Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/mbe

MBE, 19(12): 12104–12126.
DOI: 10.3934/mbe.2022563
Received: 14 June 2022
Revised: 27 July 2022
Accepted: 12 August 2022
Published: 19 August 2022

Research article

Transformer with progressive sampling for medical cellular image segmentation

Shen Jiang[1], **Jinjiang Li**[1,*] **and Zhen Hua**[2]

[1] School of Computer Science and Technology, Shandong Technology and Business University, Yantai 264005, China

[2] School of Information and Electronic Engineering, Shandong Technology and Business University, Yantai 264005, China

* **Correspondence:** Email: lijinjiang@gmail.com.

Abstract: The convolutional neural network, as the backbone network for medical image segmentation, has shown good performance in the past years. However, its drawbacks cannot be ignored, namely, convolutional neural networks focus on local regions and are difficult to model global contextual information. For this reason, transformer, which is used for text processing, was introduced into the field of medical segmentation, and thanks to its expertise in modelling global relationships, the accuracy of medical segmentation was further improved. However, the transformer-based network structure requires a certain training set size to achieve satisfactory segmentation results, and most medical segmentation datasets are small in size. Therefore, in this paper we introduce a gated position-sensitive axial attention mechanism in the self-attention module, so that the transformer-based network structure can also be adapted to the case of small datasets. The common operation of the visual transformer introduced to visual processing when dealing with segmentation tasks is to divide the input image into equal patches of the same size and then perform visual processing on each patch, but this simple division may lead to the destruction of the structure of the original image, and there may be large unimportant regions in the divided grid, causing attention to stay on the uninteresting regions, affecting the segmentation performance. Therefore, in this paper, we add iterative sampling to update the sampling positions, so that the attention stays on the region to be segmented, reducing the interference of irrelevant regions and further improving the segmentation performance. In addition, we introduce the strip convolution module (SCM) and pyramid pooling module (PPM) to capture the global contextual information. The proposed network is evaluated on several datasets and shows some improvement in segmentation accuracy compared to networks of recent years.

Keywords: medical segmentation; self-attentive mechanism; transformer; strip convolution module; pyramid pooling module

1. Introduction

Transformer [1, 2] was initially used in natural language processing and showed excellent performance, so researchers introduced transformer to the field of computer vision, such as image segmentation [3], image classification [4,5] and target detection [6–10] .Thanks to its ability to capture long-term dependencies and interactions, it can easily perform in these vision processing tasks as well. However, the transformer-based model has the non-negligible drawback that the model is computationally intensive, with a computational complexity of the square of the image size, which can cope with processing small-sized images, but becomes overwhelming when faced with images of large resolution. If the problem of high computational complexity of transformer is not solved, it becomes difficult to perform dense prediction at the pixel level and the accuracy of segmentation can hardly be guaranteed. To solve this problem, common solutions such as vision transformer (ViT) [5], by simply dividing an image into equal patches of the same size, then spreading each patch and finally feeding them into the encoder and decoder for classification operations. Experiments have shown that using this scheme solves the above problem to some extent, but the drawback of this scheme is also obvious, i.e., the simple division of the image into different patches by simple labelling may lead to some highly relevant regions segmented in different patches, destroying the original structure.

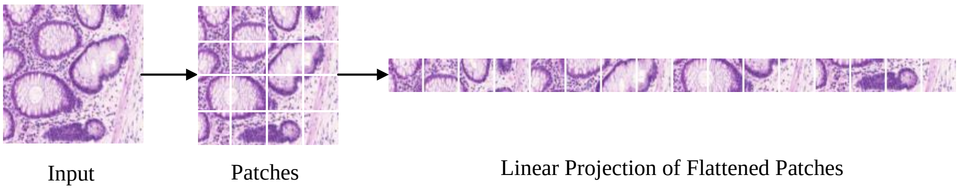

Figure 1. ViT labeled images of colorectal cancer tissue section into different patches.

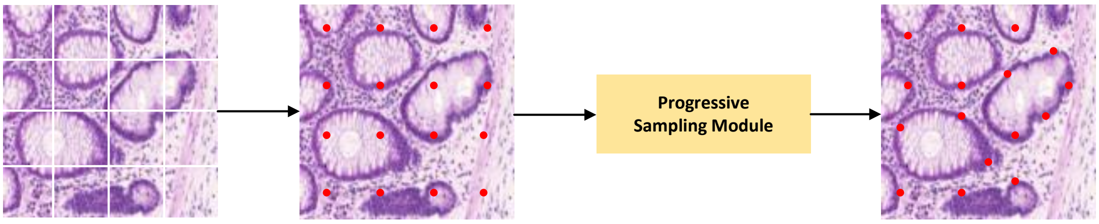

Figure 2. The input colorectal cancer tissue section image is continuously updated with its sampling position using the progressive sampling module.

As Figure 1,the input image is a cell image to be segmented, and the cell image is simply segmented into patches of uniform size by ViT, and then the flattened patches are linearly mapped. In the figure we can clearly see that a complete cell is segmented into several different patches, and there are also some non-cellular regions segmented in the patches, but these cellular regions are obviously not the regions we are interested in, and our focus should be on these cellular regions, to reduce the interference of some irrelevant regions as much as possible. In order to solve the above problem, we introduce a progressive sampling module in this paper, by which we iterate and continuously update the location of the samples. Instead of sampling a fixed block, we sample a region of interest. As Figure 2, the progressive sampling module is used to predict the offset of the next sample, which is used to update the position of the sample, gradually focusing on the important regions through iteration. Unlike the traditional ViT structure, we do not fixly split regions of high relevance in the image, causing structural damage. This approach allows us to address the computational complexity of the transformer-based model structure, while taking into account the need to focus on important regions for segmentation.

In addition, transformer-based networks have better performance under large dataset conditions, but in the face of small data sets, it will be insufficient. However, in the field of medical segmentation, the datasets are expensive to produce and require specialist knowledge to produce the labels, and the datasets are relatively rare. It is difficult to train a satisfactory model using small datasets. In this paper, a gated position-sensitive axial attention mechanism is therefore introduced again, using four gates to control the amount of information provided by the position embedding to key, query and value, all of which have learnable parameters. By borrowing this structure, our network can be trained to segment satisfactorily even with small datasets. In order to capture the contextual information more comprehensively, we introduce another SCM in the decoder to increase the perceptual field. The SCM consists of four main shapes: horizontal, vertical, left diagonal and right diagonal. With this structure, we can reduce the interference of irrelevant regions with feature learning and allow attention to fall on some important regions to be separated.

In this paper, the innovative points of our proposed network structure are as follows: (1) We use both global and local parts for feature learning, which can take into account the global contextual information as well as focus on the local detailed parts; (2) In the local structure, we use a kind of progressive sampling module to continuously update the sampled positions, so that the attention stays on the important regions to be segmented and reduces the interference of some irrelevant regions; (3) In the global structure, we introduce a SCM and a PPM in the decoder to further enhance the network's ability to perceive global contextual information.

2. Related work

2.1. Convolutional neural networks

Early medical image segmentation was dominated by traditional image segmentation algorithms, such as edge detection-based segmentation algorithms [11], threshold-based segmentation algorithms [12], region-based segmentation algorithms [13] and active contour-based segmentation algorithms [14]. However, with the development of society, these traditional medical segmentation methods can hardly meet the clinical requirements and have low segmentation efficiency. However, with the development of society, these traditional medical segmentation methods can hardly meet the clinical requirements, and the segmentation efficiency is low. In recent years, convolutional neural networks have been developed rapidly, and deep learning algorithms based on convolutional neural networks have brought breakthroughs in various fields of image processing, such as image classification and semantic segmentation, etc. Image segmentation algorithms based on deep learning have also been introduced to medical image segmentation. In the past few years, deep convolutional neural networks as the backbone network for medical segmentation, such as U-Net [15], Dense-UNet [16], Attention UNet [17], Res-UNet [18], U-Net++ [19], R2U-Net [20], 3D U-Net [21]. U-Net is a variant of FCN with a U-shaped symmetric structure, consisting of a systolic path and an expansive path, where the systolic path is used to capture contextual information and the expansive path is used to locate segmentation boundaries, and fuse shallow and deep features with the help of skip connections, but U-Net is prone to information loss through this simple skip connection, cannot extract features adequately and is only applicable to 2D images. In order to lift the limitation of U-Net on the dataset, Çiçek et al. proposed 3D U-net for processing 3D images based on U-Net. Inspired by dense connections, Dense-UNet replaced the sub-modules of U-net with dense connections to reduce the number of parameters with guaranteed accuracy, but also increased the redundancy of the network. The Res-UNet replaces the sub-module of U-Net with a residual connection, which enhances the extraction ability of the network by deepening the network depth, but increases the training time of the network to a certain extent. attention Unet, on the other hand, introduces an attention mechanism based on U-Net by adding an attention module to the encoder-decoder connection, which can enable the network to flexibly capture the connection between global and local information, but this side-by-side learning approach may corrupt the feature information of the deeper network. These neural convolutional networks excel in various segmentation tasks by virtue of their superior performance [22–26], but CNN-based networks also suffer from the non-negligible drawback of poor modelling of remotely dependent terms when performing segmentation tasks. To address this problem, Zhao et al. proposed the PSPNet network [36] to obtain multi-scale information through pyramidal pooling, and Chen et al. proposed atrous convolution [27] to model global contextual information. However, the improvement of the ability to model remote dependency terms is limited and there is still some room for improvement.

