mathematics

MDPI

Article

Tongue Segmentation and Color Classification Using Deep Convolutional Neural Networks †

Bo Yan [1,*], Sheng Zhang [1], Zijiang Yang [2], Hongyi Su [1] and Hong Zheng [1]

[1] School of Computer Science, Beijing Institute of Technology, Beijing 100081, China
[2] School of Information Technology, York University, Toronto, ON M3J 1P3, Canada
* Correspondence: yanbo@bit.edu.cn
† This paper is an extended version of the paper published in Arai, K. (eds) Advances in Information and Communication. FICC 2021. Advances in Intelligent Systems and Computing, vol 1364. Springer; pp. 649–662.

Abstract: Tongue color classification serves as important assistance for traditional Chinese medicine (TCM) doctors to make a precise diagnosis. This paper proposes a novel two-step framework based on deep learning to improve the performance of tongue color classification. First, a semantic-based CNN called SegTongue is applied to segment the tongues from the background. Based on DeepLabv3+, multiple atrous spatial pyramid pooling (ASPP) modules are added, and the number of iterations of fusions of low-level and high-level information is increased. After segmentation, various classical feature extraction networks are trained using softmax and center loss. The experiment results are evaluated using different measures, including overall accuracy, Kappa coefficient, individual sensitivity, etc. The results demonstrate that the proposed framework with SVM achieves up to 97.60% accuracy in the tongue image datasets.

Keywords: Convolutional Neural Networks (CNNs); semantic segmentation; feature extraction; classification; prediction

MSC: 68U01; 68U99

Citation: Yan, B.; Zhang, S.; Yang, Z.; Su, H.; Zheng, H. Tongue Segmentation and Color Classification Using Deep Convolutional Neural Networks. *Mathematics* **2022**, *10*, 4286. https://doi.org/10.3390/math10224286

Academic Editor: Danilo Costarelli

Received: 5 October 2022
Accepted: 14 November 2022
Published: 16 November 2022

1. Introduction

Tongue diagnosis is a very traditional and effective method which has been widely used for disease diagnosis in Traditional Chinese medicine (TCM) since more than 2000 years ago [1,2]. In TCM, it is believed that different regions of the tongue reflect the fitness of five corresponding organ systems: liver, lung, spleen, heart and kidney [1]. If a disease occurs in one of the organs of the body, an abnormality will occur in the corresponding area of the tongue, as shown in Figure 1 [3]. Since the tongue reflects the fitness of the body's corresponding organ system, it can help us to determine the body's energy pattern, the pain that one is experiencing, or the underlying cause of the disease.

The tongue information such as tongue coating, tongue shape and tongue color is linked to one's physical health in TCM. For example, in tongue color diagnosis, a light red tongue indicates a healthy condition, and a dark red tongue indicates a lack of blood and Qi (energy) [1]. Tongue color is divided into four categories: pale white, light red, red and dark red. The diagnosis of tongue color is more difficult than that of tongue coating and tongue shape. Because the feature space of different tongue color classes is similar, a formal diagnosis environment and rich clinical experience are especially necessary in the diagnosis.

Fortunately, with the development of intelligent medicine, the birth of the computer-aided tongue diagnosis system can not only help doctors to make more accurate and objective diagnosis, but also save a lot of time and improve the efficiency of diagnosis. In the past few decades, there have been many studies on the automation of tongue

diagnosis [4,5]. Most of the work focuses on segmenting tongue body from background, designing a feature extraction method [4], training a classifier based on the tongue dataset and giving the predicted disease type or treatment suggestions based on the classification results [5]. Both tongue segmentation and tongue feature representation methods can affect the diagnosis result.

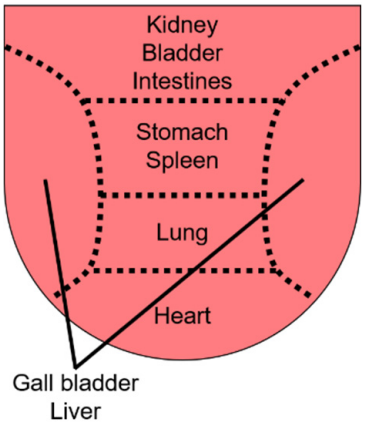

Figure 1. Areas of the tongue corresponding to different internal organs.

Tongue image segmentation is the foundation of tongue image diagnosis because the tongue image obtained by photographing includes not only tongue, but also teeth, lips, neck and face. This redundant image information interferes with the subsequent tongue image classification. In order to improve the accuracy of the subsequent classification, it is necessary to segment the tongue image first to remove the useless background information. Image segmentation can be formulated as a classification problem of pixels with semantic labels or partitioning of individual objects [6]. There are four basic traditional segmentation ways including thresholding [7], edge detection [8,9], graph theory [10,11] and active contour model [12,13] to solve the segmentation problem. These methods can achieve good segmentation results for some images acquired in a simple specific scene but behave poorly in processing complex and multi-scale images. Some methods are sensitive to light changes, some cannot segment tongue from lip successfully, and some methods have a running-speed problem during the processing phase. In recent years, a few studies have used CNN methods, but these methods often require image preprocessing and enhancement, which make the segmentation process more complex. Moreover, some methods usually leave the sawtooth area at the boundary of the tongue, which interferes with the classification of the tongue color. In addition, when the lip color is similar to the tongue, the segmentation difficulty will be greatly increased, and the accuracy of segmentation will be significantly reduced.

