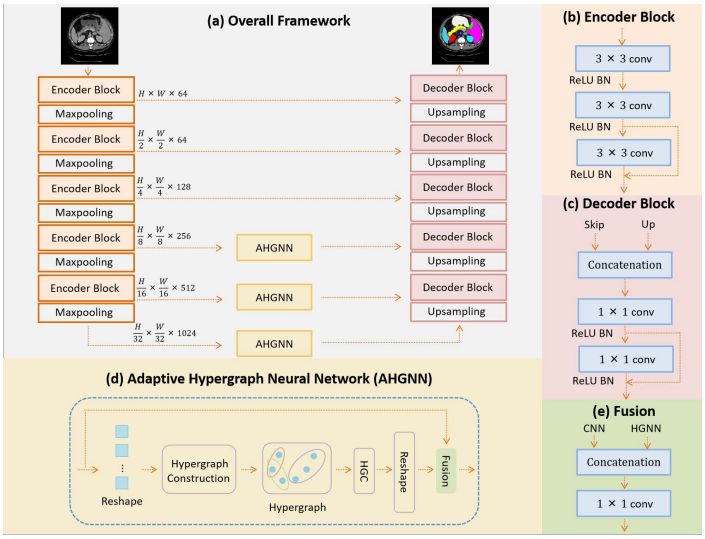


Fig. 2: The overview of proposed method.

of nodes. In this paper, a hypergraph is defined as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of $N$ nodes. We view the image patches as nodes. $\mathcal{E}$ is the set of $E$ hyperedges. The hyperedge can be derived from an incidence matrix $H \in \{0,1\}^{N \times E}$ where $H_{ne} = 1$ if the hyperedge $e \in E$ contains node $n \in N$ and $H_{ne} = 0$ otherwise.

**Adaptive Hypergraph Construction.** Recently, a few methods using hypergraph have been introduced [5,17,19]. These methods normally form hyperedges by assigning a fixed number of neighbors to each node. However, in real-world applications, such as medical imaging, objects vary in size. If hyperedges include the same number of nodes, they cannot accurately model the shape attributes of target regions. Here, we propose utilizing the degree of a node as a means to generate hyperedges. We argue that the number of connections to any node should depend on the specific context. For example, in segmentation tasks, nodes within larger segmented areas should contain more neighbors, while nodes in smaller regions should have fewer connections. Our concept of generating hyperedges based on node's degree is visually represented in Figure 3.

Formally, given an extracted feature map $I \in \mathbb{R}^{H \times W \times C}$ where $H$,$W$,$C$ are the height, width and the number of channels. The feature map $I$ is reshaped to $G \in \mathbb{R}^{HW \times C}$, which can be seen as a graph with $N = HW$ nodes with $C$ features at each node. We construct hyperedge by considering one pixel as one node in feature space. A node is considered as a patch. The hypergraph is constructed by following steps:
(1) We compute the distance matrix $B \in \mathbb{R}^{N \times N}$ between any two nodes using Euclidean distance. This distance is determined using the values across the channel dimension (Figure 3, Step-1).

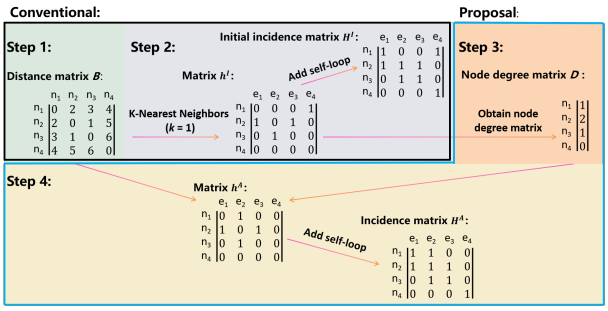


Fig. 3: The example of proposed Adaptive Hypergraph Construction. We omit $\alpha$ for simplicity.