2.2. Vision transformer

Transformer first appeared in 2017 in the google paper [2]. Transformer-based networks are uniquely suited to modelling remote dependencies, with powerful global relational modelling capabilities. Transformer made a splash in the field of natural language processing, and then transformer emerged in the field of computer vision, where it can also easily handle tasks such as image segmentation, image classification and target detection. However, this powerful modelling is based on a high computational complexity, which is the square of the image size, with a certain memory requirement and a relatively large number of parameters. To solve this problem, the pyramid vision transformer [28] fuses the transformer with the feature pyramid structure, reducing the number of tokens by convolving between each stages and reducing the computational complexity of the transformer by reducing the size of the feature map. Twin [29] reduce the number of parameters by replacing absolute position encoding with relative position encoding, and reduce the complexity by local self-attention mechanism and global self-attention mechanism. Swin-transformer [30] proposes a sliding-window structure that restricts the computation of self-attention to non-overlapping local windows, which are allowed to be connected, and through this sliding-window operation, the respective attentions are computed in the local windows, greatly reducing the computational complexity of the transformer.ViT [5] divides the images into 16 patches, and then puts the flat patches into transformer for processing, this method effectively solves the problem of computational complexity of the transformer, but the network needs to be pre-trained on a large scale dataset and migrated to a small to medium scale dataset before it can perform well. However, ViT also has some limitations, it splits the image rigidly into patches of the same size, some highly correlated regions may be split in different patches, so we have made some improvements to ViT by adding a progressive sampling module, which iteratively updates the sampling position to keep some more correlated regions in the same region as much as possible, thus improving the segmentation accuracy.

2.3. *Attention transformer*

In transformer, the complexity of self-attention computation becomes large as the input image size increases. To solve this problem, Axial-Deeplab [3] decomposes two-dimensional attention into one-dimensional attention and introduces a position-sensitive axial attention module to further improve the performance of segmentation. However, Axial-Deeplab cannot be adequately trained on small datasets, so the accuracy of segmentation is often unsatisfactory. In the field of medical segmentation, the datasets are more sparsely labelled and expensive to produce, and it is difficult to train the network with fewer images to learn image features and achieve satisfactory segmentation. MedT [31] further improved Axial-Deeplab by adding a gate mechanism to it, so that the network can be trained to produce satisfactory models on even small data sets, and we use in this paper MedT's gated axial attention to accommodate small datasets.

2.4. *Extraction of features*

In performing medical segmentation tasks such as brain, liver, chest, abdomen, etc., it is a challenge to correctly segment these images to be segmented, and we strive to do so in such a way that some highly correlated regions are segmented together, and this is where long-range dependencies become particularly important, so first we have to capture features to find the intrinsic correlation of the images. But how to fully contextual information is a difficult problem, especially on some images obtained by optical imaging, and some networks have better applicability on these images, such as [32–35]. To make better use of contextual information, PSPNet [36] incorporates four different pyramidal scales for capturing contextual information by using a pyramidal pooling model, so the ability to obtain global contextual information is further enhanced. CoAnet [37] proposes a SCM to capture contextual information while reducing interference from irrelevant regions. In this paper, in order to increase the perceptual field and better capture contextual information, our network incorporates a pyramidal pooling module in the skip connection and a strip convolutional module in the decoder, which are experimentally proven to be effective.

3. Method

In this section, we describe our network structure in detail. The network structure has two components, a global structure and a local structure, with a progressive sampling module introduced in the local and a SCM and a PPM introduced in the decoder part of the global structure, along with a gated position sensitive axial attention mechanism.

3.1. Overall structure

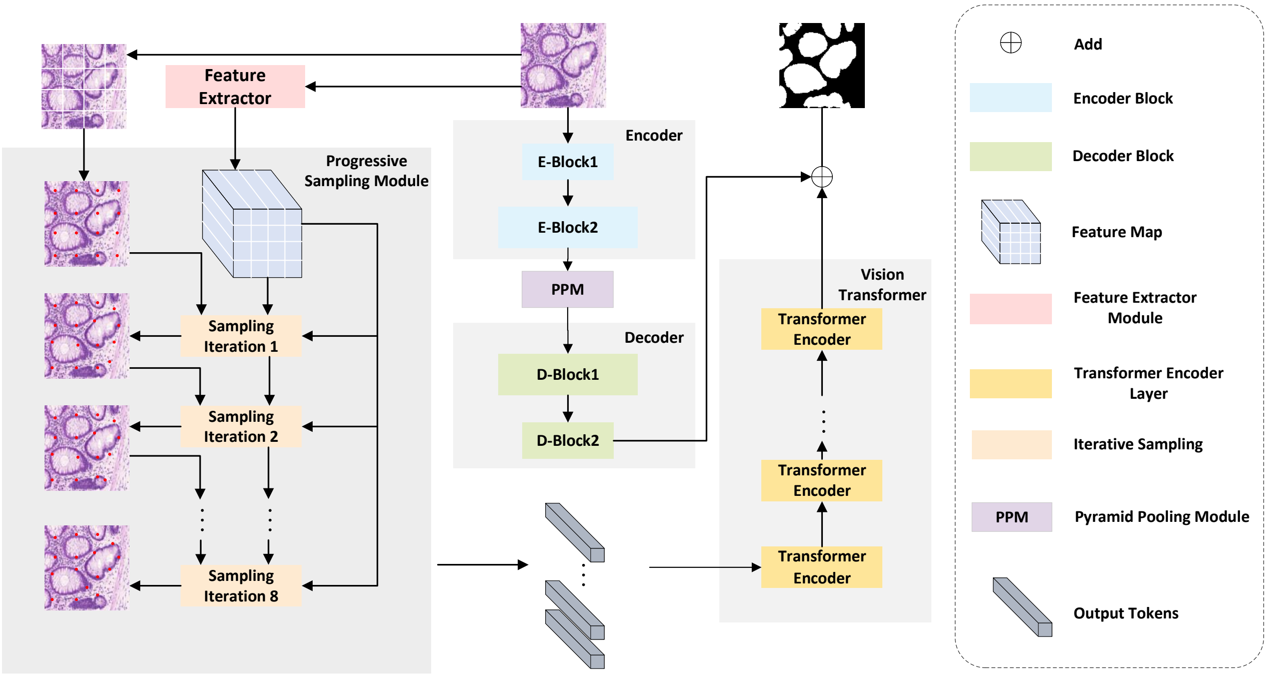

Figure 3. Diagram of the overall architecture of the network, using a global-local training strategy, with the input image being colorectal cancer tissue section.

Splitting the image into individual patches, although fast in training, cannot fully learn the dependencies between patches. In order to improve the network's ability to model between remote dependencies, we adopt a global-local training strategy, where the network is divided into two main structures, namely the global structure and the local structure, with the local structure completing the extraction of image details and the global structure learning the full-text contextual information. As Figure 3, the input image is passed through the feature extractor to obtain the corresponding feature map. In addition, the input image is first simply hard divided into 16 patches of the same size before being densely labelled, and the first labelling is done using the strategy of ViT networks to keep the dense points at the same distance. The densely labeled images and feature maps are fed into the progressive sampling module for iteration, with the updated densely labeled images and feature maps used as input for iterative sampling each time, for a total of eight iterations. The progressive sampling module outputs the updated tokens and feeds them into the visual transformer, which has the same structure as ViT and consists of multiple transformer decoders, to refine the updated tokens, and finally we adjust the output of the visual transformer. At this point, the learning of the local structure of the image is completed.

The global structure is simply divided into three components, namely the encoder, the decoder and the PPM. The input image is first processed by convolution, batch normalisation and activation functions and then fed into the corresponding two encoders, which incorporate the gated axial attention layer, which performs attention along the height and width axes respectively. The input image is then passed through the PPM to capture contextual information before being fed into the decoder, which consists of a strip convolution module that captures the features of the image in four shapes. Finally, we perform an element-by-element accumulation of the structure of the global and local structures to obtain the resultant map of the segmentation.

3.2. Progressive sampling module

Vision Transformer takes the input image and divides it into 16 patches of the same size for the corresponding operation, but this rigid division often results in some height-related regions being divided into different patches, affecting the performance of the segmentation. For this reason, we have introduced a progressive sampling module into the local structure, which iterates to continuously correct the sampling positions to ensure that these height-related regions are in the same patch, and we will describe the progressive sampling module in more detail below.