As mentioned above, the base of TCM diagnosis is abnormal tongue changes, including color, fur color, tongue thickness, cracks and tooth marks. From the imaging point of view, these anomalies are composed of miscellaneous features such as color distribution, geometric features and texture features. However, most researchers only focus on one of these characteristics. Because the features of different tongue color classifications are similar and tongue color diagnosis is more difficult than tongue coating and tongue shape, we will start with tongue color classification to study the feature extraction method of tongue color classification. Most tongue diagnosis works [14–19] adopt statistical methods and some traditional machine learning methods. However, the accuracy of these tongue color classification methods usually cannot meet the needs of actual diagnosis. The main reason is that the characteristics of most tongue datasets are complex and difficult to distinguish. Moreover, the tongue color features are very similar, which results in the overlapping of different tongue color features in color space. Additionally, these above-mentioned methods mostly use a pre-designed feature extractor to extract low-level features, which cannot

completely represent the characteristics of the tongue features, especially the fine-grained features such as tongue color. Most importantly, the collected tongue images usually contain non-tongue areas such as the face, neck and even clothing. These non-tongue areas are not only large in size, but also rich in color, often causing interference in the process of extracting tongue color features. Therefore, removing the useless background noise will undoubtedly help improve the accuracy of classification.

In order to overcome the limitations of the existing methods, we make a significant improvement and extension to our previous work [20] and use some efficient methods to remove the interference of unrelated factors and learn the features of different classes. First, we achieve automatic segmentation of the tongue from the background using semantic-based CNNs. To enhance the ability of the network to extract features of different scales, we add multiple atrous spatial pyramid pooling (ASPP) modules and a multi up-sampling process on the basis of the DeepLabV3+ network to enrich the bottom and the multi-scale information of the image. Next, we apply CNNs models to extract tongue color features and add an effective loss function called center loss [21] during the training phase to enhance the discriminative ability of the four tongue color features. This method enjoys time-efficiency and high accuracy without a large training dataset. In addition, with the combination of softmax loss and center loss, a robust classification model to extract high-level features of the tongue color is trained [22]. The contribution of center loss is presented in Section 3.

The main process of the proposed method is as follows: first, we segment the tongue body from the background using semantic-based CNNs in order to eliminate the unrelated features. Then, the high-dimensional features of tongue color are extracted by CNNs models with center loss to enhance the distinguishability of different tongue color features. In order to validate the performance of the proposed method, this paper uses several classical feature extraction methods combined with center loss for comparison. The experimental results show that the proposed method can improve the performance of the tongue color classification effectively.

The main contributions of this paper are as below:

(i) A dataset containing 2174 tongue images is constructed.

(ii) A novel network called SegTongue is proposed to segment the tongue from the background to eliminate the interference features of the background. The result demonstrates that the new network is able to extract the multi-scale features of the tongue more effectively through the quantitative and qualitative analysis.

(iii) The structure of the feature extraction model is constructed with center loss to train a robust classification model to extract high-level features of tongue colors. The results show that the accuracy of tongue image resolution is significantly improved after using the center loss function.

(iv) SegTongue is combined with a feature extraction network which is improved using the center loss function. The results show that this two-stage model has a significant improvement in tongue classification compared to a single general classification network.

The remaining content of this paper is organized as follows. Section 2 provides a review of the related works. Section 3 discusses the proposed methods. Section 4 presents the description of datasets and the experimental results. Finally, Section 5 concludes the paper.

2. Related Work

2.1. Tongue Image Segmentation

As mentioned earlier, most tongue segmentation methods still use traditional image processing methods, which can be approximately classified into four main categories: thresholding, edge detection, graph theory and active contour model. For the thresholding-based method, a tongue image is first divided into several sub-blocks, then the optimal threshold value of each sub-block is calculated by the iterative method, and the segmentation is carried out according to the threshold matrix formed by each local optimal threshold value [7,8]. The second is edge detection. Fu et al. [9] used radial edge detection on the

tongue image to get the initial contour, and then used Snake to get the contour of the tongue. Similarly, Zhang et al. [10] also applied the snake minimum energy function for edge convergence and obtained the ideal effect of tongue extraction. The other subcategory is based on graph theory. Wei et al. [11] proposed a tongue image segmentation method based on quadtree and GrabCut. This method first used quadtree decomposition to initially segment the collected tongue image, then used the color mean of similar regions to optimize the Gaussian mixture model parameters in the GrabCut algorithm to complete the tongue image segmentation. The last traditional method is called the active contour model. Guo et al. [13] proposed a two-stage K-means clustering method and combined it with the active contour model to segment the tongue image. Although these traditional methods have achieved good results in some data sets, they still have some shortcomings and limitations [23]. For example, some methods do not have good computing complexity, some are sensitive to changes in light, some fail to separate the tongue from the lips, and some even confuse the part of neck with the tongue. Therefore, it is still a big challenge to segment the tongue image using traditional methods.

