RAFF-Net: An improved tongue segmentation algorithm based on residual attention network and multiscale feature fusion

Haibei Song, Zonghai Huang, Li Feng, Yanmei Zhong, Chuanbiao Wen and Jinhong Guo

Abstract

Objective: Due to the complexity of face images, tongue segmentation is susceptible to interference from uneven tongue texture, lips and face, resulting in traditional methods failing to segment the tongue accurately. To address this problem, RAFF-Net, an automatic tongue region segmentation network based on residual attention network and multiscale feature fusion, was proposed. It aims to improve tongue segmentation accuracy and achieve end-to-end automated segmentation.

Methods: Based on the UNet backbone network, different numbers of ResBlocks combined with the Squeeze-and-Excitation (SE) block was used as an encoder to extract image layered features. The decoder structure of UNet was simplified and the number of parameters of the network model was reduced. Meanwhile, the multiscale feature fusion module was designed to optimize the network parameters by combining a custom loss function instead of the common cross-entropy loss function to further improve the detection accuracy.

Results: The RAFF-Net network structure achieved Mean Intersection over Union (MIoU) and F1-score of 97.85% and 97.73%, respectively, which improved 0.56% and 0.46%, respectively, compared with the original UNet; ablation experiments demonstrated that the improved algorithm could contribute to the enhancement of tongue segmentation effect.

Conclusion: This study combined the residual attention network with multiscale feature fusion to effectively improve the segmentation accuracy of the tongue region, and optimized the input and output of the UNet network using different numbers of ResBlocks, SE block, multiscale feature fusion and weighted loss function, increased the stability of the network and improved the overall effect of the network.

Keywords

Deep learning, tongue segmentation, attention mechanism, multiscale feature fusion, weighted loss function

Submission date: 3 October 2022; Acceptance date: 15 October 2022

Introduction

Traditional Chinese medicine (TCM) is characterized by the concept of “inspecting the exterior to judge the interior,” and as a traditional auxiliary method of diagnosis, tongue inspection in TCM has been under clinical practice for thousands of years. According to TCM, the tongue manifestation reveals information about human health and disease, for example, the changes of the tongue appearance could be treated as the indicators of the pathogenic qi, zang-fu organs, qi-blood, body fluids, moreover, and the
changes of the tongue appearance can reveal the disease progress. Therefore, tongue inspection displays a certain clinical significance.\textsuperscript{1} The performance of TCM tongue diagnosis involves the practitioner observing and distinguishing the changes of the patient’s tongue to conduct syndrome differentiation, however, the diagnostic results of TCM tend to be influenced by the profession of practitioners and external factors. Therefore, it is of great importance to explore the objective measurement of tongue diagnosis to facilitate the modernization of TCM treatment.\textsuperscript{2–6} Besides, the objectification and concretization of tongue diagnosis have emerged as a trend of research in the related field.\textsuperscript{7–8}

Recently, tongue segmentation techniques based on deep learning have gradually emerged as a research hotspot. For example, Huang et al.\textsuperscript{9} applied deep learning methods to construct a tongue segmentation model for the segmentation of mobile-acquired tongue images in open and complex environments. Ruan et al.\textsuperscript{10} constructed an efficient tongue image segmentation model by optimizing the UNet network and designed a new network to specifically handle tongue edge segmentation. Cai et al.\textsuperscript{11} designed a novel auxiliary loss function to improve the tongue segmentation performance of supervised deep convolutional neural networks. Wang et al.\textsuperscript{12} proposed a cascade combination-based tongue segmentation algorithm for convolutional neural networks. Moreover, Zhou et al.\textsuperscript{13} proposed a new tongue localization and segmentation model for multitask learning using a feature pyramid network with residual blocks for feature extraction. Zhou et al.\textsuperscript{14} combined UNet’s backbone segmentation network and morphological layer postprocessing to improve the accuracy and speed of single image processing. Zhang et al.\textsuperscript{15} advanced the DeepLabV3 network decoder to significantly mitigate the ambiguity of tongue image segmentation and added a postprocessing module to improve the accuracy of segmentation near and away from the tongue edge. Besides, Jiang et al.\textsuperscript{16} designed a tongue image segmentation method using an enhanced Hue, Saturation, Value (HSV) color model convolutional neural network. Huang et al.\textsuperscript{17} employed a deep residual network as an encoder, a feature pyramid network as a decoder, and fused multiscale feature maps to achieve better tongue segmentation. Zhang et al.\textsuperscript{18} combined Atrus convolutional algorithm, Atrus spatial pyramid pooling module and fully connected conditional random field to refine the edges of segmented tongue images.

The abovementioned work manifests that deep learning techniques possess certain feasibility and advantages in tongue segmentation, but due to the rapid technological update, the advantages and disadvantages of different models in tongue segmentation should be explored and discussed. Generally speaking, tongue features tend to pose a higher challenge to the encoder of the network to extract effective features, such as the varied size and shape of the tongue, the color of the tongue body similar to the color of the lips and face, the unclear outline of the tongue edge, as well as pathogenic manifestation, such as tooth marks, punctures, and cracks on the tongue. At the same time, due to the limited samples for training tongue neural networks, the overfitting problem can easily occur; although some overly deep networks achieve good segmentation, the amount of parameters is too large, which reduces the performance of the network in clinical applications.

Taking into account the abovementioned issues in tongue segmentation, the algorithm proposed in this article mainly has the following improvements.

1. Based on the UNet backbone network, different numbers of ResBlocks combined with the Squeeze-and-Excitation (SE) block was used as the encoder to improve the ability of the downsampling network to extract image-layered features. It could effectively filter out the effects of tongue image noise (such as facial hair, tongue distribution, tongue angle, etc.).

2. The decoder structure of UNet was simplified by removing the 3×3 convolutional layers in the upsampling process, reducing computation and number of parameters and simplifying model; the coded feature layer in the skip connection layer was spliced with the decoded feature map after 1×1 convolution, better recovering the detail location information of feature maps lost in the downsampling layer in the encoder.

3. A multiscale feature fusion method was proposed to fuse the decoder feature maps of each layer after deconvolution and 1×1 convolution, and optimize the network parameters by combining the custom weighted loss function to improve the performance of network tongue segmentation. It could effectively solve the detailed segmentation problems such as tongue tip and edge concavity in tongue body segmentation.