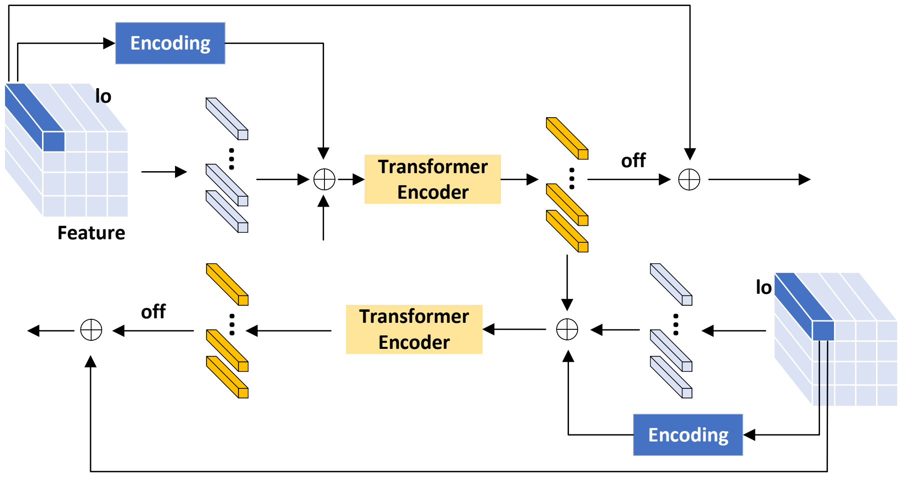

Figure 4. Detailed flow diagram of the progressive sampling module [38].

As Figure 4, in the progressive sampling module, we input the corresponding feature map $F \in R^{c \times h \times w}$, where c represents the number of channels of the feature, h and w represent the height and width of the feature, respectively. First, we sample the input feature map at the initial position lo, i.e.,

$$M' = Feature(Lo), \tag{3.1}$$

where $M' \in R^{c \times n \times n}$, $n \times n$ indicates the number of samples of the image, which is sampled by a bilinear interpolation method, $lo \in R^{2 \times n \times n}$, $Feature$ represent the sampling location and feature extraction, respectively. The sample position lo is passed through the position encoder to produce the corresponding result Lo, i.e.,

$$Lo = P \times lo, \tag{3.2}$$

where $P \in R^{c \times 2}$, $lo \in R^{2 \times n \times n}$ and $Lo \in R^{c \times n \times n}$ are projected onto the encoding matrix of $c \times n \times n$ by multiplying the channels of the sampled positions from 2 to c. Immediately afterwards, we add the output of the position encoder Lo, the result of the sampling M' at lo and the result of the last sampling M'' at lo element by element, i.e.,

$$Sum = Lo \oplus M' \oplus M'', \tag{3.3}$$

where $\oplus$ is the element-by-element summation and $Sum \in R^{c\times n\times n}$ is the cumulative result, Sum is fed into the transformer encoder to obtain the corresponding output, i.e.,

$$Out{=}T(Sum), \tag{3.4}$$

where T is the transformer encoder and $Out \in R^{c\times n\times n}$ is the output. Finally we use Out to calculate the position offset, i.e.,

$$off = Q \times Out, \tag{3.5}$$

where $Q \in R^{2\times c}$ is the offset matrix, a linear transformation of Out is the learnable parameter, and $off \in R^{2\times n\times n}$ represents the offset. The resulting offset off and the initial sample position lo are accumulated to obtain the updated sample position lo', which is expressed as

$$lo' = lo + off, \tag{3.6}$$

where $lo \in R^{2\times n\times n}$, $off \in R^{2\times n\times n}$, in the first iteration, we use the form of the ViT model, divided equally into the same intervals.

3.3. Strip convolution module

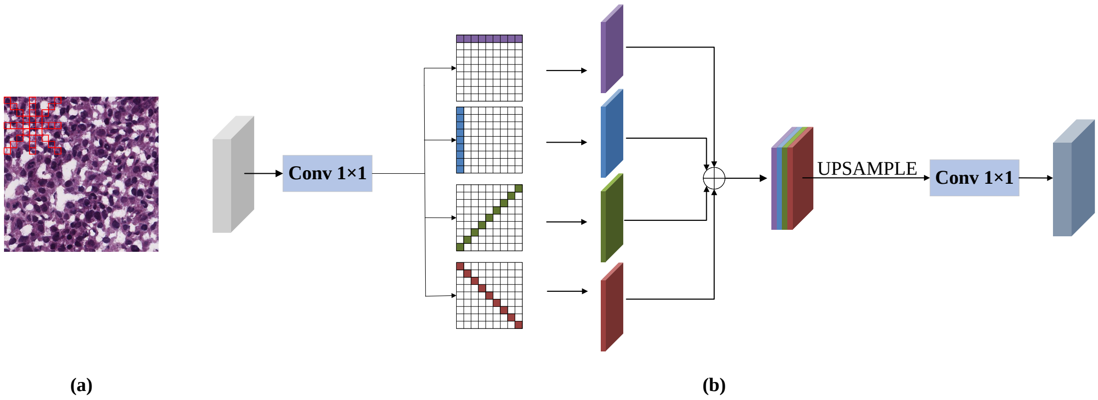

Figure 5. (a) Four shapes are used to capture the features of the image to be segmented. (b) SCM [37], consisting of four main shapes, namely horizontal, vertical, left diagonal, and right diagonal.

Strip convolution module, initially used for road segmentation, the road spans a large and continuous distribution with complex road conditions, the traditional CNN convolution is generally in a square area to learn the features, for this complex road condition, the performance is not satisfactory, in order to better learn the features at different scales, the SCM is used to capture the features of the image to be segmented, each module contains four different shapes of convolution, namely horizontal, vertical, left diagonal and right diagonal, which capture the features of the image along the four directions and can capture the dependencies between features well. The medical image we are to segment is also similar to the road image with a complex situation, and the traditional CNN convolution cannot learn the features comprehensively, so in this paper, we introduce the SCM in the global structure to adapt to the complex medical image.

As in Figure 5(b), the input tensor $x \in R^{c_1 \times h_1 \times w_1}$, c_1 denotes the dimension of the tensor, h_1 denotes the height of the tensor, w_1 denotes the width of the tensor, and the tensor x is convolved by and fed into the paths of four shapes, namely horizontal, vertical, left diagonal and right diagonal, after which the feature maps generated by the band convolution of the four shapes are concatenated and then upsampled and fed into the convolution of 1×1 to obtain the final output.

The detailed calculation process for pointwise convolution we express in the formula

$$Res[x, y] = (O * w)_E[x, y] = \sum_{i=-a}^{a} o[i + E_h a, j + E_w a] w[k + i], \tag{3.7}$$

where Res represents the corresponding output at position (x, y), w is the filter of size $2a + 1$, and the table at $E = (E_h, E_w)$ is the filter orientation, which is controlled by the four different shapes of convolution, when $E = (1, 0)$ is the filter in the vertical direction, $E = (0, 1)$ is the filter in the horizontal direction, $E = (1, 1)$ is the filter in the right diagonal direction, and $E = (-1, 1)$ is the filter in the left diagonal direction. If $a = 12$, the filter size is $2a + 1 = 25$, the same parametric number as the conventional convolution 5×5 filter, and if $E = (0, 1)$, then $E_h = 0, E_w = 1$, then the calculation is only in the horizontal direction, then it is a horizontal direction filter. In the SCM, we associate each position of the input feature map with four directions of generation, as in Figure 5(b), associating each position with all four directions further improves feature extraction and captures the remote dependencies of the image.

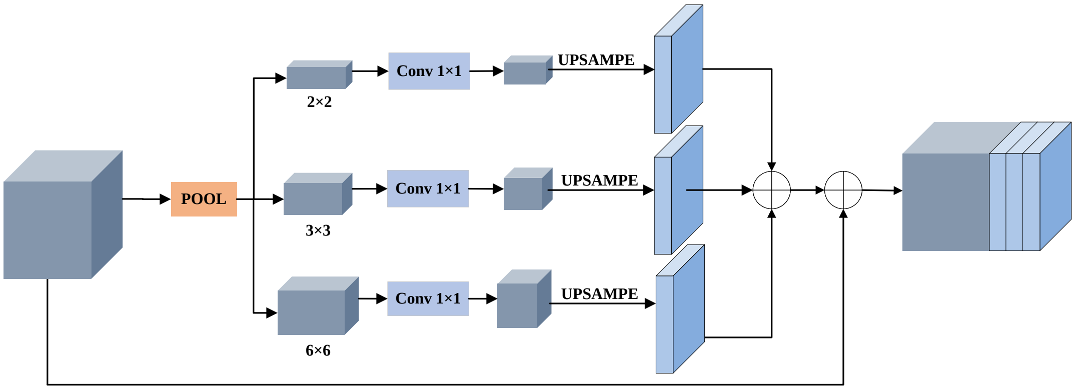

Figure 6. Pyramid pooling model [36]. Three main scales, 2×2, 3×3 and 6×6, are used to parse the features, and the final results are obtained after upsampling and stitching.