Fortunately, with the development of deep learning, the deep convolutional neural network has been widely used in medical image processing, which also provides a new direction for tongue image segmentation [24–29]. However, due to the difficulty of the source data collection and high requirement of image segmentation for image annotation, few studies have applied deep learning technology to tongue image segmentation. Recently, several studies have applied deep learning techniques to tongue segmentation, demonstrating its superiority over traditional methods. Gao et al. [24] combined symmetry and edge detection with convolution neural network to perform tongue segmentation. Huang et al. [25] developed an improved U-Net to train and segment tongue images so that the tongue body can be extracted from images collected in an open environment using certain validated devices. Qu et al. [26] proposed an image quality evaluation method based on brightness statistics to determine whether the input image is preprocessed or not. After the image preprocessing, they used the Segnet to segment the tongue image. However, due to the preprocessing method of brightness statistics, the generalization ability of the whole network is reduced. Lin et al. [27] proposed an end-to-end trainable tongue image segmentation method using a deep convolutional neural network based on ResNet. They presented a method for the segmental tongue of the forward network without preprocessing. This method is not limited by the image brightness, and it can improve the poor generalization ability of the segmentation network which depends on preprocessing. However, the boundary of the tongue body processed by these methods is poor and often has a jagged shape. In addition, when the tongue size distribution in the data set varies greatly, the segmentation accuracy will be greatly reduced.

2.2. Tongue Image Classification

Tongue classification is a classic task, and with the development of deep learning, some studies have applied machine learning and deep learning methods to tongue recognition. For example, Hu et al. [30] designed a neural network framework and trained a prediction model which can construct herbal prescriptions using the real-world tongue images. Ma et al. [31] used a complexity perception method based on deep convolutional neural network and aimed to automatically classify patients into nine constitution types. Lu et al. [32] proposed a two-phase deep neural network for tongue diagnosis. Cao et al. [16] computed tongue statistics information based on Lab color space and used the XGBOOST classifier to predict tongue images. Ding et al. [18] extracted features based on the theory of local object appearance and shape and built a doublet for further use in SVM. Zhang et al. [33] used PCA to reduce the dimensions of the tongue data and optimized kernel parameters of SVM by genetic algorithm (GA). Li et al. [15] presented an approach analyzing tongue color based on spectra with a spectral angle mapper. Kamarudin et al. [17] proposed a two-stage tongue classification method based on a support vector machine (SVM) whose support vectors are reduced by a k-means clustering identifier. Zhou et al. [19] adopted

Content-Based Image Retrieval (CBIR) to extract the visual features of tongue images and used k-means as classifier. Mansour et al. [34] developed a classifier based on a deep neural network to determine the existence of diseases from tongue images captured by IoT devices. Li et al. [35] used facial landmark detection combined with ResNet34 to have the self-captured tongue photos segmented and then classified.

Although the above-mentioned methods have made great progress in the development of tongue image classification, the accuracy of tongue color classification is not high. None of these methods did tongue segmentation first to remove the useless background parts. Therefore, the noise, such as lips and faces, affects the accuracy of the subsequent tongue classification process. Furthermore, some of the above methods use pre-designed feature extractors to extract low-level features. Nevertheless, these features cannot fully represent the high-level features of the tongue image, and the tongue color features are also very complicated, which further reduces the accuracy of the tongue image recognition by the network. Therefore, there is an urgent need to come up with a new way to solve the problem left behind.

3. Methods

The proposed automatic tongue-diagnosis system contains two functions: tongue image segmentation and tongue image classification. For these two different tasks, we first propose a new semantic-based CNN called SegTongue to segment the tongue from the background to eliminate the interference features from the background. Then, we apply the center loss function to different feature extraction networks for tongue image classification. Finally, we combine the two networks to get an automatic tongue diagnosis system. The details of the proposed method are described in the following sections.

3.1. Atrous Convolution

Atrous convolution originated from signal processing which was developed for the efficient computation of the wavelet transform [36]. In traditional CNNs, the repeated max-pooling layer after convolution layer and striding operations reduces the spatial resolution in feature maps significantly. Therefore, a lot of detailed information is ignored during the feature extraction stage using the traditional CNNs. In addition, the repeated convolution-pooling operation adds a large number of parameters and reduces the computational efficiency. Atrous convolution provides us a way to expand the receptive field without increasing the number of convolution pooling. Compared to the traditional convolution methods, it can improve computational efficiency and retain the detailed information. Mathematically, the atrous convolution can be expressed as below:

$$f(h,w) = \sum_{i=1}^{H} \sum_{j=1}^{W} g(h + r \times i, w + r \times j) k(i,j) \quad (1)$$

where $f(h,w)$ is the output of the input $g(h,w)$ after the convolution kernel k is convolved at location (h,w). H and W are the length and width of the convolution kernel k and and r is the dilation rate. When the convolution kernel is square and the size is $k \times k$, atrous convolution enlarges the receptive domain from $k \times k$ to $k + (k-1)(r-1)$. As shown in Figure 2, a 3×3-size atrous convolution kernel has the same receptive field as a standard 3×3 convolution kernel when r is equal to 1. However, when r is equal to 2, it has a 5×5 receptive field.

The tongue boundary is a global region composed of many small-sized local regions. In the inner region of the tongue boundary, the gray-scale values of the pixels change continuously. In the outside region of the tongue boundary (background region), the gray-scale values of the pixels change irregularly. Atrous convolution extracts the subtle features of tongue boundary without using a max-pooling layer, which avoids features loss. So, the dilated convolution can help us better extract tongue features.