**Relevant works**

**UNet**

Ronneberger et al.\textsuperscript{19} proposed the UNet network structure and successfully applied it to medical image segmentation featuring small sample datasets, besides improved models with UNet as the backbone network\textsuperscript{20–24} have demonstrated its excellent performance in various applications. As shown in Figure 1, the UNet network is divided into two parts: feature extraction for the encoder and feature recovery for the decoder. The encoder refers to the process of downsampling the image, while the decoder is the process of upsampling the feature map. The corresponding encoder and decoder are stitched together for the feature map through a skip connection, which can achieve the purpose of recovering the image detail information through the upsampling layer.
The UNet network consists of 23 convolutional layers. The encoder extracts the feature information of the image through a typical convolutional neural network structure, consisting of $3 \times 3$ convolutional layers, rectified linear units (ReLU) function and $2 \times 2$ maximum pooling layers, and performs 4 times downsampling. After each pooling operation, the feature map size is halved while the number of channels is doubled. Corresponding to the encoder part, the decoder performs upsampling through $2 \times 2$ deconvolution layers, followed by $3 \times 3$ convolution layers to gradually recover the image information, and performs a total of 4 times upsampling, and each upsampling will double the feature map size while halving the number of channels. The feature maps corresponding to the encoder and decoder are stitched together by skip connection, which can preserve better-detailed location information with a shallow network to assist in segmentation.

**Residual networks**

At present, neural network technology has obtained extensive achievements in the research fields of computer vision, automatic control, signal processing, and assisted decision making. Theoretically, the deeper the layers of neural networks is, the better the results will be, but in the actual processing, the deeper the network is, the harder to train and the model will be in a state where it is difficult to converge. The ResNet network\(^2\)\(^5\) solves the problem of disappearing gradients and difficult optimization of deep networks by introducing a skip-connected residual network, and the residual network structure block superimposes the output of the upper layer onto the input of the lower layer, improving the network performance.

Figure 2 illustrates the basic residual units that make up ResNet50, ResNet101, and ResNet152, consisting of a constant connection path and a residual path. The residual path consists of two $1 \times 1$ convolutional layers, a $3 \times 3$ convolutional layer, batch normalization, and ReLU activation function, and then the results of the two paths are summed to the output. At the same time, the skip connection does not introduce an additional number of parameters and computational complexity. Compared with the structure of...
ResNet34, the basic structure of the residual block is computationally optimized, that is, two $3 \times 3$ convolutional layers are replaced by two $1 \times 1$ convolutional layers and one $3 \times 3$ convolutional layer, which achieves the purpose of maintaining the accuracy and reducing the computation.

Attention mechanism

Attention mechanisms have been widely employed in various fields such as image processing, speech recognition, and natural language processing for deep learning in recent years.\(^{26–28}\) By introducing the attention mechanism, neural networks focus on the information that is more critical to the current task among a large amount of input information, reduce the attention to other distracting information, and improve the efficiency and accuracy of task processing. Therefore, attention resources can be used to quickly extract features with high-level semantics from a large amount of information.

The SE block is a channel attention method proposed by Hu et al.,\(^ {29}\) which can learn the relationship between channels in the model, enhance important features and suppress useless features. As shown in Figure 3, the SE block is divided into two basic stages: Squeeze and Excitation.

SE involves explicit modeling of the weights on each channel and then weighting the original feature map so that each channel has a different level of importance, which also implements the channel attention mechanism. Starting from a single image, image features are extracted, and the feature map dimension of the current feature layer is $W \times H \times C$. The $[H, W]$ dimension of the feature map is averaged pooled or maximally pooled (in our work, averaged pooling was conducted), and the size of the pooled feature map changes from $W \times H \times C$ to $1 \times 1 \times C$. The operation first goes through a fully connected layer, which reduces the number of channels from $C$ to $C/r$, and then goes through the ReLU function. The hyper parameter $r$ controls the amount of computation of the SE block and exerts different effects on the network performance. The value of $r$ will be discussed specifically in the experimental section. Then it goes through a fully connected layer where the number of channels is expanded from $C/r$ to $C$, followed immediately by a sigmoid function. After the obtained weight of $1 \times 1 \times C$ for each channel $C$, the weights are applied to the feature map $W \times H \times C$, that is, each channel is multiplied by its respective weight.

Improved network model

**RAFF-NET**

RAFF-Net applies the UNet network as the backbone network and is mainly divided into three parts: encoder, decoder, and multiscale feature fusion (as shown in Figure 4), including five levels of encoder, four levels of decoder, and a multiscale feature fusion module connected to the decoder and the last level of the encoder, which can realize end-to-end tongue segmentation training.

The main function of the encoder is to extract the useful features of the image, filter out the image noise, and retain the high-level semantic information features, exerting a substantial impact on the final performance of the tongue segmentation. The original UNet is downsampled 4 times in the encoder module, and each layer of the feature map passes through two $3 \times 3$ convolutional layers with ReLU functions and a $2 \times 2$ maximum pooling layer. Considering ResNet is more capable of extracting target features, the improved ResNet50 was used as the encoder of the improved model, and the SE block was fused between the residual modules of ResNet to enhance the encoder’s ability to extract image-layered features.

The encoder removed the ResNet50 final average pooling layer and the fully connected layer; the first $7 \times 7$ convolutional layer, the maximum pooling layer, and the number of residual modules of 3, 4, 6, and 3, respectively, were retained. All the residual modules formed four residual module groups, and the number of input and output image channels of each residual module group were denoted by IN and OUT, respectively (Figure 5). Due to the complexity and performance of the model, the encoder of the improved model halved the number of channels of the original ResNet50, so the residual module was changed from channel number 32 to the final output channel number 1024. Meanwhile, the input image aspect size of the encoder was $512 \times 512$, which is significantly different from the traditional input image aspect size of $224 \times 224$ for ResNet. The SE block was added after the first $7 \times 7$ convolutional layer and the four residual module groups in Figure 5(b). The number of channels and aspect dimensions of the SE block were consistent with the output feature map dimensions of its previous modules. The hyper parameter $r$ of each SE block was set to 4. The placement of the SE blocks will affect the overall network performance, so the exact placement will be analyzed in the experimental section. The encoder was downsampled 4 times, and the input image size was $3 \times 512 \times 512$, which was finally downsampled to $1024 \times 16 \times 16$ pixel size with the modified ResNet and SE blocks.