(2) To formulate the hypergraph, the distance between $n$-th node and other nodes is represented in the $n$-th column of the matrix $B$ (Figure 3, Step-2). For each column, we utilize an algorithm such as K-Nearest Neighbors to generate the matrix $h^I \in \mathbb{R}^{N \times E_A}$. The $n$-th column of $h^I$ represents the most similar Top-$k$ neighbor nodes of $n$-th node. Then the self-loop is added to obtain the initial incidence matrix $H^I$, as discussed in [5]. In this scenario, the $n$-th node and its nearest neighbors, represented by the set $M$, form a hyperedge (each column of $h^I$ can be viewed as a hyperedge). Finally, a total of $E_A = N$ hyperedges are obtained. The element $h^I_{mn} = 1$ if $m \in M$ and $h^I_{mn} = 0$ otherwise.
(3) As a novel strategy, we propose constructing hyperedges based on the degree of nodes. Unlike the conventional approach where each hyperedge contains the same number of nodes, we determine the number of connections according to the node degree matrix $D = \sum_{e=1}^{E_A} h^I$ (summation of each row of $h^I$). Additionally, we introduce a scaling factor $\alpha$, to adjust the degree of connectivity for each node $D^\alpha = \alpha \times D$. This adjustment allows to generate a degree matrix $D$ for determining better connection of the hyperedges. This process is given in (Figure 3, Step-3).
(4) We can relate the degree of any node to its importance. A higher degree indicates that the node is involved in more hyperedges, highlighting its significance. We can use this observation to formulate new incidence matrix $H^A \in \mathbb{R}^{N \times E_A}$. More specifically, in the Step-3, using the matrix $h^I$ we calculate the degree of each node, resulting in a degree matrix $D^\alpha$. The degree of each node is considered a measure of its connectivity. For any node, if the degree is $p$, we choose the Top-$p$ nearest neighbors according to distance matrix $B$ to form a hyperedge. By applying this procedure to each node, we obtain the matrix $h^A$ which contains hyperedges based on the importance of each node. Finally, the self-loop is added to obtain the final $H^A$ (Figure 3, Step-4). We have provided sample examples in the supplementary material.

**Local Hypergraph Construction Using Sliding Window.** The widely used sliding window-based convolution mechanism uses a kernels of fixed size to

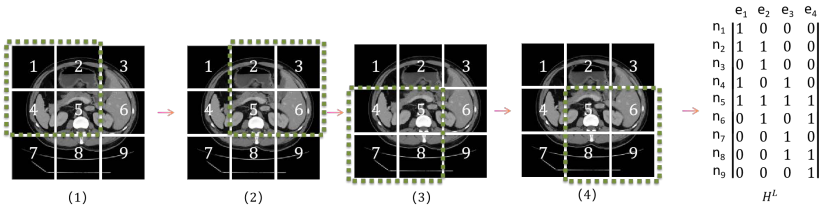


Fig. 4: Example of local hypergraph construction

convolve the feature maps while interacting with the local neighbor pixels. This process is extended across the entire image through the use of stride operations. We adopt this concept for hypergraph to extract local features. In a convolution operation with a kernel size of $KS \times KS$ and a stride of $S$, the kernel will slide $T$ times to cover the entire feature map. During each sliding window operation, we consider the nodes (patches) within the coverage of kernel as a single hyperedge. The hyperedge can be represented as $H^L \in \mathbb{R}^{N \times E_L}$. In this case, there is a total of $E_L = T$ hyperedges, with each hyperedge containing $KS \times KS$ nodes. A simple example is shown in the Figure 4. Additionally, this concept can be extended to incorporate various convolution methods, such as Dilated Convolution. This hypergraph construction enables creation of hyperedges to learn local features in a manner similar to the convolution operation.

### 2.2 Hypergraph Convolution

We use the general Hypergraph Convolution (HGC) process [5], which can be defined as:

$$Y = \sigma\left(D^{-\frac{1}{2}} H W Z^{-1} H^T D^{-\frac{1}{2}} G \Theta\right) \tag{1}$$

Where $H \in \mathbb{R}^{N \times (E_A + E_L)}$ is the concatenation of $H_A$ and $H_L$, representing the combination of these two types of hyperedges. $D = \sum_{e=1}^{E} H$ is the node degree matrix and $Z = \sum_{n=1}^{N} H$ is the hyperedge degree matrix of $H$. $D$ and $Z$ are used to normalize $H$. $W$ is a diagonal matrix representing the weights of hyperedges. We set it to an identity matrix in our experiments. The $G, Y \in \mathbb{R}^{HW \times C}$ are the input and output features. The $\Theta \in \mathbb{R}^{C \times C}$ is the training weight and $\boldsymbol{\sigma}(\cdot)$ is activation function.