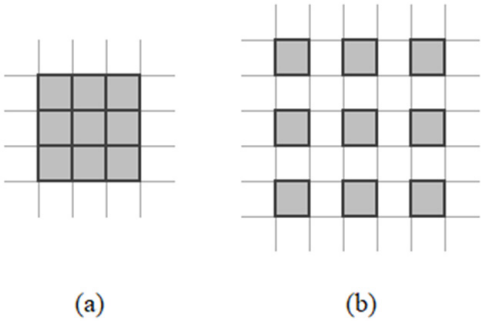

Figure 2. A schematic diagram of the atrous convolution at different expansion rates. (**a**) $r = 1$; (**b**) $r = 2$.

3.2. Multi-Scale Feature Extraction Methods

In recent years, CNNs have been widely used in many domains such as speech recognition, object recognition, object detection and many other domains. CNNs take advantage of the use of many layers which have different functions to extract high-level features automatically. These CNNs-based models including AlexNet [37], GoogLeNet [38], Vgg [39], ResNet [40], DenseNet [41] have achieved good performance in ImageNet Large Scale Visual Recognition Competition (ILSVRC). However, the repetition of the continuous maximum pooling layer followed by a down-sampling layer in these DCNN models greatly reduces the spatial resolution, which causes the loss of a lot of detailed information and makes it more difficult to process the multi-scale images.

Recently, it has been proven that we can get a better performance by using pyramid representations for multi-scale images. Spatial pyramid pooling (SPP) was first introduced in CNNs by He et al. [42]. The network structure using SPP is capable of producing a fixed-size representation without considering the size or scale of the input image. It is robust to the change of the object shape. DeepLabV2 [43] replaces the traditional CNNs with parallel atrous convolutional layers and proposes a scheme called atrous spatial pyramid pooling (ASPP), which uses different sampling rates for parallel branches as shown in Figure 3. This is equivalent to processing original images with scale-different filters that have multiple effective fields of view, which can capture image context at multiple scales.

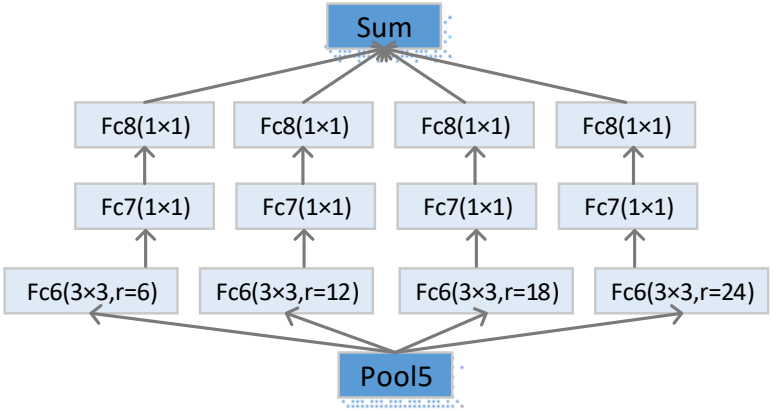

Figure 3. Illustration of ASPP with different atrous rates in each brunch.

Atrous convolution provides different sizes of the field of view using different sampling rates and ASPP captures the local features of the tongue boundary in a small field of view. These two operations can find the optimal trade-off between accurate localization and context assimilation. Hence, the combination of the two operations can recognize the tongue boundary precisely and segment tongues of different sizes robustly.

3.3. Proposed Methodology

Figure 4 shows the proposed framework to predict tongue images. The overall procedure can be divided into two parts: tongue segmentation and tongue classification. The input of the framework is a raw tongue image, then the tongue boundary is segmented using trained semantic-based models. After that, we generate a new tongue image excluding the background area using mask operation. Next, we input the segmented tongue image to the classification model and obtain its feature representation. Eventually, we get the tongue color labels by the classifier. The algorithms will be introduced in the rest of this section in details.

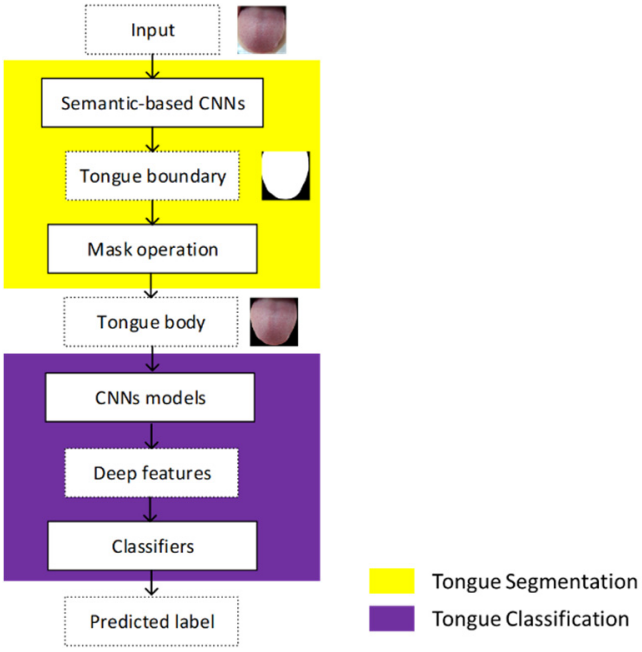

Figure 4. The overall framework of the proposed algorithm.