The ResBlock in the ResNet50 structure contains two basic blocks: the Convolution Block and the Identity Block. As displayed in Figure 2, the Identity Block consists of two $1 \times 1$ convolutional layers, one $3 \times 3$ convolutional layer, batch normalization, ReLU activation function, and constant connectivity. Figure 6 illustrates the disparity between the Convolution Block and the Identity Block, where $W$ and $H$ are the length and width dimensions of the feature map, IN and OUT are the number of feature map channels before and after the residual module processing, respectively, and int() is the integer taking operation. As shown in Figure 6, the left no branch is the convolution
block, whose input and output dimensions are variable and cannot be continuously connected in series for changing the network dimension; $1 \times 1$ convolution is performed in the shortcut path (shortcut path), and then the dimension is changed in the main path (main path) and corresponds to the shortcut path. The yes branch on the right side of Figure 6 is an Identity Block with the same input and output dimensions, which can be concatenated and used to deepen the network.

In the decoder module, similar to the upsampling method in UNet, the $2 \times 2$ deconvolution with a step size of 2 was used, and the length and the width of the feature map were expanded to 2 times the original size with each upsampling, and a total of 4 upsampling operations were performed, and the original map size of $512 \times 512$ pixels was finally restored. The original UNet decoder halves the number of channels of the output feature maps after each upsampling, while the improved decoder output 128 channels of feature maps for each layer (the number of 128 channels was the result of the performance comparison in the experimental part), and removed the two $3 \times 3$ convolutional layers in the original UNet decoder after the splicing of the feature maps for each layer, which reduces the computation and the number of parameters and simplifies the model, and this is also the key operation that makes the number of parameters of the improved network model equal to the original UNet.

In order to facilitate the decoder to better recover the detailed location information that is lost in the downsampling layer of the feature map during upsampling in the encoder, a skip connection between the encoder and decoder at the corresponding feature layer was performed. This approach allows the decoder to recover image information more efficiently by leveraging the encoder’s detailed location information in the shallow network. Unlike the skip connection process of the original UNet, our encoder feature layer went through $1 \times 1$ convolution to reach the same number of channels of the decoder feature layer, and then the two are spliced together. The feature map in the encoder successively went through the residual block, SE block, and $1 \times 1$ convolution layer, while the feature map of its upper decoder layer was upsampled and the $1 \times 1$ convolution layer was operated, and then the two results were stitched together. After the feature map splicing, it passed through the batch normalization layer and ReLU function, and then went through the $2 \times 2$ deconvolution layer operation, and then repeated the above splicing process.

Combination of shallow and deep feature map is shown in Figure 7. The encoder and decoder were connected by $1 \times 1$ convolutional layers, which can better recover the detail location information lost in the downsampling layer of the feature map in the encoder, and the encoder output result was transmitted to the ResBlock module in the next layer, and the decoder recovered by the upsampling module, where $n$ denotes the number of ResBlocks in the encoder.

For more accurate segmentation of the tongue, a multiscalar feature fusion module was added to deepen the network to obtain detailed features at each layer (Figure 4). The lower-level features have a higher resolution and contain more location and detail information, but there is low semanticity and more noise due to less convolution; although the higher-level features have stronger semantic information, the resolution is quite limited. Consequently, they are less perceptive of details. Unlike the original UNet model that only performs feature supervision in the last layer of the decoder, our improved model, the multiscalar feature fusion module performed multiscale, and multilevel feature learning for the whole network and fed the supervised information of each layer into the current layer, which achieves the purpose of learning more rich graded features and could effectively solve the detail segmentation problem in tongue segmentation. In the multiscalar feature fusion module, firstly, through $1 \times 1$ convolution and upsampling at different scales, 5 feature maps of each level (including the last encoder level and 4 decoder levels) were obtained, and the feature map dimensions were all $1 \times 512 \times 512$, and then these 5 feature maps were fused, after which the fusion map was obtained through $1 \times 1$ convolution and sigmoid.
activation function, and the final output was a 2-channel 512 × 512 probabilistic map with the same size as the original map. Each pixel position in the probability map had a probability value corresponding to each species (tongue or background). By taking out the index of the maximum probability value at each pixel position, the segmentation effect map of the model can be output.

**Loss function**

The commonly used loss functions in image segmentation models are cross-entropy, mean squared difference, and Dice loss functions, but these loss functions are equally weighted for each image, and a variety of optimized loss functions have been proposed by research scholars in recent years. In the process of tongue image acquisition, there are significant differences in the size of the tongue due to factors such as tongue extension strengths of the subjects and various facial tilt angles. For tongue images with a small percentage of the tongue body, the calculated loss value is small when there is an obvious error in segmentation and thus leads to undersegmentation. The advantage of the cross-entropy loss function is that it can...
better measure the difference between the predicted and true values. So a new loss function was constructed based on the cross-entropy loss function as the loss function of the improved model. A balance factor $\omega$ was introduced to construct the weighted loss function calculated as:

$$L_{\text{total}} = \sum_{n=1}^{Q} \omega_n l_n + \omega_{\text{mix}} l_{\text{mix}}$$  (1)

$l_n$ denotes the loss of the output feature map of the 4-level decoder and the last-level encoder in Figure 4, $Q = 5$; $l_{\text{mix}}$ is the loss of the multiscale feature fusion result of the Q-level output. For each loss, we used the standard cross-entropy to calculate the loss as follows:

$$1 = \sum_{i} - [y_i \log a_i + (1 - y_i) \log (1 - a_i)]$$  (2)

$y_i$ denotes the true value of the pixel category of the $i$th picture and $a_i$ the predicted value of the pixel category for the $i$th picture. $\omega_n$ and $\omega_{\text{mix}}$ are the weight of each loss. The setting is calculated as follows:

$$\omega = \begin{cases} \frac{\sum_{i=0}^{N} |T_i|}{N * |T_i|}, & p \in T_i \\ 1, & p \notin T_i \end{cases}$$  (3)

$N$ denotes the number of all labels, $p$ represents the number of pixel points in the sample, and $T_i$ indicates the number of all pixels of the current label.