### 2.3 Optimization

The combination of Cross Entropy loss and Dice loss is used to train the network:

$$L = a * L_{\text{Cross entropy}} + b * L_{\text{Dice}} \tag{2}$$

Where $a$, $b$ are hyperparameters and set to 0.3 and 0.7 based on our experiments. Additionally, we utilize the outputs from five different levels at the decoder part to adopt the loss aggregation strategy, as outlined in [21]. We calculate the loss for $2^4 - 1$ combination predictions from the last four stages of decoder network.

# 3 Experiments

## 3.1 Dataset

We have used Synapse and ACDC datasets. Synapse is a publicly available multi-organ segmentation dataset of the MICCAI 2015 Multi-Atlas Abdomen Labeling Challenge. It totally contains 30 abdominal CT scans. While 18 cases are used for training and 12 cases for testing following the previous work [2]. We report the results in terms of Dice Similarity Coefficient (DSC: %) as main measurement and 95% Hausdorff Distance (HD95: mm). ACDC is a public cardiac MRI dataset. It consists of 100 exams with three labels, left ventricle (LV), right ventricle (RV) and myocardium (MYO). The dataset is divided into 70 training samples, 10 validation samples, and 20 testing samples according to [2].

## 3.2 Implementation details

The input image resolution is set to 224×224. The entire framework is trained using SGD optimizer. For Synapse dataset, the network is trained for 600 epochs with batch size of 8 and the learning rate is set to 0.003. For ACDC dataset, the network is trained for 200 epochs with batch size of 8 and the learning rate is set to 0.05. We set the $k$=1 in for KNN to generate the initial incidence matrix $h^I$. We use kernel size $KS$=3 and a stride $S$=1 for local hypergraph construction based on our experiments.

## 3.3 Ablation study

The ablation study is conducted to evaluate the effectiveness of different components of our framework. Following the previous work, we selected common configurations such as K neighbors (set to 1), kernel size (set to 3), and stride (set to 1). These settings are widely used for extracting features, while $k$ = 1 is a standard choice to initialize hyperedge construction. We provide experiments using tuning parameter ($\alpha$) to adjust effect of node's degree. Table 1 (a) shows the experiment results using different scaling factor $\alpha$ for computing the ratio of node's degree. Based on our experiments, we observe that using a low value

Table 1: (a) Ablation study for factor of node's degree, (b) hypergraph construction methods, (c) different correlational modeling schemes

| $\alpha$ | DSC(%) |
|---|---|
| 0.5 | 83.43 |
| 1.0 | **84.03** |
| 1.5 | 83.84 |
| 2.0 | 82.90 |

(a)

| Method | DSC(%) |
|---|---|
| KNN | 82.81 |
| Adaptive | 83.27 |
| Local | 83.65 |
| Ours | **84.03** |

(b)

| Method | DSC(%) |
|---|---|
| Transformer[4] | 82.63 |
| GCN[7] | 83.27 |
| Hypergraph(Ours) | **84.03** |

(c)

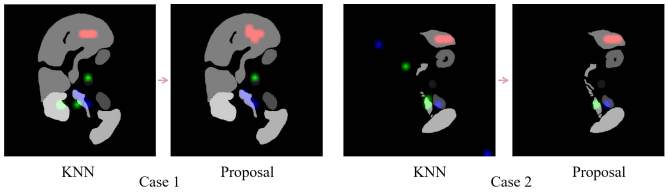


Fig. 5: The visualization results of some of our constructed hyperedges. The red, green and blue area denote the three hyperedges. Compared with KNN, which assigns a fixed number of neighbors to each node, our method adaptively assigns neighbors according to the size of segmented area.