3.3.1. Tongue Segmentation

The traditional convolutional base of the DeepLab-type network is often too deep, causing the loss of some low-level details. Inspired by the good performance of the U-net in medical image processing, we decided to increase the number of iterations of fusions of low-level information and high-level information on the basis of the DeepLabV3+ original network. Furthermore, we propose to add more ASPP modules to capture multiscale features at different depth levels before each fusion process. Through these operations, we can improve the network's ability to capture low-level features and multi-scale features and obtain a higher segmentation accuracy.

Now, we will show the overall architecture of the new CNNs based on semantics named SegTongue. As shown in Figure 5, the basic convolutional model uses the residual neural network Resnet101, which can be decomposed into five parts as conv1, conv2, conv3, conv4 and conv5. Except conv1, all the remaining parts (conv2, conv3, conv4 and conv5) are simple repetitions of the three convolutional layers where the kernel size is $[1 \times 1, 3 \times 3, 1 \times 1]$, and the number of repetitions is [3, 4, 23, 3]. Among the last 4 parts, the atrous convolution rate and step size are different, which are [1, 1, 1, 2] and [1, 2, 2, 1], respectively. After the image passes through each convolution part, its current output will be saved and fed into the ASPP module to extract the multi-scale information of the current layer. When the image is convolved through the entire Resnet101, it will be merged with the ASPP processed output and then up-sampled in turn until it is convolved into one channel through a LastConv layer to obtain the final mask result.

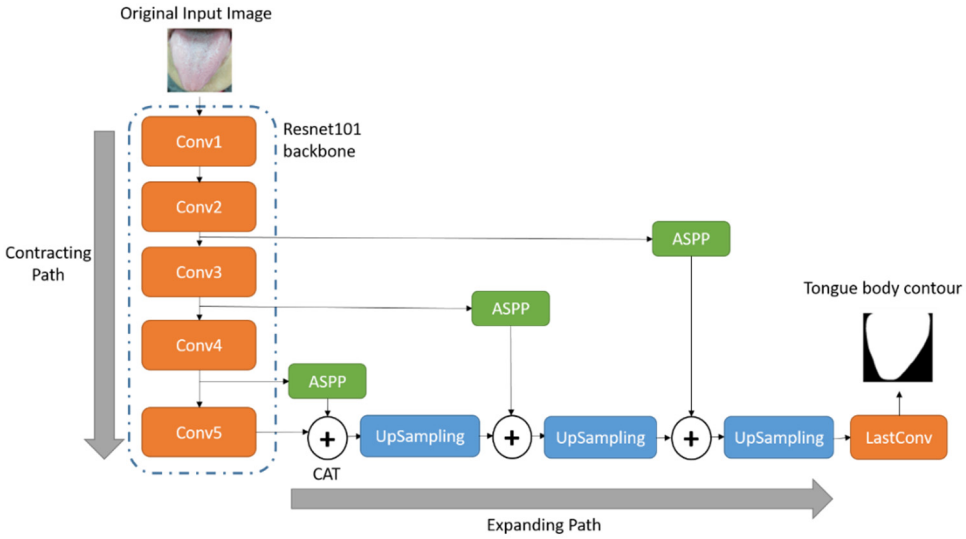

Figure 5. Overall architecture of semantic-based CNNs (SegTongue).

The training set contains the original images and ground truth images. The ground truth images are labeled by doctors. The tongue area is marked as the same RGB value, and non-tongue areas are kept original. During the training phase, we fine tune the model weights of pre-trained Resnet-101 and apply this to the segmentation task as described in [44]. The classifier in Resnet-101 is replaced with a classifier having two targets (including tongue and background) as the number of segmentation classes.

3.3.2. Tongue Classification

In this paper, we use AlexNet, Vgg-16, ResNet-18 and DenseNet-101 to extract tongue-color features. We mainly modify fully connected layers of the four models. First, the number of neurons of fully connected layers is set to 1024 based on the scale of our dataset. Then, the output of the last fully connected layer is fed to a four-way softmax producing a distribution over the four tongue color labels. Meanwhile, the output of the last fully connected layer is saved as files which are later separately read by SVM or RF to train classifiers. The format of the input images is specified as (224, 224, 3).

We train four models using standard SGD [37] with a momentum of 0.9 and weight decay of 0.0005. The update rule for the weight parameters of each batch is provided below:

$$w_{i+1} = w_i + v_{i+1} \tag{2}$$

$$v_i = 0.9 \cdot v_i - 0.005 \cdot \varepsilon \cdot w_i - \varepsilon \cdot \frac{\partial L}{\partial w} \tag{3}$$

where i is iteration index, v is momentum, ε is learning rate, and L is the loss value calculated by softmax loss function for the ith batch data.