**Experimental results and analysis**

**Dataset and preprocessing**

The tongue image dataset was obtained from the community outpatient clinic of the Third Affiliated Hospital of Chengdu University of Traditional Chinese Medicine, which was acquired by the tongue-face instrument (camera Sony IMX317, light source LED, color temperature 6500k) developed by Chengdu University of Traditional Chinese Medicine, with the ethical support certification number 2021KL-027. The tongue image dataset consists of 1461 images, with Portable Network Graphic (PNG) format and $1920 \times 1920 \times 3$ pixels original image size. Mask labels are JavaScript object syntax (JSON) format files of tongue areas manually labeled by professional physicians using labelme.

The experiments started with preprocessing the images, in order not to lose the image information, the method of adding white background edges was used, and the original images were uniformly converted to $1920 \times 1920 \times 3$ pixels square before training, and then the filled square was scaled to $512 \times 512 \times 3$ size images.

**Experimental process**

The experiments were based on Windows Server 2016 operating system, with Intel(R) Xeon(R) E5-2678 V3 processor and 256GB memory. Tensorflow 2.6.0 and Keras 2.6.0 were used as the main model deep learning framework to build the model, and graphic processing unit (GPU) was used to accelerate the training and testing of the network mode. Besides, the graphics card model was NVIDIA TESLA V100, the Compute Unified Device Architecture (CUDA) version was 11.3, and the experimental code was implemented based on Python 3.9.

The RAFF-Net model was based on UNet as the backbone network, and the dataset was randomly divided into 80% training dataset and 20% test dataset with 1169 and 292 pictures, respectively. Adam was used as the optimizer for the experiments because it can adjust the learning rate adaptively during training and has a faster
convergence rate, with the initial learning rate set to 1e-5. The batch size was set to 12 in the training phase, and the batch size was set to 1 in the testing phase, and a total of 150 rounds were trained. The changes of the loss values during the iterations are shown in Figure 8. After 120 batches of training, the loss of the model remained stable.

**Evaluation indicators**

The performance of RAFF-Net was evaluated from multiple perspectives using metrics such as MIoU, F1-score, accuracy, recall, the number of model parameters, Million Floating-point Operations per Second (MFLOPs) and Runtime to better explore the potential of the improved model for tongue segmentation applications.

Accuracy and recall: Accuracy refers to the proportion of correctly predicted tongue regions to all predicted lingual regions, and recall refers to the proportion of correctly predicted lingual regions to all actual lingual regions.

\[
P = \frac{TP}{TP + FP} \quad (4)
\]

\[
R = \frac{TP}{TP + FN} \quad (5)
\]

P denotes Precision and R denotes Recall. TP denotes the lingual region of the network output, which is also the lingual region; TN, the background region of the network output, which is also the background region; FP, the lingual region of the network output, but the background region; and FN, the background region of the network output, but the lingual region.

MIoU, a standard metric for computing semantic segmentation of models, can calculate the average of the ratio of intersection and union of all categories. In the experiment, it is calculated as the value between the real image and the predicted graph and is calculated as follows:

\[
\text{MIoU} = \frac{1}{k + 1} \sum_{i=0}^{k} \frac{TP}{TP + FP + FN} \quad (6)
\]

\(k + 1\) is the number of pixel categories.

In order to balance the effects of accuracy and recall and to evaluate a classifier more comprehensively, the F1-score is the summed average of accuracy and recall and is calculated as

\[
\text{F1-score} = \frac{2 \times P \times R}{P + R} \quad (7)
\]

P denotes Precision, R denotes Recall, and F1-score is the summed mean of the two, between 0 and 1. The closer to 1, the better the result.

Parameters are the parameters contained in the model, which can visually reflect the model size. MFLOPs measure the number of forward-propagating operations in the neural system. Runtime is used to measure the training speed of network model.

**Experimental results**

**Ablation experiment.** Since there are variable terms in the improved model, the settings in the experiments are first proved to be optimal by the ablation experiments. The presence of “±” in the following data indicates the mean and variance obtained through three replicate experiments. The ablation experiments mainly include the following:

1. Comparison of the results of the encoder module, SE block on the impact of the value of the hyper parameter r in the performance.
2. Encoder module, comparison of the results on the impact of the SE block’s addition position in the segmentation network on the performance.
3. Decoder module, comparison of the results on the impact of the different number of channels on the performance when skip-connected splicing.
4. Comparison of the results of the performance impact of different modules.

**Table 1.** Effect of parameter r on the performance of SE block.

| Parameter r | MIoU (%) | F1-score (%) | Parameters (MB) |
|-------------|----------|--------------|-----------------|
| 2           | 92.11 ± 0.0031 | 92.07 ± 0.0028 | 7.98           |
| 4           | 97.85 ± 0.0021 | 97.73 ± 0.0023 | 7.32           |
| 8           | 93.35 ± 0.0041 | 93.07 ± 0.0036 | 6.99           |
| 16          | 94.51 ± 0.0023 | 94.27 ± 0.0022 | 6.82           |

SE: Squeeze-and-Excitation; MB: Megabyte.
As shown in Table 1, the r size was set to 2, 4, 8, and 16, respectively, and it was observed that the number of model parameters gradually decreased as the r value gradually became larger, but the decrease was not significant. MIoU and F1-score reveal that the optimal results are obtained on the tongue image dataset when r is 4. Therefore, the hyperparameter r of the SE block is set to 4 in all other subsequent experiments.

The SE block can be added to different positions of the network structure. In our study, the ResNet network combined with the SE block was applied to improve the semantic extraction ability, so the SE block was placed in the encoder part in the experiment. With f1, f2, f3, f4, and f5 denoting the output of feature maps at each level of the 5-level encoder, the SE blocks were added to the encoder at the following positions:

1. S0: without the addition of SE block.
2. S1: after encoders f1, f2, f3, f4, and f5.
3. S2: after encoders f2, f3, f4, and f5.
4. S3: after encoder f2, f3, and f4.

As shown in Table 2, the maximum improvement in model performance was observed when the SE block was placed at position S2. In the tongue image dataset, when the SE block was added at the S2 position, the MIoU improved by 22.23% compared to that without the SE block. The SE block was placed at the S2 position in the rest of the experiments.

| SE blocks positions | MIoU (%) | F1-score (%) | Recall (%) | Precision (%) |
|---------------------|----------|--------------|------------|---------------|
| S0                  | 75.62 ± 0.0013 | 75.48 ± 0.0018 | 74.20 ± 0.2102 | 76.81 ± 0.2085 |
| S1                  | 96.27 ± 0.0061 | 96.20 ± 0.0057 | 94.27 ± 0.3062 | 98.21 ± 0.3248 |
| S2                  | 97.85 ± 0.0021 | 97.73 ± 0.0023 | 98.72 ± 0.2311 | 96.75 ± 0.2723 |
| S3                  | 96.31 ± 0.0027 | 96.23 ± 0.0029 | 94.63 ± 0.4098 | 97.88 ± 0.4311 |

SE: Squeeze-and-Excitation.