Table 2: Synapse dataset.

| Method | DSC | HD95 | Para. |
|---|---|---|---|
| V-Net[18] | 68.81 | - | 43.60 |
| DARR[6] | 69.77 | - | - |
| R50 U-Net[2] | 74.68 | 36.87 | - |
| U-Net[22] | 76.85 | 39.70 | 37.67 |
| R50 Att-UNet[2] | 75.57 | 36.97 | - |
| Att-UNet[2] | 77.77 | 36.02 | 34.80 |
| R50 ViT[2] | 71.29 | 32.87 | - |
| TransUnet[2] | 77.48 | 31.69 | 105.13 |
| MTM[23] | 78.59 | 26.56 | 78.87 |
| SwinUNet[1] | 79.12 | 21.55 | 27.17 |
| AFTer-UNet[24] | 81.02 | - | - |
| MISSFormer[11] | 81.96 | 18.20 | 42.46 |
| TransCASCADE[20] | 82.68 | 17.34 | 35.27 |
| ScaleFormer[10] | 82.86 | 16.81 | 113.81 |
| MaxFormer[14] | <u>83.66</u> | <u>15.89</u> | 88.93 |
| Ours | **84.03** | **13.26** | 43.08 |

Table 3: ACDC dataset.

| Method | DSC |
|---|---|
| R50 U-Net[2] | 87.55 |
| R50 Att-UNet[2] | 86.75 |
| TransUnet[2] | 89.71 |
| SwinUNet[1] | 90.00 |
| TransUnet+[15] | 90.42 |
| MTM [23] | 90.43 |
| PVT-CASCADE[20] | 91.46 |
| TransCASCADE[20] | 91.63 |
| MAXFormer[14] | **92.15** |
| Ours | <u>92.02</u> |

of alpha is necessary to avoid redundant information. We also conduct experiments using different hypergraph construction methods (Table 1(b)). These methods are baseline (K-Nearest Neighbors), Only degree assignment approach (Adaptive), Hypergraph Construction Using Sliding Window (Local) and our proposed approach. We also compare different correlational modeling methods such as Transformer, GCN and Hypergraph in Table 1(c).

### 3.4 Comparison with the State-of-the-art Methods

Table 2 and 3 report the experiment results for the Synapse and ACDC datasets, respectively. According to the results, our method demonstrates better and competitive performance in comparison to other SOTA methods. Our hypergraph construction process does not increase parameter count and maintains low computational cost by using low-resolution feature space. To verify the effectiveness of our proposed method, we visualize the hyperedges formed by adaptive hypergraph construction strategy, shown in Figure 5. Our hypergraph construction method effectively models complex relationships and connectivity among pixels with similar attributes. However, in our experiments, we observed some unnec-

essary connections between nodes. Therefore, we aim to improve the efficiency of hypergraph construction further.

## 4 Conclusion

In this work, we present a hypergraph-based framework for medical image segmentation task. We proposed novel hypergraph construction method that can effectively capture the feature and local level information. Our method achieves state-of-the-art results on two publicly available segmentation datasets.

**Acknowledgments.** This work was supported in part by Ritsumeikan Advanced Research Academy (RARA) Program and the Grant in Aid for Scientific Research from the Japanese Ministry for Education, Science, Culture and Sports (MEXT) under the Grant Nos. 20KK0234, 21H03470, and 20K21821, and in part by Zhejiang Provincial Natural Science Foundation of China (No. LZ22F020012).

**Disclosure of Interests.** The manuscript is approved for publication by all authors. All authors declare no conflicts of interest regarding the submission and publication.