However, in our tongue color classification task, a major challenge is that the deeply learned features of different classes are not easy to separate because the features are too similar to distinguish. In order to overcome this challenge, the center loss is used to extract the features of the same tongue-color class to their centers and obtain the highly discriminative features for tongue-color classification. The center loss was first introduced to enhance the discrimination power of the deep-learned features in face recognition. The softmax loss makes the features of different classes apart, and the center loss makes the features of the same class move to its center. With the combination of softmax loss and center-loss function, we can measure the performance of the trained model with the inter-class loss and the intra-class loss. The formulation of softmax loss and center loss are as below:

$$L_S = -\sum_{i=1}^{m} log \frac{e^{W_{y_i}^T x_i + b_{y_i}}}{\sum_{j=1}^{n} e^{W_j^T x_j + b_j}} \tag{4}$$

where x_i denotes the ith feature with the dimension of d, W_j denotes the jth column of W in the last fully connected layer with the dimension of $d * n$, b is the bias, m is the size of the mini-batch, and n is the number of classes.

$$L_C = \frac{1}{2}\sum_{i=1}^{m} ||x_i - c_{y_i}||_2^2 \tag{5}$$

where c_{y_i} denotes the features' center of the y_i^{th} class. The update rule of c_{y_i} is computed as:

$$\Delta c_j = \frac{\sum_{i=1}^{m} \delta \cdot (c_j - x_i)}{1 + \sum_{i=1}^{m} \delta} \tag{6}$$

where $\delta = 1$ if $y_i = j$, and $\delta = 0$ if not.

The equation of combination of softmax loss and center loss is as below:

$$L = L_s + \lambda L_c \tag{7}$$

It is trainable and can be optimized by SGD. A hyper parameter λ is brought in to balance two loss functions. Take the combination of two loss functions using AlexNet as an example. During the training phase, the center loss value is calculated by the second fully connected layer and the labels. The softmax loss value is calculated by the last fully-connected layer and the labels. Then, we sum the two loss values according to the balance parameter λ.

Now, the detailed procedures of the training tongue color feature extraction model is shown in Figure 6. First, the tongue body of the images is acquired by the well-trained semantic-based CNNs model as described in Section 3.3.1. Then, the dataset is divided into the training set and testing set using ten-fold cross-validation. In the training phase, a mini batch of the training set is used as the input of CNNs during forward propagation. After that, the loss value of softmax and center loss are calculated using the output of forward propagation and the ground truth. We update the parameters of CNNs using back propagation and standard SGD.

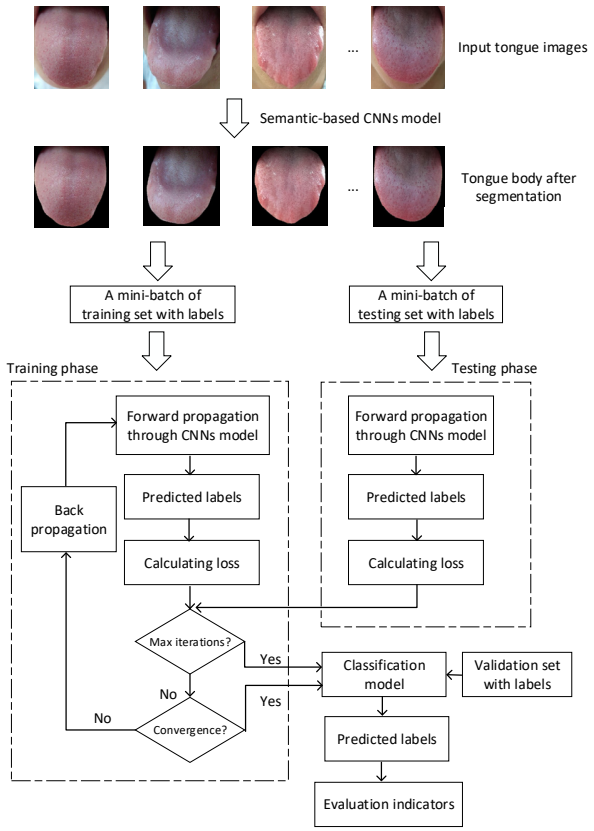

Figure 6. Procedures of training feature extraction models based on CNNs.

During the training phase, several models are saved by snapshots. Nevertheless, not all of them have a good performance. The best one is chosen when the accuracy and the loss value of the training set and testing set are close and do not change drastically with the increasing number of iterations. The model can learn the general natural features of all tongue images, not the own characteristics of some tongue images, thus avoid overfitting and underfitting.

4. Experiment Result and Discussion

In this section, we use the method described in Section 3 to perform the experiment and compare the result with some popular algorithms. Firstly, we introduce the dataset details in Section 4.1. Then, the result of the tongue segmentation is demonstrated in Section 4.2 and the result of the tongue classification is demonstrated in Section 4.3.

4.1. Dataset

The dataset is collected from a Chinese medicine hospital using a tongue acquisition device. We collected each patient's tongue image to construct the tongue image dataset. It took several months to collect about 2500 photos. Then, we deleted blurry and far-away images which do not meet the medical standards. Eventually, we obtained 2174 photos. The significance of our dataset is that it contains a large base of samples based on the actual clinical cases. The collection process is done in a random manner without manual intervention. The color classification label of each picture is labeled by the doctor using a consistent standard. In our research, tongue color is divided into four types, namely, pale white, light red, red and dark red. The number of each type is 816, 720, 326 and 312, respectively. In addition, considering that tongue segmentation requires more accurate pixel-level labels, we select 500 images from the dataset for pixel-level annotation to construct a tongue-segmentation dataset.