Table 3 presents the effect of the number of channels connected by skip on the performance.

| Channel number | MIoU (%) | Parameters (MB) | FLOPs (MFLOPS) |
|----------------|----------|-----------------|----------------|
| 64             | 80.40 ± 0.0037 | 6.73            | 14.02          |
| 128            | 97.85 ± 0.0021 | 7.32            | 15.26          |
| 256            | 97.35 ± 0.0018 | 9.06            | 18.91          |
| 512            | \         | 14.80           | 30.94          |

FLOPs: Floating-point Operations per Second; MFLOPs: Million Floating-point Operations per Second; SE: Squeeze-and-Excitation; MFLOPs: Million Floating-point Operations per Second.

As shown in Table 1, the r size was set to 2, 4, 8, and 16, respectively, and it was observed that the number of model parameters gradually decreased as the r value gradually became larger, but the decrease was not significant. MIoU and F1-score reveal that the optimal results are obtained on the tongue image dataset when r is 4. Therefore, the hyperparameter r of the SE block is set to 4 in all other subsequent experiments.

The SE block can be added to different positions of the network structure. In our study, the ResNet network combined with the SE block was applied to improve the semantic extraction ability, so the SE block was placed in the encoder part in the experiment. With f1, f2, f3, f4, and f5 denoting the output of feature maps at each level of the 5-level encoder, the SE blocks were added to the encoder at the following positions:

1. S0: without the addition of SE block.
2. S1: after encoders f1, f2, f3, f4, and f5.
3. S2: after encoders f2, f3, f4, and f5.
4. S3: after encoder f2, f3, and f4.

As shown in Table 2, the maximum improvement in model performance was observed when the SE block was placed at position S2. In the tongue image dataset, when the SE block was added at the S2 position, the MIoU improved by 22.23% compared to that without the SE block. The SE block was placed at the S2 position in the rest of the experiments.

Table 3 presents the effect of the number of channels connected by skip on the performance. In order to simplify the model, the decoder module in the improved model was set to the consistent number of channels in the skip connection and the number of upsampling channels, and the channel numbers of 64, 128, 256, and 512 were adopted for the experiments. It reveals that the number of 256 channels reduced the MIoU by 0.50% compared with the number of 128 channels, and the performance was reduced, while the number of parameters and computation volume of 256 channels were not advantageous compared with 128 channels. The 64-channel number was not considered because the segmentation accuracy was not high and the 512-channel was not considered due to the number of parameters that caused the model to be too large and no operational performance results, so the final improved model selected the 128-channel number as the selected value for the decoder module, and the number of channels for all other experimental decoder modules was set to 128.

The original UNet network was used as the basis for the different module additions, where the encoder used the residual network while the decoder both used the simplified method of removing the convolution after skip-connected splicing. As seen in Table 4, after the encoder was replaced with the improved ResNet50, the MIoU of the model was 72.91% without combining the SE block for optimization, which was very ineffective; it could be attributed to a large area of the left image edge being incorrectly identified as...
the target region, which was not seen in the other module ablation experiments. After the residual network was combined with the SE block as the encoder structure, the weighted loss function and the multiscale feature fusion method were used simultaneously, and the tongue segmentation effect was optimal, and the MIoU was 96.38% after replacing the weighted loss function using the traditional cross-entropy, and the segmentation effect was significantly reduced.

Figures 9 to 12 present the detailed analysis of the experimental results of the tongue dataset images in terms of lip color, facial tilt angle, hair, and background, respectively. C1 to C6 in each figure correspond to the ablation experiment module in Table 4, where C1 is the base model UNet network.

Figure 9 illustrates the effect of the proximity of lip color to tongue color on the tongue body segmentation experiment. In the figure, the tongue color was basically the same as the lip color, and there were convex parts on both sides of the tongue body in some cases. C1 in Figure 9 was a relatively accurate segmentation effect that can be achieved by using UNet for the original image, but the excessive transmission of low-level details could easily lead to oversegmentation, so the existing model still needs to be improved to accurately segment the part of the tongue edge that is close to the lip color. In the image segmentation, the attention mechanism typically focuses on a specific target, so C3 compared to C2 was to add the constraints of the channel attention SE module on top of the encoder for ResBlocks, and the mis-segmentation case of the background was significantly reduced, but the oversegmentation was severe. According to C4/C5 of the results in Figure 9, both weighted loss function and multiscale feature fusion improved the tongue side oversegmentation situation, and after the combination of the two in C6, RAFF-Net could segment the tongue side and lip of many scenes, especially the right side of the image P1 tongue and the left side of the P4 tongue, and the segmentation edge was more smooth and could filter out the interference of the adjacent area lips better.

Figure 10 presents examples of different degrees of left, right, and downward tilt of the face during imaging, and it is evident that the incomplete display of the tongue body and the left–right asymmetry contribute to the difficulty of tongue segmentation. The performance of C1 in Figure 10 in the angular tilt category outperformed significantly the tongue segmentation effect of C1 under the influence of lip color in Figure 9, but there was still a significant gap between its refinement in the tongue tip part and tongue side compared with the RAFF-Net algorithm. In several other modular ablation cases, C3 segmentation edge jitter was very obvious, and C4 and C5 undersegmented seriously for the tongue tip part of image P1.

Figure 11 presents comparison of the effect of facial hair on the results of tongue segmentation experiments. The facial whiskers around the tongue body in Figure 11 played a larger interference role. In Figure 11, C1 was segmented by the UNet network, the lower lip of P1 and the chin of P3 were oversegmented into the tongue body part, and the interference of the facial whiskers were filtered out under the RAFF-Net effect of C6, and a more accurate tongue body segmentation could be achieved. P2 in Figure 11 showed little change in the processing of different models from C1 to C6, the undersegmentation of the left side of the tongue body was obvious in C3, and the tongue tip segmentation was more complete in C6 compared with the other models. Whiskers in the upper and lower parts of the mouth in both P1 and P3 had a serious impact on the tongue segmentation, and the multiscale fusion in C5 was more resistant to interference than the weighted loss function in C4. From the above, it can be seen that the multiscale feature fusion displayed better detail segmentation from the perspective of the two modules with enhanced detail features, and the improved model with the best results after the three modules fast together under the constraint of the custom weighted loss function.