3.4. Pyramid pooling module

In a convolutional neural network, the size of the pixel on the feature map of the output of each layer of the convolutional neural network is the perceptual field, which represents our ability to extract features. The perceptual field of a convolutional neural network is lower than the theoretical value, which means that the network cannot fully extract some important information from the images. To solve this problem, the concept of global average pooling is proposed, and although it can extract global contextual information well, it also struggles with some complex scenes. Most of the medical segmentation images are more complex, if the key information of the image is not extracted, it will inevitably affect the accuracy of the segmentation. For this reason, we introduced the PPM, as in Figure 6, the input feature map is adaptively pooled to turn the feature map into $2 \times 2, 3 \times 3, 6 \times 6$. In the original model, the feature maps are of four different sizes, i.e., 1×1, 2×2, 3×3 and 6×6, and through experiments we find that the three scales work best.In order to maintain the weight of the global features, three feature maps of different sizes are sequentially convolved by 1×1 to change the channel size of the feature map, then bilinear interpolation upsampling is used to restore the size of the feature map to the same size as the original feature map, and finally they are stitched with the original feature map to obtain the final pyramid pooled features.The number of layers and size of the pyramid are variable, and here we cull a feature map with a scale of 1×1 . With this different size of feature map, the features in the image can be easily extracted.

3.5. Gate-controlled position-sensitive axial attention mechanism

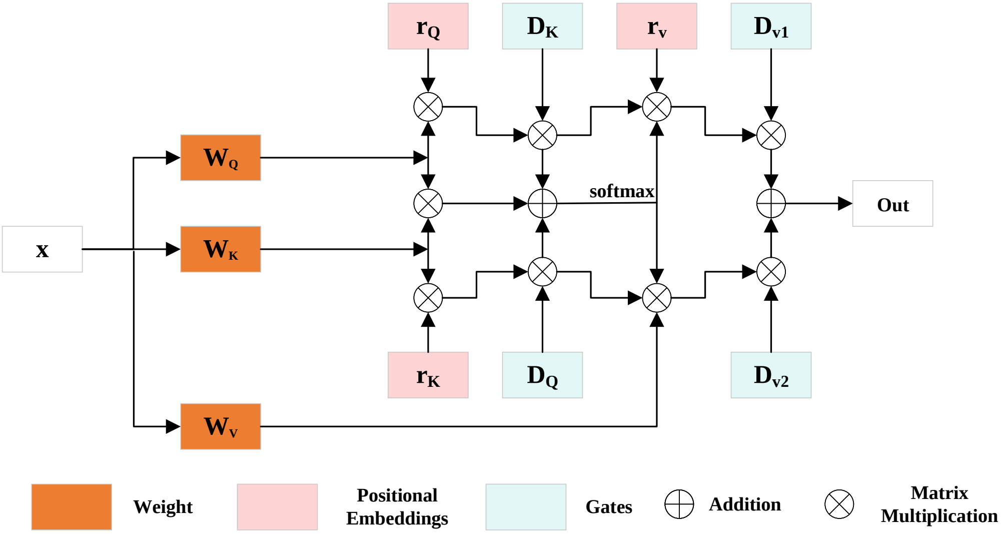

Figure 7. Use of gated axial attention layers [31] along the height and width axes, respectively.

To reduce the computational complexity, Axial-attention decomposes two-dimensional attention into two one-dimensional attention to compute, i.e., to perform attention along the height and width axes respectively, while adding the location bias code to self-attention, which can effectively capture the location information, the axial attention layer along the width axis can be expressed by the formula

$$Out_{mn} = \sum_{b=1}^{w} softmax(q_{mn}^{T}k_{mb} + q_{mn}^{T}r^{q}{}_{mb} + k^{T}{}_{mb}r_{mb}^{k})(v_{mb} + r_{mb}^{v}) \tag{3.8}$$

input a feature map $x \in R^{c_2 \times h_2 \times w_2}$, where c_2 is the channel, h_2 is the height and w_2 is the width. $Out_{mn} \in R^{c_3 \times h_2 \times w_2}$ represents the output of axial-attention, where value $v = xW_V$, query $q = xW_Q$, key $k = xW_K$, Out_{mn} represents the output at any position $m \in \{1, ..., h\}$, $n \in \{1, ..., w\}$, r_{mb}^{q} is the position code of the query, r_{mb}^{k} is the position code of the key and r_{mb}^{v} is the position code of the value. The axial attention to the layers along the height axis is similar to that of the width axis.

The added axial attention can compute the contextual information well while taking into account the efficiency. In the case of some large datasets, the key position encoding, the position encoding of the query and the position encoding of the value are well learned, but when encountering small datasets, these position encodings are difficult to learn, and if the position encoding is not accurate, the performance of the segmentation will inevitably be affected, for this reason, we introduce the gated position sensitive axial attention mechanism, and the formula can be expressed as

$$Out_{mn} = \sum_{b=1}^{W} softmax(q_{mn}^{T}k_{mb} + D_Q q_{mn}^{T}r^{q}{}_{mb} + D_K k^{T}{}_{mb}r_{mb}^{k})(D_{V1}v_{mb} + D_{V2}r_{mb}^{v}) \tag{3.9}$$

where D_Q, D_K, D_{V1}, D_{V2} are all learnable parameters and are added gating mechanisms that control the learning of relative position codes. As Figure 7, the input feature map $x \in R^{c_2 \times h_2 \times w_2}$, and the weight matrix corresponding to value, query, and key, respectively W_Q, W_K and W_V, multiply them by each other, and then process them through the position encoding and gating mechanisms respectively to get the final output $Out_{mn} \in R^{c_3 \times h_2 \times w_2}$.

3.6. Loss function

In this paper we use binary cross-entropy to train the network. In binary classification problems, binary cross-entropy is a commonly used loss function and is given by

$$Loss(\theta, \overset{\wedge}{\theta}) = -\frac{1}{HW} \sum_{x=0}^{H-1} \sum_{y=0}^{W-1} \theta(x, y) \log(\overset{\wedge}{\theta}(x, y)) + (1 - \theta(x, y)) \log(1 - \overset{\wedge}{\theta}(x, y)) \tag{3.10}$$

where H and W represent the height and width of the corresponding image respectively, $\theta(x, y)$ represents the pixels of the true value at the location of the image (x, y) and $\overset{\wedge}{\theta}(x, y)$ represents the pixels of the predicted value at the location of the image (x, y).

4. Experiment result

For training, we used the Adam optimiser to optimise the network, the batch size we set to 2, the network model was saved once every 10 iterations and the learning rate was set to 0.001. Through experiment, we found that when training the network, the size of the loss function value leveled off and stopped decreasing after 300 rounds, so we set the number of rounds to 400. We implemented the model using the PyTorch framework and trained it on NVIDIA TITAN RTX GPUs.

4.1. Datasets

We use three main datasets to train the network, namely the GLAnd segmentation (GLAS) dataset [39], the MoNuSeg dataset [40, 41] and the CVC-ClinicDB dataset [42]. The GLAnd Segmentation (GLAS) dataset and the MoNuSeg dataset are both small datasets, while the CVC-ClinicDB dataset is a large dataset.

GLAnd segmentation (GLAS) dataset. The dataset consists of 165 images from stage T3 or T42 colorectal cancer stained H&E stained tissue sections from different patients and processed on different occasions. Of these 165 images, we selected 85 images as the training set and the remaining 80 images as the validation and test set. As the size of the images in the original dataset was inconsistent, we resized the images uniformly to 128 × 128 to facilitate training.

MoNuSeg datasets. These datasets are derived from annotated H&E stained tissue images of different organs from different patients, captured by magnification of 40x, and stained and labelled by multiple hospitals. With multiple organs, different patients, and different staining schemes, this training set can be applied to train a cellular nucleus segmentation network. The original dataset was an XML file annotated with cell nucleus boundaries, and we pre-processed the annotated file through MATLAB to obtain tif format images. The dataset contains 51 images, and we used 37 images as the training set and 14 images as the test and validation set for the network. To reduce the pressure on the image memory, we also resized the images to a uniform size of 256 × 256 .

CVC-ClinicDB dataset. This dataset differs from the above two datasets in that it is a large-scale dataset. CVC-ClinicDB is a database of frames extracted for the colonoscopy video from 31 colonoscopy sequences. The training set contains a total of 1450 images, but these images vary in size and to facilitate training of the network, we set the image resolution to 128 × 128. The validation and test sets have a total of 62 images, which we also resized to the size of 128 × 128.

Table 1. Comparison of F1 score and IoU score with other methods on the MoNuSeg dataset, GLAS dataset and CVC-ClinicDB dataset, respectively.