It is obvious that the number of red and dark red type is relatively small compared to pale white and light red type, resulting in the unbalanced number of samples per type. In order to address this problem, which is normal in the field of deep learning [45], oversampling is widely used because of its robustness in most analyzed scenarios [46]. Thus, data augmentation is done for red and dark red types. Every image in these two categories is rotated by a random angle between -30° to 30° with a probability of 0.5. These images are also left–right flipped with a probability of 0.5. After the data augmentation, we obtained 658 red tongue images and 632 dark red tongue images. The balance between each type helps the model to have better generalization capability. In our experiment, we use three datasets according to the number of tongue images in each type, namely, Tongue-2400, Tongue-2040, and Tongue-1560. Dataset Tongue-2400 contains 600 pale white tongue images, 600 light red tongue images, 600 red tongue images and 600 dark red tongue images, adding up to 2400 images in total. Similarly, dataset Tongue-2040 contains 510 images in each of four types, and Tongue-1560 contains 390 images in each type.

4.2. Experiment Result of Segmentation

4.2.1. Evaluation Criterion

The performance of the tongue segmentation is measured by the pixel accuracy (PA), Dice coefficient (Dice) and mean intersection over union score (mIoU). In our segmentation work, 0 represents the background and 1 represents tongue area. The calculation method can be described as the following equations:

$$PA = \frac{\sum_{i=0}^{k} p_{ii}}{\sum_{i=0}^{k} \sum_{j=0}^{k} p_{ij}} \tag{8}$$

$$Dice = \frac{2|T_g \cap T_p|}{|T_g| + |T_p|} \tag{9}$$

$$mIOU = \frac{1}{2}\left(\frac{|T_g \cap T_p|}{|T_g \cup T_p|} + \frac{|B_g \cap B_p|}{|B_g \cup B_p|}\right) \quad (10)$$

The parameter k represents the maximum number of the labels. Since there are only two classes (0 and 1 represent the background and tongue body, respectively), k is set to 1 here. The parameter p_{ij} represents the number of pixels whose label is i but predicted to be j. The parameter p_{ii} represents the number of correctly predicted pixels. The parameter T_p and B_p represent the tongue region and non-tongue region (background) of the prediction of the model; T_g and B_g represent the tongue region and non-tongue region (background) of the ground truth.

4.2.2. Implementation Details

The model is implemented using PyTorch framework and trained on Ubuntu 16.04 OS with 2.2 GHz Intel Xeon E5-2660, 32 GB RAM and a NVIDIA Tesla P100 graphic card with 16 GB memory. We construct four branches in the ASPP module and the atrous rate of each branch is set to 6, 12, 18 and 24, respectively, which can capture the tongue boundary in different scales as shown in Figure 3. If the batch size is too small, the Batch Norm layer is incapable of estimating the mean and variance of the whole dataset accurately, which causes the model to diverge. If the batch size is too large, the model falls into local convergence. Thus, we decided to set the batch size to 8 for compromise. In addition, the Adam algorithm is utilized to optimize with learning rate as 0.001 and momentum 0.9 according to the recommended value by Kingma et al. [47]. In the experiment, ten-fold cross-validation is utilized for each model to evaluate the performance and the pre-trained Resnet-101 network is applied in our segmentation task as described in Section 3.3.1.

4.2.3. Experimental Results and Analysis

To evaluate our model, we compare our method with the four recent deep learning methods: FCN, SegNet, DeepLabV3+ and U-net. It should be noted that FCN, U-Net and DeepLabV3+ are widely used for image segmentation in medical imaging or natural scenes, and they are the baseline for the comparison in this paper.

To quantitatively observe the effectiveness of the proposed model, we select three metrics, including pixel accuracy (PA), mean intersection over union (mIoU) and Dice coefficient (Dice) for the performance measurement. The experimental results are provided in Table 1 and Figure 7. The experimental results show that the proposed method provides a better result compared to the baseline methods. For example, as shown in Table 1, in terms of PA, the SegTongue achieves gains of 8.6% compared to FCN, 4.0% compared to U-Net, 6.7% compared to SegNet and 3.2% compared to DeepLabV3+, respectively. Furthermore, SegTongue is also superior to other baseline models based on the other two metrics. In addition, as can be seen from Figure 7, the SegTongue exhibits higher stability compared to Segnet and U-net. A possible reason for this fact is that the predictions by those baseline methods are not able to extract the boundary and the multi-scale features of the tongue well. Thus, these models have outliers when dealing with boundaries. On the contrary, the SegTongue model can effectively deal with multi-scale and detailed features.

In order to observe the improvement of the proposed model more intuitively, we conduct a qualitative analysis of the experimental results. We selected several pictures containing lips, teeth and tongue fur with abnormal colors as experimental data. Segnet, U-net, DeepLabV3+ are used as the baseline for comparison. Some examples of experimental results are shown in Figure 8. In Figure 8a,b, we can see that the segmentation of the tongue is severely affected by the lips and teeth, especially the lips, whose color is very close to the tongue. From the experimental results, the segmentation of the Segnet network as the benchmark is the worst. The U-net and DeepLabV3+ are slightly better, although they are still not good. Our proposed SegTongue model avoids the interference of lips and teeth and achieves a better tongue segmentation. In Figure 8c, the tongue is covered with a layer of green tongue fur, resulting in an abnormal color. The experimental results show that Segnet and DeepLabV3+ produce holes or incomplete images during processing. The reason may

be that the abnormal tongue coating is identified as the background mistakenly. In contrast, our SegTongue segments the tongue well without being disturbed by abnormal tongue coating. In order to observe the effect of our model on multi-scale images, we chose to perform experiments on tongue images at different scales. The results show that although the tongues vary in size, our proposed model divides them well. All these improvements prove the effectiveness of detail and multi-scale feature extraction in SegTongue. It achieves higher accuracy and stability than the baseline.