Figure 12 illustrates that the overextended tongue body or the angle of facial tilt resulted in the tongue tip background changing from mostly facial chin skin to the white inner wall of the imaging device. According to C1 in Figure 12, the difficulty of UNet in tongue segmentation of such case was in the lower half of the tongue, where the tongue tip ends of P1, P2, and P3 were obviously undersegmented, and the tongue tip segmented by P4 was jittery and had oversegmented regions. The segmentation performance of C4 weighted loss function and C5 multiscale fusion module was slightly better than that of UNet in C1, while RAFF-Net in C6 was more accurate for such special scenes of the RAFF-Net of C6 is more accurate for tongue segmentation in this particular scene. Obviously, by integrating the improved encoder, multiscale feature fusion,

### Table 4. Different module ablation experiments on performance effects.

| Number | ResNet | SE | Weighted loss function | Multiscale feature fusion | MIoU (%) |
|--------|--------|----|------------------------|--------------------------|----------|
| C1     | x      | x  | x                      | x                        | 97.29 ± 0.0036 |
| C2     | √      | x  | x                      | x                        | 72.91 ± 0.0012 |
| C3     | √      | √  | x                      | x                        | 93.32 ± 0.0035 |
| C4     | √      | √  | √                      | x                        | 97.22 ± 0.0016 |
| C5     | √      | √  | x                      | √                        | 96.38 ± 0.0072 |
| C6     | √      | √  | √                      | √                        | 97.85 ± 0.0021 |

SE: Squeeze-and-Excitation.
and weighted loss function, we could reduce the noise transmission, enhance the detail segmentation effect, and improve the constraints of tongue images with different characteristics, so as to achieve a better tongue image segmentation under the condition of diverse tongue display shapes.

Figure 13 is a failure case that is difficult to solve at present in the experiment. The tongue coating was greasy and severely unevenly distributed, with obvious midline marks at the tip of P1 and unsmooth edges at the tip of P2. In Figure 13, from the UNet network of C1 to the RAFF-Net network of C6, there were obvious undersegmentation or mis-segmentation in the tongue tip part of the greasy tongue coating. Combined with other classical networks and tongue segmentation networks in Table 5, there were no better segmentation results, indicating that the improved method of this study also existed unsolvable segmentation scenarios, and further experiments were needed to analyze this particular situation.

Comparison experiments with other segmentation methods. The tongue segmentation results of the RAFF-Net were compared with those of several typical models such as UNet, U2Net, SE UNet, Attention UNet, ResNet, DeepLabV3, and all the results were obtained after 150 epochs of training were obtained, and the results are shown in Table 5. The performance of the latest improved models related to tongue segmentation was also compared, involving several tongue segmentation models mentioned in the literature. To ensure the objectivity of the model performance judging metrics, seven metrics, MIoU, F1-score, Recall, Precision, number of parameters, MFLOPs, and Runtime, were chosen for a comprehensive comparison.

Among the three basic frameworks of UNet, DeepLabV3, and ResNet, the UNet network obtained better segmentation results than both DeepLabV3 and ResNet, and the UNet network had the best MIoU metrics with 1/5 and 1/9 of the number of parameters of DeepLabV3 and ResNet, respectively, which shows that the UNet model has very excellent performance advantages in the tongue segmentation task. Overall, the indicators MIoU and F1-score were relatively
stable, while the indicators Recall and Precision were more fluctuating.

In the UNet model improved by the attention mechanism, there was overfitting for segmentation of small datasets with large variation, and both attention mechanism-based models showed a trend of decreasing segmentation MIoU. RAFF-Net could adapt well to the interference of different tongue shapes and differences in tooth mark details. Compared with UNet and other tongue segmentation model,9,15,17,42,43 our model RAFF-Net improved MIoU by 1.54%, 2.63%, 0.70%, 6.18%, 1.70%. The comparison between the improved algorithm RAFF-Net and OETNet9 in this article revealed that RAFF-Net had fewer numbers of parameters, and the computing time of both was basically comparable. The application scenarios of the two tongue segmentation model were obviously different, such as shooting distance, light changes, image characteristics, etc. In terms of implementation principles, the encoder, and decoder of RAFF-Net were modified more compared to the original UNet, while OETNet focused on the improvement of the skip connection layer and feature map fusion, and both feature map fusion modules were based on the significant map fusion model of the U2Net37 network, the difference lied in the setting of weights in the weighted loss function. The weights were all 1 in U2Net, the weights were set according to the proportion of the tongue to the whole image in OETNet, and RAFF-Net was set according to the sample label ratio.

In terms of the number of model parameters and FLOPs, the addition of the residual module superimposed on the SE block and the multiscale feature fusion module not only did not increase the parameters and computational workload of the original backbone model, but also slightly reduced the number of parameters of the improved network model 7.32M compared to the original UNet model 7.40M, as shown in Table 5.

In terms of the running time of the models, RAFF-Net, UNet, and OETNet consumed the least time with insignificant difference, followed by SE UNet, Attention UNet, and DENet models. The U2Net model took the longest time. Thus, it can be seen that the network training time was related to the number and complexity of model parameters, but not linearly, and the training time was also related to the convergence speed and inference ability of the network model itself.
Figure 11. Comparison of the effect of facial hair on the results of tongue segmentation experiments.