Network	MoNuSeg		GLAS		CVC-ClinicDB	
	F1	IoU	F1	IoU	F1	IoU
Ours	**78.06**	**64.09**	**81.22**	**69.86**	**87.52**	**80.51**
MedT [31]	77.55	63.46	78.88	66.76	83.21	74.95
KiU-Net [43]	76.34	61.83	77.21	64.57	72.48	56.43
U-Net [15]	72.58	57.9	77.63	65.58	74.49	65.97
U-Net++ [19]	75.55	60.95	76.72	63.84	77.91	68.84
Attention Unet [17]	77.37	63.26	78.43	66.29	68.33	57.2
R2U-Net [20]	71.57	55.97	63.88	51.65	46.39	36.99
Res-Unet [18]	70.14	54.09	76.72	63.62	68.3	58.83
Channel-Unet [44]	74.49	59.64	78.73	66.43	76.4	66.67

4.2. Quantitative results

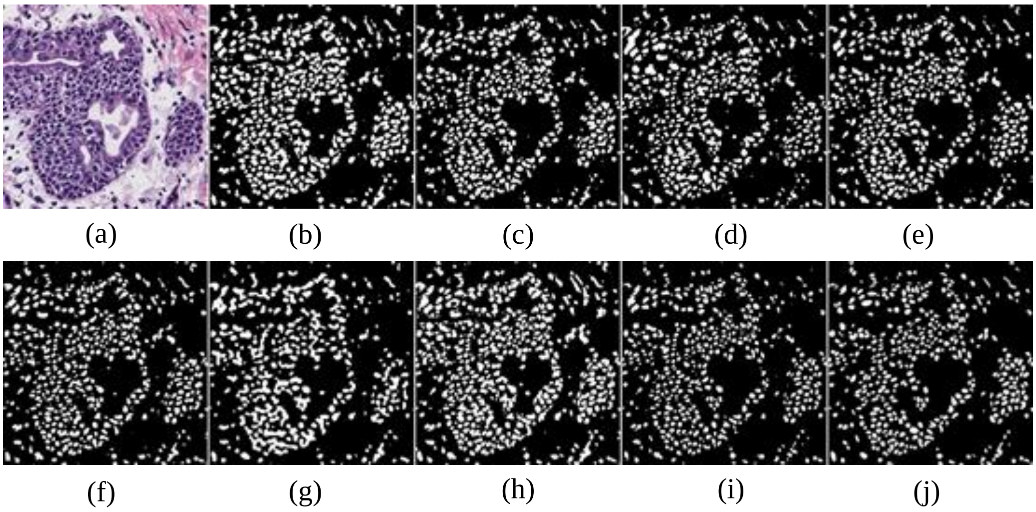

Figure 8. Prediction maps and corresponding masks for the network on the MoNuSeg dataset. (a) input, (b) U-Net, (c) MedT, (d) KiU-Net, (e) Attention Unet, (f) Channel-Unet, (g) Res-Unet, (h) U-Net++, (i) Ours, (j) mask.

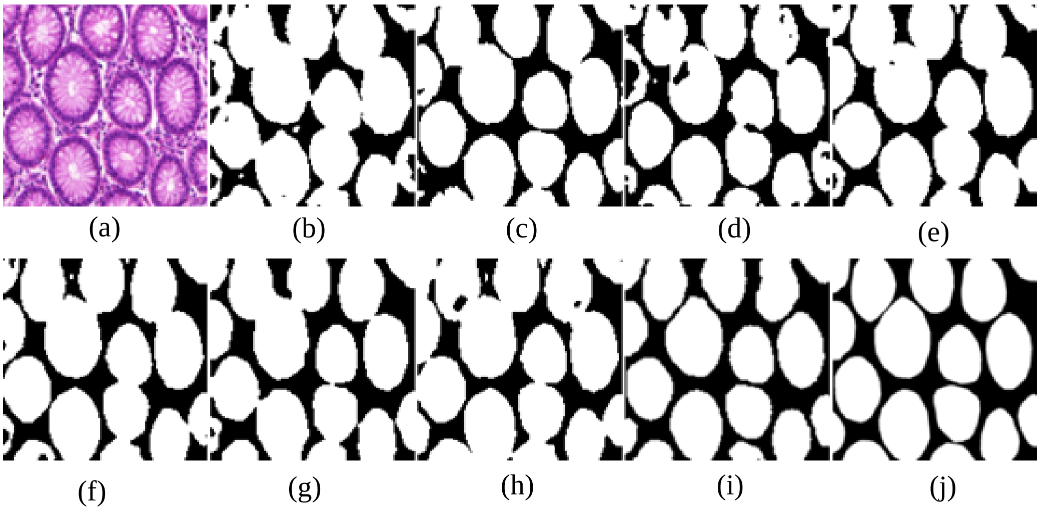

Figure 9. Prediction maps and corresponding masks for the network on the GLAS dataset. (a) input, (b) U-Net, (c) MedT, (d) KiU-Net, (e) Attention Unet, (f) Channel-Unet, (g) Res-Unet, (h) U-Net++, (i) Ours, (j) mask.

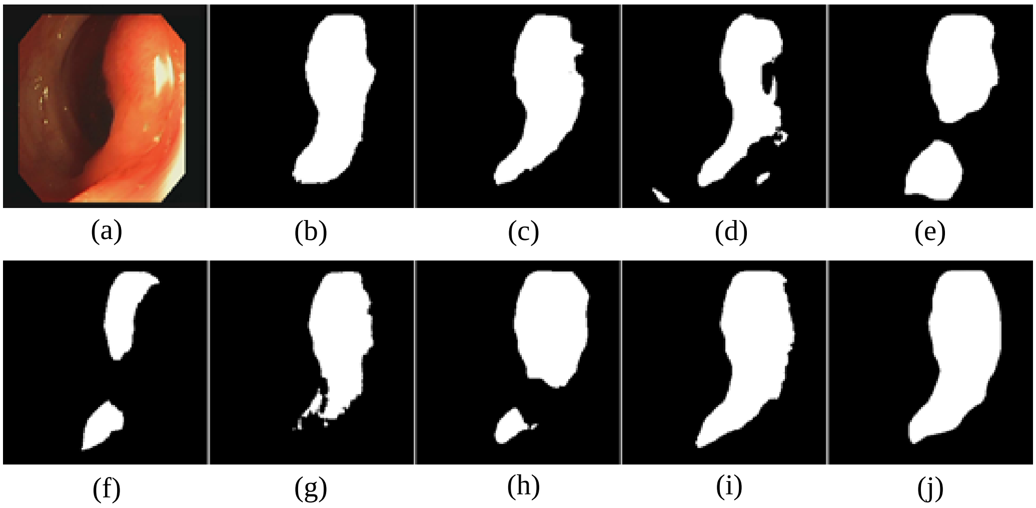

Figure 10. Prediction maps and corresponding masks for the network on the CVC-ClinicDB dataset. (a) input, (b) U-Net, (c) MedT, (d) KiU-Net, (e) Attention Unet, (f) Channel-Unet, (g) Res-Unet, (h) U-Net++, (i) Ours, (j) mask.

In this section, we first perform a quantitative analysis, comparing our method with the mainstream methods used for medical segmentation in recent years, tested with F1 score [45] and IoU score [46] on the GLAnd segmentation dataset, the MoNuSeg dataset, and the CVC-ClinicDB dataset. In the following, we provide the results of the qualitative analysis, as well as a graph of the results for the segmentation of the dataset.We use F1 score and IoU score to evaluate the effectiveness of segmentation, so the evaluation is done on MATLAB. IoU is an evaluation metric to evaluate the segmentation ability, it is to calculate the ratio of intersection and concatenation of the set of true and predicted values, F1 denotes the summed average of precision and recall, its formula can be expressed as

$$IoU = \frac{TP}{FP + FN + TP} \tag{4.1}$$

$$F1 = \frac{2TP}{FP + FN + 2TP} \tag{4.2}$$

where TP is the number of pixels for which the model predicts a positive true value, FP is the number of pixels for which the model predicts a positive true value and FN is the number of pixels for which the model predicts a negative true value.

As Table 1, we performed quantitative analysis on three datasets separately and with other methods. These methods are based on convolutional neural networks, such as U-Net network [15], U-Net++ network [19], R2U-Net network [20], Res-Unet network [18], Channel-Unet networks [44], and also there are Unet with attention, such as attention Unet networks [17], there are also transformer-based networks, such as MedT networks [31] and KiU-Net networks [43]. In small datasets, i.e., the MoNuSeg dataset and the GLAS dataset, these transformer-based networks and networks with attention have significantly better segmentation performance than neural convolution-based networks, as it is difficult to have few datasets to train a satisfactory neural convolution network, and attention has a positive effect.