## References

1. Cao, H., Wang, Y., Chen, J., Jiang, D., Zhang, X., Tian, Q., Wang, M.: Swin-unet: Unet-like pure transformer for medical image segmentation. arXiv preprint arXiv:2105.05537 (2021)
2. Chen, J., Lu, Y., Yu, Q., Luo, X., Adeli, E., Wang, Y., Lu, L., Yuille, A.L., Zhou, Y.: Transunet: Transformers make strong encoders for medical image segmentation. arXiv preprint arXiv:2102.04306 (2021)
3. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE transactions on pattern analysis and machine intelligence **40**(4), 834–848 (2017)
4. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
5. Feng, Y., You, H., Zhang, Z., Ji, R., Gao, Y.: Hypergraph neural networks. In: Proceedings of the AAAI conference on artificial intelligence. vol. 33, pp. 3558–3565 (2019)
6. Fu, S., Lu, Y., Wang, Y., Zhou, Y., Shen, W., Fishman, E., Yuille, A.: Domain adaptive relational reasoning for 3d multi-organ segmentation. In: Medical Image Computing and Computer Assisted Intervention–MICCAI 2020: 23rd International Conference, Lima, Peru, October 4–8, 2020, Proceedings, Part I 23. pp. 656–666. Springer (2020)
7. Han, K., Wang, Y., Guo, J., Tang, Y., Wu, E.: Vision gnn: An image is worth graph of nodes. Advances in Neural Information Processing Systems **35**, 8291–8303 (2022)
8. Han, Y., Wang, P., Kundu, S., Ding, Y., Wang, Z.: Vision hgnn: An image is more than a graph of nodes. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 19878–19888 (2023)

9. Huang, H., Lin, L., Tong, R., Hu, H., Zhang, Q., Iwamoto, Y., Han, X., Chen, Y.W., Wu, J.: Unet 3+: A full-scale connected unet for medical image segmentation. In: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 1055–1059. IEEE (2020)
10. Huang, H., Xie, S., Lin, L., Iwamoto, Y., Han, X., Chen, Y.W., Tong, R.: Scaleformer: Revisiting the transformer-based backbones from a scale-wise perspective for medical image segmentation. arXiv preprint arXiv:2207.14552 (2022)
11. Huang, X., Deng, Z., Li, D., Yuan, X.: Missformer: An effective medical image segmentation transformer. arXiv preprint arXiv:2109.07162 (2021)
12. Khan, S., Naseer, M., Hayat, M., Zamir, S.W., Khan, F.S., Shah, M.: Transformers in vision: A survey. ACM computing surveys (CSUR) **54**(10s), 1–41 (2022)
13. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)
14. Liang, Z., Zhao, K., Liang, G., Li, S., Wu, Y., Zhou, Y.: Maxformer: Enhanced transformer for medical image segmentation with multi-attention and multi-scale features fusion. Knowledge-Based Systems **280**, 110987 (2023)
15. Liu, Y., Wang, H., Chen, Z., Huangliang, K., Zhang, H.: Transunet+: Redesigning the skip connection to enhance features in medical image segmentation. Knowledge-Based Systems **256**, 109859 (2022)
16. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3431–3440 (2015)
17. Lostar, M., Rekik, I.: Deep hypergraph u-net for brain graph embedding and classification. arXiv preprint arXiv:2008.13118 (2020)
18. Milletari, F., Navab, N., Ahmadi, S.A.: V-net: Fully convolutional neural networks for volumetric medical image segmentation. In: 2016 fourth international conference on 3D vision (3DV). pp. 565–571. Ieee (2016)
19. Peng, J., Yang, J., Xia, C., Li, X., Guo, Y., Fu, Y., Chen, X., Cui, Z.: Make u-net greater: An easy-to-embed approach to improve segmentation performance using hypergraph. Comput. Syst. Sci. Eng. **42**(1), 319–333 (2022)
20. Rahman, M.M., Marculescu, R.: Medical image segmentation via cascaded attention decoding. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 6222–6231 (2023)
21. Rahman, M.M., Marculescu, R.: Multi-scale hierarchical vision transformer with cascaded attention decoding for medical image segmentation. In: Medical Imaging with Deep Learning. pp. 1526–1544. PMLR (2024)
22. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18. pp. 234–241. Springer (2015)
23. Wang, H., Xie, S., Lin, L., Iwamoto, Y., Han, X.H., Chen, Y.W., Tong, R.: Mixed transformer u-net for medical image segmentation. In: ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 2390–2394. IEEE (2022)
24. Yan, X., Tang, H., Sun, S., Ma, H., Kong, D., Xie, X.: After-unet: Axial fusion transformer unet for medical image segmentation. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. pp. 3971–3981 (2022)
25. Zhou, Z., Siddiquee, M.M.R., Tajbakhsh, N., Liang, J.: Unet++: Redesigning skip connections to exploit multiscale features in image segmentation. IEEE transactions on medical imaging **39**(6), 1856–1867 (2019)