Figure 12. Comparison of the effect of imaging background on the results of tongue segmentation experiments.
**Discussion**

The morphology and features of the tongue body in the tongue image can reflect the nature and depth of disease of the five organs, such as chronic gastritis, coronary heart disease, as well as chronic obstructive pulmonary disease. The automatic and precise segmentation of the tongue region can assist the physician in disease diagnosis. Therefore, a UNet-based tongue segmentation network,

![Figure 13.](image_url)  
*Figure 13. Experimental comparison of failed tongue segmentation cases.*

**Table 5.** Comparison results of MIoU, F1-score, Recall, Precision, number of parameters, FLOPs and Runtime for different models.

| Models       | MIoU (%)       | F1-score (%)    | Recall (%)      | Precision (%)   | Parameters (MB) | FLOPs (MFLOPS) | Runtime (MS) |
|--------------|----------------|-----------------|-----------------|-----------------|----------------|----------------|--------------|
| UNet19       | 97.29 ± 0.0036 | 97.29 ± 0.0038  | 98.32 ± 0.2549  | 96.28 ± 0.2632  | 7.40           | 15.51          | 59           |
| DeepLabV3+   | 95.37 ± 0.0047 | 95.34 ± 0.0052  | 99.77 ± 0.3901  | 91.28 ± 0.3362  | 40.12          | 82.16          | 118          |
| ResNet 101   | 96.43 ± 0.0027 | 96.61 ± 0.0035  | 97.22 ± 0.1912  | 95.61 ± 0.1743  | 67.60          | 141.05         | 156          |
| U2Net37      | 97.28 ± 0.0052 | 97.13 ± 0.0054  | 98.27 ± 0.4124  | 96.01 ± 0.3992  | 42.00          | 88.03          | 426          |
| SE UNet38    | 96.75 ± 0.0043 | 96.88 ± 0.0042  | 96.12 ± 0.2638  | 97.65 ± 0.2931  | 8.99           | 18.83          | 73           |
| Attention UNet39 | 97.16 ± 0.0026 | 97.23 ± 0.0021  | 97.53 ± 0.3150  | 96.93 ± 0.3523  | 8.94           | 18.74          | 82           |
| OETNet9      | 96.31 ± 0.0054 | 96.59 ± 0.0049  | 97.95 ± 0.3028  | 95.27 ± 0.2921  | 7.78           | 15.65          | 58           |
| DeepLabV3+   | 95.22 ± 0.0027 | 95.05 ± 0.0026  | 98.21 ± 0.4775  | 92.09 ± 0.4369  | 45.71          | 94.03          | 126          |
| TISNet17     | 97.15 ± 0.0042 | 97.16 ± 0.0037  | 98.02 ± 0.3037  | 96.32 ± 0.2541  | 36.13          | 78.22          | 175          |
| DENet2       | 91.67 ± 0.0042 | 91.17 ± 0.0044  | 99.30 ± 0.1753  | 84.27 ± 0.2214  | 8.37           | 17.02          | 72           |
| IrisNet43    | 96.15 ± 0.0027 | 96.50 ± 0.0026  | 96.97 ± 0.3198  | 96.03 ± 0.3332  | 1.54           | 3.72           | 66           |
| RAFF-Net     | 97.85 ± 0.0021 | 97.73 ± 0.0023  | 98.72 ± 0.2311  | 96.75 ± 0.2723  | 7.32           | 15.26          | 61           |

FLOPs: Floating-point Operations per Second; MFLOPs: Million Floating-point Operations per Second; SE: Squeeze-and-Excitation; MS: millisecond.
RAFF-Net, was proposed in this article to achieve end-to-end automatic segmentation. RAFF-Net used a ResNet50 with an improved fused attention mechanism as the encoder, which can more effectively extract image features; a streamlined decoder refined the segmentation effect by skip connection while reducing parameters and used a weighted loss function combined with a multiscale feature fusion module to feed the supervised information from each layer to the current layer, further improving the segmentation accuracy. Comparative experiments were done on the tongue image dataset and good results were obtained, which verified the effectiveness of the method in this article. RAFF-Net could also segment the tongue body effectively for the images with lip interference and lesions affecting the tongue body region. However, it is also noticed that some of the segmented tongue body boundaries are not very smooth. Therefore, in the next work, some latest skip-join operations which focus on the segmentation of the detailed information of the tongue body boundary can be introduced. Considering the actual segmentation effect of the tongue body, the segmentation accuracy can also be improved by postprocessing modules such as morphological hole filling and removal of small disconnected areas. Also, pretraining such as generative adversarial networks and migration learning can be employed for tongue segmentation.

Acknowledgments: This study has been approved by the ethics committee of affiliated hospital of Chengdu University of Traditional Chinese Medicine. The clinical trial registration number is 2021KL-027.

Declaration of conflicting interests: The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding: The authors disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported in part by the National Key Research and Development Program of the Ministry of Science and Technology of China (grant numbers 2018YFC1707606), National Nature Foundation of China (grant numbers 82004504), Chinese Medicine Administration of Sichuan Province (grant numbers 2021MS199).

Ethical approval: Written informed consent for publication of this article was obtained from the Chengdu University of Traditional Chinese Medicine and all authors.

Guarantor: CW.

Contributorship: HS was involved in conceptualization, writing original draft, and methodology. ZH was involved in validation and methodology. LF was involved in data curation and YZ in investigation. CW was involved in project administration, resources, and funding acquisition and JG in supervision and resources.

ORCID iDs: Haibei Song https://orcid.org/0000-0001-9793-8668
Jinhong Guo https://orcid.org/0000-0003-2659-3150