However, the effect of attention is not as obvious when training networks with large data sets, e.g., the U-net network performs significantly better than the KiU-Net network on the CVC-ClinicDB data set. From the table, it is easy to see that our network has a clear advantage when comparing with other methods. For example, compared with the better performing network MedT, the F1 score and IoU score are 2.34 and 3.10 higher respectively on the GLAS dataset, which is a significant improvement in segmentation performance, and also on other datasets.

Figures 8–10, represent the different network prediction maps on the MoNuSeg dataset, GLAS dataset and CVC-ClinicDB dataset respectively. It is obvious from the figures that the prediction maps of our network better restore the detailed parts of the images compared with those of the other networks. Thanks to the inclusion of the PPM and the SCM, our network captures the details more accurately. As in Figure 8, our network is extremely similar to mask in that some small detailed parts can be captured by our network, but other networks such as Res-Unet are more noisy and KiU-Net appears to have multiple cell nuclei regions connected together. In Figure 9 which multiple cells are connected together, except for our network, there are some discrepancies compared to mask, such as the KiU-Net network, which shows obvious prediction errors and no restoration of cell interiors, and the prediction map of the Channel-Unet network, which also shows obvious noise, whereas our network, due to the inclusion of a gated position-sensitive axial attention mechanism, it can capture long-range dependencies and therefore predicts more accurately. In Figure 10 which, such as attention Unet, channel-Unet, res-Unet, and U-Net++ split a complete polyp block into two parts, U-Net and MedT do not make the above mistakes but lack the reduction of details. The progressive sampling module introduced by our network gradually focuses attention on the region of interest through iteration, reducing the irrelevant region of interference, and thus the predicted effect maps are significantly better than the other methods.

4.3. Ablation experiments

Table 2. F1 score and IoU score corresponding to ablation experiments using three datasets.

Network	MoNuSeg		GLAS		CVC-ClinicDB	
	F1	IoU	F1	IoU	F1	IoU
Ours	**78.06**	**64.09**	**81.22**	**69.86**	**87.52**	**80.51**
Removal SCM	77.53	63.47	80.29	68.04	85.27	77.45
Remove PPM	77.34	63.15	80.14	67.95	85.12	77.32
Replace progressive sampling	77.15	62.86	79.63	67.88	84.93	77.02
Replace gated axial attention	76.63	62.17	78.89	66.93	86.12	78.04

We conducted ablation experiments on the MoNuSeg dataset, the GLAS dataset and the CVC-ClinicDB dataset to demonstrate the effectiveness of the incorporation module. As shown in Figure 11, the first and second rows indicate testing on the MoNuSeg dataset, the third and fourth rows indicate testing on the GLAS dataset, and the fifth and sixth rows indicate testing on the CVC-ClinicDB dataset. First, we replace the SCM in the decoder of the global structure with a normal 2D convolution, with a convolution kernel size of 3, a step size of 1 and a padding of 1. Only the 2D convolution is used instead of the SCM, and the rest of the module remains unchanged. As Figure 11, the original network in the second and sixth columns, the normal 2D convolution is clearly inferior to the strip convolution module in capturing details, e.g., on the Glas dataset, the prediction map of the network with the SCM removed shows incorrect predictions, with regions that should not intersect overlapping, and is clearly not sufficiently reductive when predicting on the CVC-ClinicDB dataset. As Table 2, the F1 and IoU score corresponding to the original network are 0.53 and 0.62 higher, respectively, on the MoNuSeg dataset than the network with the SCM removed, and also had a significant advantage on the other datasets, indicating that the added SCM, which can more nearly improve the extraction of features to capture the remote dependencies of the images, has a positive effect on the performance of the segmentation.

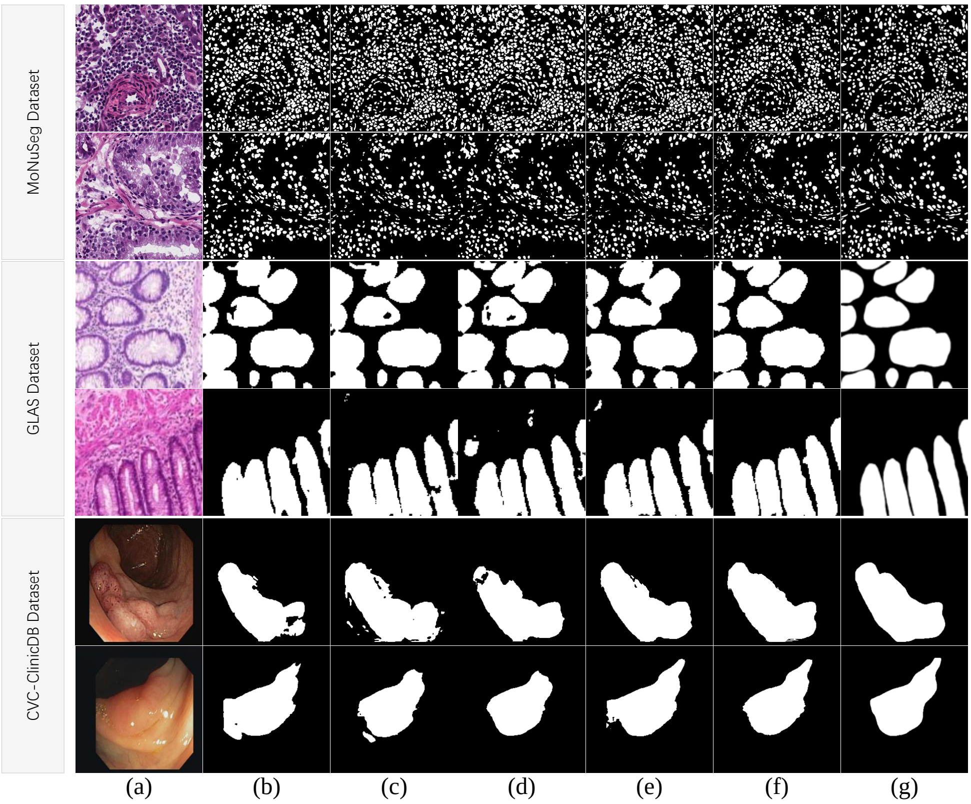

Figure 11. Qualitative results of ablation experiments performed on three datasets. (a) Input images, (b) Network prediction map with SCM replaced by 2D convolution, (c) Network prediction map with PPM removed, (d) Network prediction map with gated axial attention replaced by axial attention, (e) Network prediction map with progressive sampling replaced by hard segmentation, (f) Network prediction map without any modification, (g) mask.

In addition, to verify the effectiveness of adding the PPM, we conducted another ablation experiment on the module, which was eliminated and left otherwise unchanged, as shown in Figure 11, the third row shows the prediction map for this model, which appears more noisy compared to the original network, and is not adequately restored within some cells. Removing the PPM, the F1 score and IoU score are also significantly lower than the original network. the PPM uses three scales to extract key information from the image, and therefore has some improvement in the accuracy of the segmentation. We again replace the progressive sampling module in the local region with a hard segmentation, which splits the image into 16 patches of equal length and width. Then, after each patches is processed by five encoders and five decoders in turn, the 16 patches are then subjected to a stitching operation to restore them to the original image size, and then the feature maps of the global part and the local part are added together.As shown in the prediction map of the altered model for the fifth row in Figure 11, this hard segmentation may split regions of high relevance in the image, causing structural damage, so the segmentation is not as effective as the original network, and the F1 score and IoU score do not reach the original network. Finally, we gated axial attention by replacing it with axial attention, as shown in the fourth row of Figure 11, it is clear that on the MoNuSeg and Glas datasets, the performance is the worst, and the corresponding F1 score and IoU score are also the lowest, thanks to the fact that the addition of gated axial attention can extend the original network structure, and the network can be trained satisfactorily on small-scale datasets.

4.4. Loss function convergence

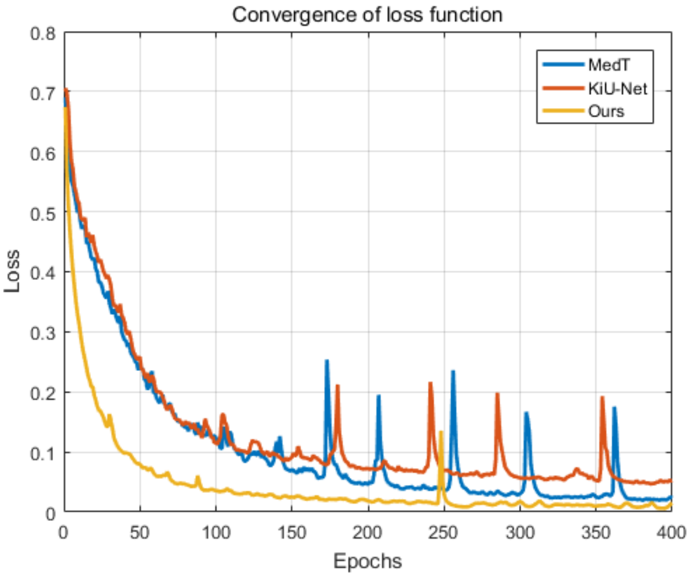

Figure 12. Comparison of our network with KiU-Net and MedT loss convergence.