References
1. Chen H, Chen S, Jiang C, et al. Computational tongue color simulation in tongue diagnosis. Color Res Appl 2022; 47: 121–134.
2. Li L, Luo Z, Zhang M, et al. An iterative transfer learning framework for cross-domain tongue segmentation. Concurr Comput Pract Exp 2020; 32. Article ID: e5714.
3. Liu W, Zhou C, Li Z, et al. Patch-driven tongue image segmentation using sparse representation. IEEE Access 2020; 8: 41372–41383.
4. Sun P, Yang X and Ban Y. Weakly supervised deep learning for tooth-marked tongue recognition. Opt Sensing Imaging Technol 2019; 13: 11338.
5. Rhee S-Y. Tongue area segmentation and tongue cracks detection using CNN. J Korean Inst Intell Syst 2020; 30: 258–264.
6. Fan S, Chen B, Zhang X, et al. Machine learning algorithms in classifying TCM tongue features in diabetes mellitus and symptoms of gastric disease. Eur J Integr Med 2021; 43. Article ID: 101288.
7. Gao S, Guo N and Mao D. LSM-SEC: tongue segmentation by the level set model with symmetry and edge constraints. Comput Intell Neurosci 2021; 2021. Article ID: 6370526.
8. Zhou C, Fan H, Zhao W, et al. Reconstruction enhanced probabilistic model for semisupervised tongue image segmentation. Concurr Comput Prat Exp 2020; 32. Article ID: e5849.
9. Huang Z, Miao J, Song H, et al. A novel tongue segmentation method based on improved U-Net. Neurocomputing 2022; 500: 73–89.
10. Ruan Q, Wu Q, Yao J, et al. An efficient tongue segmentation model based on UNet framework. Int J Pattern Recognit Artif Intell 2021; 35. Article ID: 2154035.
11. Cai Y, Wang T, Liu W, et al. A robust interclass and intraclass loss function for deep learning based tongue segmentation. Concurr Comput Prat Exp 2020; 32. Article ID: e5849.
12. Wang L, Tang Y, Chen P, et al. Two-phase convolutional neural network design for tongue segmentation. J Image Graph 2018; 23: 1571–1581.
13. Zhou C, Fan H and Li Z. Tonguenet: accurate localization and segmentation for tongue images using deep neural networks. IEEE Access 2019; 7: 148779–148789.
14. Zhou J, Zhang Q, Zhang B, et al. Tonguenet: a precise and fast tongue segmentation system using UNet with a morphological processing layer. App Sci Basel 2019; 9. Article ID: 3128.
15. Zhang X, Bian H, Cai Y, et al. An improved tongue image segmentation algorithm based on DeepLab3 + framework. IET Image Proc 2022; 16: 1473–1485.
16. Jiang L, Xu B, Ban X, et al. A tongue image segmentation method based on enhanced HSV convolutional neural network. International Conference on cooperative design. Vis Eng 2017; 10451: 252–260.
17. Huang X, Zhang H, Zhuo L, et al. TISNet-enhanced fully convolutional network with encoder-decoder structure for tongue image segmentation in traditional Chinese medicine. Comput Math Methods Med 2020; 2020. Article ID: 6029258.
18. Zhang X, Guo Y, Cai Y, et al. Tongue image segmentation algorithm based on deep convolutional neural network and fully conditional random fields. J Beijing Univ Aeronaut Astronaut 2019; 45: 2364–2374.
19. Ronneberger O, Fischer P and Thomas B. UNet: convolutional networks for biomedical image segmentation. Switzerland: Springer International Publishing, 2015.
20. Zunair H and Hamza AB. Sharp UNet: depthwise convolutional network for biomedical image segmentation. Comput Biol Med 2021. doi: 10.48550/arXiv.2107.12461
21. Guan S, Khan AA, Sikdar S, et al. Fully dense UNet for 2d sparse photoacoustic tomography artifact removal. IEEE J Biomed Health Inform 2019; 24: 568–576.
22. Zhou Y, Huang W, Dong P, et al. D-UNet: a dimension-fusion U shape network for chronic stroke lesion segmentation. IEEE/ACM Trans Comput Biol Bioinf 2019; 18: 940–950.
23. Paluru N, Dayal A, Jenssen HB, et al. Anam-Net: anamorphic depth embedding-based lightweight CNN for segmentation of anomalies in COVID-19 chest CT images. IEEE Trans Neural Netw Learn Syst 2021; 32: 932–946.
24. Weng Y, Zhou T, Li Y, et al. NAS-UNet: neural architecture search for medical image segmentation. IEEE Access 2019; 7: 44247–44257.
25. He K, Zhang X, Ren S, et al. Deep residual learning for image recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), (2016), pp.770–778.
26. Peng L, Yang Y, Wang Z, et al. MRA-Net: improving VQA via multi-modal relation attention network. IEEE Trans Pattern Anal Mach Intell 2020; 44: 318–329.
27. Chen H and Shi Z. A spatial-temporal attention-based method and a new dataset for remote sensing image change detection. Remote Sens (Basel) 2020; 12. Article ID: 1662.
28. Tang X, Meng F, Zhang X, et al. Hyperspectral image classification based on 3-D octave convolution with spatial-spectral attention network. IEEE Trans Geosci Remote Sens 2020; 59: 2430–2447.
29. Hu J, Shen L and Sun G. Squeeze-and-excitation networks. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), (2019), pp.7132–7141.
30. Zhang Z and Sabuncu MR Generalized cross entropy loss for training deep neural networks with noisy labels. Adv Neural Inf Process Syst 2018; 31: 8778–8788.
31. Milletari F, Navab N and Ahmadi S. V-Net: fully convolutional neural networks for volumetric medical image segmentation. 2016 Fourth International Conference on 3D Vision. (2016), pp.565–571.
32. Dickson MC, Bosman AS and Malan KM. Hybridised loss functions for improved neural network generalisation. Pan-African Artificial Intelligence and Smart Systems Conference 2021. Windhoek, Namibia, (2022) 169–181.
33. Steno P, Alsadoon A, Prasad PWC, et al. A novel enhanced region proposal network and modified loss function: threat object detection in secure screening using deep learning. J Supercomput 2021; 77: 3840–3869.
34. Winkelmaier G and Parvin B. An enhanced loss function simplifies the deep learning model for characterizing the 3D organoid models. Bioinformatics 2021; 37: 3084–3085.
35. Su J, Liu Z, Zhang J, et al. DV-Net: accurate liver vessel segmentation via dense connection model with D-BCE loss function. Knowl Based Syst 2021; 232: 107471.
36. Zhang Y. A review of recent evaluation methods for image segmentation. In: International Symposium on Signal Processing & Its Applications 2001: 1: 148–151.
37. Qin X, Zhang Z, Huang C, et al. U-2-Net: going deeper with nested U-structure for salient object detection. Pattern Recognit 2020; 106. Article ID: 107404.
38. Roy AG, Navab N, Wachinger C, et al. Concurrent spatial and channel ‘squeeze & excitation’ in fully convolutional networks. In: Proceedings of the Medical Image Computing and Computer-assisted Intervention. (2018), pp.421–429.
39. Oktay O, Schlemper J, Folgoc L, et al. Attention U-Net: learning where to look for the pancreas. (2018). 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).
40. He K, Zhang X, Ren S, et al. Deep residual learning for image recognition. 2016 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), (2016), pp.770–778.
41. Chen LC, Papandreou G, Schroff F, et al. Rethinking Atrous Convolution for semantic image segmentation. 2017 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), (2017).
42. Tang H, Wang B, Zhou J, et al. DE-Net: dilated encoder network for automated tongue segmentation. 2020 25th International Conference on Pattern Recognition (ICPR). IEEE, 2021.
43. Hamed Mozaffari M and Lee W-S. Dilated convolutional neural network for tongue segmentation in real-time ultrasound video data. 2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM). IEEE, 2021.