The convergence of the loss function is an important feature associated with the network. Faster convergence allows for less training and is by benefit to the training network. As Figure 12, we tested our method and the MedT network and KiU-Net separately on the GLAS dataset, it is evident that our network clearly converges faster, and in the first 100 rounds, the loss decreases more and the slope of the decrease in the other networks is significantly less than our network. In addition, after 100 rounds, the MedT and KiU-Net burr more, while our network declines smoothly, and a similar pattern appears on the other datasets.

4.5. Effect of batch-size

To test the effect of batch-size on the results, we trained two networks on three datasets with batch-szie set to 1, 2 and 4, respectively. As Table 3, both on our network and the MedT network, the

Table 3. Effect of different batch-size on experimental results.

Network	batch-size	MoNuSeg		GLAS		CVC-ClinicDB	
		F1	IoU	F1	IoU	F1	IoU
	1	77.98	63.87	81.03	69.75	87.31	80.26
Ours	2	78.06	64.09	81.22	69.86	87.52	80.51
	4	78.21	64.18	81.34	69.93	87.64	80.63
	1	77.38	63.25	78.67	66.61	83.11	73.98
MedT	2	77.55	63.46	78.88	66.76	83.21	74.95
	4	77.63	63.55	78.96	66.83	83.36	75.06

Table 4. Effect of different epochs on experimental results.

Network	epochs	MoNuSeg		GLAS		CVC-ClinicDB	
		F1	IoU	F1	IoU	F1	IoU
Ours	300	77.93	63.86	81.06	69.75	87.36	80.32
Ours	400	78.06	64.09	81.22	69.86	87.52	80.51
Ours	500	78.15	64.16	81.32	69.91	87.61	80.60
KiU-Net	300	76.12	61.68	77.08	64.32	72.31	56.41
KiU-Net	400	76.34	61.83	77.21	64.57	72.48	56.43
KiU-Net	500	76.45	61.95	77.34	64.67	72.56	56.53

corresponding F1 score and IoU score gradually increased as the batch-size increased, but the increase was not significant, e.g., when the batch-size was changed from 2 to 4 in the MedT network, the corresponding F1 score and IoU score on the MoNuSeg dataset only increased by 0.08 and 0.09, which had little impact on the experimental results. To reduce the pressure of training, we set the batch-size to 2 uniformly.

4.6. The impact of epoch

In order to test the optimal number of rounds, we tested two networks, our network and KiU-Net, and the number of rounds of training is also related to the convergence of the loss function, when the loss function of the network converges fast, the number of rounds required is less, for example, our network converges faster when training, and has completed the convergence of the function at 300 rounds. After 300 rounds, it leveled off and the F1 score and IoU score did not increase much. However, KiU-Net,on the MoNuSeg dataset, had some improvement when increasing the number of rounds from 300 to 400, with its corresponding F1 score and IoU score increasing by 0.22 and 0.15, respectively. For the sake of experimental fairness, we set the number of training rounds to 400 for all networks.

4.7. Effect of image size

The image size of the MoNuSeg dataset is 1024 × 1024, and we adjusted the resolution of the image to 256 × 256. To test the effect of image resolution on segmentation performance, we trained the network on the MoNuSeg dataset with two sizes separately. As Table 5, when the image resolution is increased, there is a certain improvement on the segmentation performance, but the improvement is not

Table 5. Effect of different resolutions on experimental results.

Network	images size	F1	IoU
Ours	256 x 256	78.06	64.09
Ours	1024 x 1024	78.23	64.28
KiU-Net	256 x 256	76.34	61.83
KiU-Net	1024 x 1024	76.51	61.98

too large, for example, in KiU-Net, the F1 score and IoU score only increase by 0.17 and 0.15 when the image resolution is increased. When training the network, we again have to consider the hardware cost required, for this reason, the image resolution of the dataset we use to train the network is 256 × 256, by sacrificing the small cost performance to reduce the complexity of training.

5. Discussion

Compared with other medical image segmentation models, our model has the following advantages: (1) On the basis of ViT, we use a progressive sampling module, which can update the sampling position and continuously correct the sampling position, which solves the problem that the hard division of ViT makes the highly correlated regions of the image split into different regions. (2) We replaced the conventional CNN with a SCM, which contains four different shapes of convolution, namely horizontal, vertical, left diagonal and right diagonal, which captures the features of an image along four directions and can better capture the dependencies between features, compensating for the inability of conventional CNN convolution to learn features comprehensively on medical images. (3) In the model, we also added PPM to capture the multi-scale features of the image to further enhance the image perceptual field, so that the model can extract features of both large and small targets of the image with high robustness. (4) We use a gated position-sensitive axial attention mechanism that allows the model to exhibit superior performance even on small data sets.

The model is improved on the basis of ViT and has some drawbacks, such as the large number of parameters in the model and the computational complexity related to the square of the token, which we can improve later by certain methods, such as reducing the number of tokens by convolution between stages, etc.

6. Conclusions

In this paper, the image segmentation process can be broadly divided into two major parts, namely the global structure, which is used to link the context and model the full-text information, and the local structure, which focuses on the extraction of details. In the global structure, we add the PPM and the SCM to further improve the network's ability to extract features, and we add the progressive sampling module to the local structure to focus the network on regions of interest for segmentation. In addition, we added a gated position-sensitive axial attention mechanism to enable the network to train a satisfactory model even on small data sets. In order to verify the effectiveness of the added modules, we also perform ablation experiments on these modules, and we demonstrate experimentally that the addition of these modules is beneficial to the performance of segmentation. Finally, we again discuss the effects of batch-size, epoch and image resolution on the experiments separately, taking into account the hardware conditions and setting the parameters to appropriate values.

Acknowledgments

The authors acknowledge the National Natural Science Foundation of China (61772319, 62002200), Shandong Natural Science Foundation of China (ZR2021QF134 and ZR2021MF068), and Yantai science and technology innovation development plan (2022JCYJ031).

Conflict of interest

The authors declare that there is no conflict of interest.

References

1. J. Devlin, M. W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, preprint, arXiv: 1810.04805. https://doi.org/10.48550/arXiv.1810.04805
2. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, et al., Attention is all you need, *Adv. Neural Inf. Process. Syst.*, **30** (2017).
3. H. Wang, Y. Zhu, B. Green, H. Adam, A. Yuille, L. C. Chen, Axial-Deeplab: Stand-alone axial-attention for panoptic segmentation, in *European Conference on Computer Vision*, (2020), 108–126. https://doi.org/10.1007/978-3-030-58548-8
4. M. Chen, A. Radford, R. Child, J. Wu, H. Jun, D. Luan, et al., Generative pretraining from pixels, in *International Conference on Machine Learning*, (2020), 1691–1703.
5. A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, et al., An image is worth 16x16 words: Transformers for image recognition at scale, preprint, arXiv: 2010.11929. https://doi.org/10.48550/arXiv.2010.11929
6. X. Zhu, W. Su, L. Lu, B. Li, X. Wang, J. Dai, Deformable detr: Deformable transformers for end-to-end object detection, preprint, arXiv: 2010.04159. https://doi.org/10.48550/arXiv.2010.04159
7. M. Zheng, P. Gao, R. Zhang, K. Li, X. Wang, H. Li, et al., End-to-end object detection with adaptive clustering transformer, preprint, arXiv: 2011.09315. https://doi.org/10.48550/arXiv.2011.09315
8. Z. Dai, B. Cai, Y. Lin, J. Chen, Up-detr: Unsupervised pre-training for object detection with transformers, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (2021), 1601–1610. https://doi.org/10.1109/CVPR46437.2021.00165
9. Z. Sun, S. Cao, Y. Yang, K. M. Kitani, Rethinking transformer-based set prediction for object detection, in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, (2021), 3611–3620. https://doi.org/10.1109/ICCV48922.2021.00359
10. Z. An, X. Wang, B. Li, Z. Xiang, B. Zhang, Robust visual tracking for uavs with dynamic feature weight selection, *Appl. Intell.*, **2022** (2022), 1–14. https://doi.org/10.1007/s10489-022-03719-6

11. R. Muthukrishnan, M. Radha, Edge detection techniques for image segmentation, *Int. J. Comput. Sci. Inf. Technol.*, **3** (2011), 259. https://doi.org/10.5121/ijcsit.2011.3620

12. N. Otsu, A threshold selection method from gray-level histograms, *IEEE Trans. Syst. Man Cybern.*, **9** (1979), 62–66. https://doi.org/10.1109/TSMC.1979.4310076

13. H. G. Kaganami, Z. Beiji, Region-based segmentation versus edge detection, in *2009 Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, (2009), 1217–1221. https://doi.org/10.1109/IIH-MSP.2009.13

14. M. Kass, A. Witkin, D. Terzopoulos, Snakes: Active contour models, *Int. J. Comput. Vision*, **1** (1988), 321–331. https://doi.org/10.1007/BF00133570

15. O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, (2015), 234–241. https://doi.org/10.1007/978-3-319-24574-4

16. X. Li, H. Chen, X. Qi, Q. Dou, C. W. Fu, P. A. Heng, H-denseunet: Hybrid densely connected unet for liver and tumor segmentation from ct volumes, *IEEE Trans. Med. Imaging*, **37** (2018), 2663–2674. https://doi.org/10.1109/TMI.2018.2845918

17. O. Oktay, J. Schlemper, L. L. Folgoc, M. Lee, M. Heinrich, K. Misawa, et al., Attention u-net: Learning where to look for the pancreas, preprint, arXiv: 1804.03999. https://doi.org/10.48550/arXiv.1804.03999

18. X. Xiao, S. Lian, Z. Luo, S. Li, Weighted res-unet for high-quality retina vessel segmentation, in *2018 9th International Conference on Information Technology in Medicine and Education (ITME)*, (2018), 327–331. https://doi.org/10.1109/ITME.2018.00080

19. Z. Zhou, M. M. Rahman-Siddiquee, N. Tajbakhsh, J. Liang, Unet++: A nested u-net architecture for medical image segmentation, in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, (2018), 3–11. https://doi.org/10.1007/978-3-030-00889-5

20. M. Z. Alom, M. Hasan, C. Yakopcic, T. M. Taha, V. K. Asari, Recurrent residual convolutional neural network based on u-net (r2u-net) for medical image segmentation, preprint, arXiv: 1802.06955. https://doi.org/10.48550/arXiv.1802.06955

21. Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, O. Ronneberger, 3D u-net: Learning dense volumetric segmentation from sparse annotation, in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, (2016), 424–432. https://doi.org/10.1007/978-3-319-46723-8_49

22. C. Zhao, Y. Xu, Z. He, J. Tang, Y. Zhang, J. Han, et al., Lung segmentation and automatic detection of covid-19 using radiomic features from chest CT images, *Pattern Recognit.*, **119** (2021), 108071. 2021. https://doi.org/10.1016/j.patcog.2021.108071

23. X. Liu, A. Yu, X. Wei, Z. Pan, J. Tang, Multimodal mr image synthesis using gradient prior and adversarial learning, *IEEE J. Sel. Top. Signal Process.*, **14** (2020), 1176–1188. https://doi.org/10.1109/JSTSP.2020.3013418

24. X. Liu, Q. Yuan, Y. Gao, K. He, S. Wang, X. Tang, et al., Weakly supervised segmentation of covid19 infection with scribble annotation on CT images, *Pattern Recognit.*, **122** (2022), 108341. https://doi.org/10.1016/j.patcog.2021.108341

25. J. He, Q. Zhu, K. Zhang, P. Yu, J. Tang, An evolvable adversarial network with gradient penalty for covid-19 infection segmentation, *Appl. Soft Comput.*, **113** (2021), 107947. https://doi.org/10.1016/j.asoc.2021.107947

26. N. Mu, H. Wang, Y. Zhang, J. Jiang, J. Tang, Progressive global perception and local polishing network for lung infection segmentation of covid-19 ct images, *Pattern Recognit.*, **120** (2021), 108168. https://doi.org/10.1016/j.patcog.2021.108168

27. L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A. L. Yuille, Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, *IEEE Trans. Pattern Anal. Mach. Intell.*, **40** (2017), 834–848. https://doi.org/10.1109/TPAMI.2017.2699184

28. W. Wang, E. Xie, X. Li, D. P. Fan, K. Song, D. Liang, et al., Pyramid vision transformer: A versatile backbone for dense prediction without convolutions, in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, (2021), 568–578. https://doi.org/10.1109/ICCV48922.2021.00061

29. H. H. Newman, F. N. Freeman, K. J. Holzinger, *Twins: A study of Heredity and Environment*, Univ. Chicago Press, 1937.

30. Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, et al., Swin transformer: Hierarchical vision transformer using shifted windows, in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, (2021), 10012–10022. https://doi.org/10.1109/ICCV48922.2021.00986

31. J. M. J. Valanarasu, P. Oza, I. Hacihaliloglu, V. M. Patel, Medical transformer: Gated axial-attention for medical image segmentation, in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, (2021), 36–46. https://doi.org/10.1007/978-3-030-87193-2

32. R. Meleppat, M. Matham, L. Seah, An efficient phase analysis-based wavenumber linearization scheme for swept source optical coherence tomography systems, *Laser Phys. Lett.*, **12** (2015), 055601. https://doi.org/10.1088/1612-2011/12/5/055601

33. R. K. Meleppat, E. B. Miller, S. K. Manna, P. Zhang, E. N. Pugh Jr, R. J. Zawadzki, Multiscale hessian filtering for enhancement of OCT angiography images, in *Ophthalmic Technologies XXIX*, **10858** (2019), 64–70. https://doi.org/10.1117/12.2511044

34. R. K. Meleppat, K. E. Ronning, S. J. Karlen, M. E. Burns, E. N. Pugh, R. J. Zawadzki, In vivo multimodal retinal imaging of disease-related pigmentary changes in retinal pigment epithelium, *Sci. Rep.*, **11** (2021), 1–14. https://doi.org/10.1038/s41598-021-95320-z

35. R. K. Meleppat, M. V. Matham, L. K. Seah, Optical frequency domain imaging with a rapidly swept laser in the 1300nm bio-imaging window, in *International Conference on Optical and Photonic Engineering (icOPEN 2015)*, **9524** (2015), 721–729. https://doi.org/10.1117/12.2190530

36. H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, Pyramid scene parsing network, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, (2017), 2881–2890. https://doi.org/10.1109/CVPR.2017.660

37. J. Mei, R. J. Li, W. Gao, M. M. Cheng, Coanet: Connectivity attention network for road extraction from satellite imagery, *IEEE Trans. Image Process.*, **30** (2021), 8540–8552. https://doi.org/10.1109/TIP.2021.3117076

38. X. Yue, S. Sun, Z. Kuang, M. Wei, P. H. Torr, W. Zhang, et al., Vision transformer with progressive sampling, in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, (2021), 387–396. https://doi.org/10.1109/ICCV48922.2021.00044

39. K. Sirinukunwattana, J. P. Pluim, H. Chen, X. Qi, P. A. Heng, Y. B. Guo, et al., Gland segmentation in colon histology images: The glas challenge contest, *Med. Image Anal.*, **35** (2017), 489–502. https://doi.org/10.1016/j.media.2016.08.008

40. N. Kumar, R. Verma, D. Anand, Y. Zhou, O. F. Onder, E. Tsougenis, et al., A multi-organ nucleus segmentation challenge, *IEEE Trans. Med. Imaging*, **39** (2019), 1380–1391. https://doi.org/10.1109/TMI.2019.2947628

41. N. Kumar, R. Verma, S. Sharma, S. Bhargava, A. Vahadane, A. Sethi, A dataset and a technique for generalized nuclear segmentation for computational pathology, *IEEE Trans. Med. Imaging*, **36** (2017), 1550–1560. https://doi.org/10.1109/TMI.2017.2677499

42. J. Bernal, F. J. Sánchez, G. Fernández-Esparrach, D. Gil, C. Rodríguez, F. Vilariño, Wm-dova maps for accurate polyp highlighting in colonoscopy: Validation vs. saliency maps from physicians, *Comput. Med. Imaging Graphics*, **43** (2015), 99–111. https://doi.org/10.1016/j.compmedimag.2015.02.007

43. J. M. J. Valanarasu, V. A. Sindagi, I. Hacihaliloglu, V. M. Patel, Kiu-net: Overcomplete convolutional architectures for biomedical image and volumetric segmentation, *IEEE Transa. Med. Imaging*, **41** (2021), 965–976. https://doi.org/10.1109/TMI.2021.3130469

44. Y. Chen, K. Wang, X. Liao, Y. Qian, Q. Wang, Z. Yuan, et al., Channel-unet: A spatial channel-wise convolutional neural network for liver and tumors segmentation, *Front. Genet.*, **10** (2019), 1110. https://doi.org/10.3389/fgene.2019.01110

45. N. Chinchor, B. M. Sundheim, Muc-5 evaluation metrics, in *Fifth Message Understanding Conference (MUC-5): Proceedings of a Conference Held in Baltimore, Maryland*, 1993. https://doi.org/10.3115/1072017.1072026

46. S. Niwattanakul, J. Singthongchai, E. Naenudorn, S. Wanapu, Using of jaccard coefficient for keywords similarity, in *Proceedings of the International Multiconference of Engineers and Computer Scientists*, **1** (2013), 380–384.