Table 1. Segmentation performance of different methods.

Method	PA	Dice	mIoU
FCN	89.07%	88.12%	87.44%
U-net	93.68%	93.23%	92.61%
Segnet	90.95%	90.15%	89.54%
DeepLabV3+	94.53%	94.02%	93.27%
SegTongue	**97.61%**	**97.12%**	**96.32%**

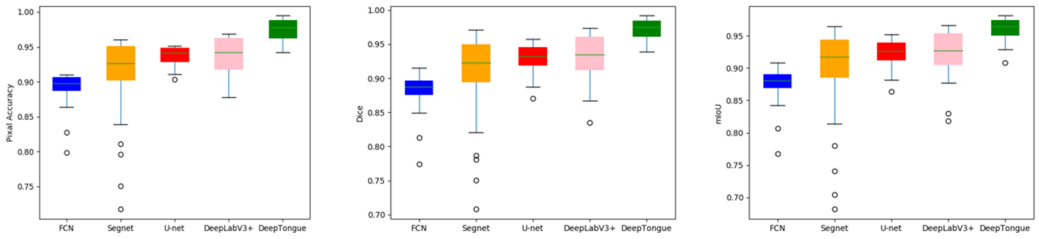

Figure 7. Comparison box plot.

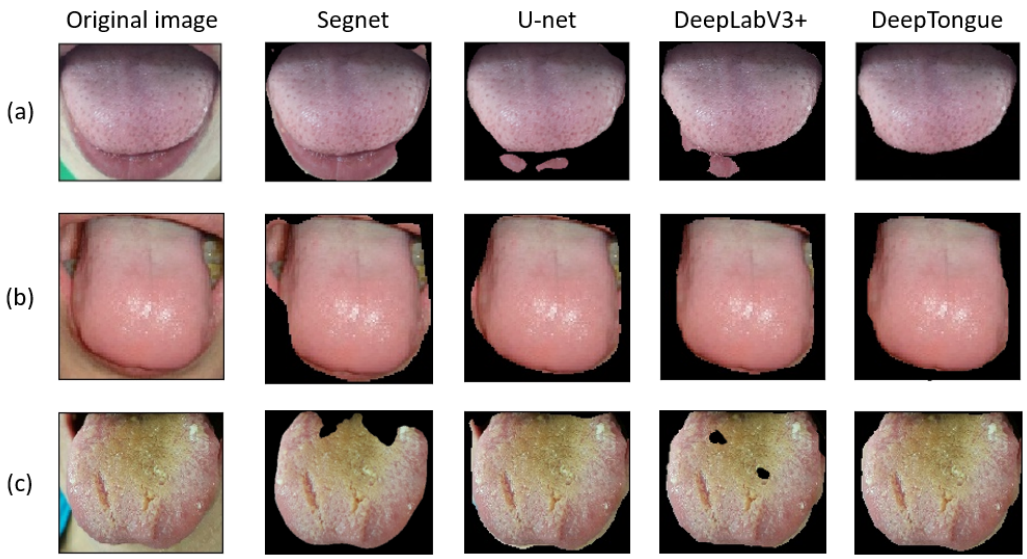

Figure 8. Visual results of segmentation for tongue images. (**a**) the segmentation of the tongue which is severely affected by the lips, (**b**) the segmentation of the tongue which is severely affected by the lips and teeth and (**c**) the tongue is covered with a layer of green tongue fur, resulting in an abnormal color.

4.3. Experiment Result of Classification

4.3.1. Evaluation Criterion

To evaluate the performance of the proposed method on tongue image classification, we adopt some effective evaluation indicators such as overall accuracy (OA), individual

sensitivity (Sens) and Cohen's Kappa to observe the classification results. All three metrics are defined as below:

$$OA = \frac{\sum_{i=0}^{k} x_{ii}}{N} \times 100\% \tag{11}$$

$$Sens = \frac{\sum_{i=0}^{k} x_{ii}}{\sum_{i=0}^{k} x_{gi}} \times 100\% \tag{12}$$

$$OE = \frac{\sum_{i=0}^{k} (x_{pi} \cdot x_{gi})}{N^2} \times 100\% \tag{13}$$

$$Kappa = \frac{OA - OE}{1 - OE} \tag{14}$$

The parameter k represents the maximum number of the labels. Since there are four classes labeled from 0 to 3, k is set to 3 here. x_{ii} is the number of correctly predicted images in class i. x_{gi} represents the number of class i in ground truth and x_{pi} represents the number of models predicted by the model as class i. N is the total number of the datasets.

4.3.2. Implementation Details

In order to observe the degree of improvement of tongue image classification accuracy by pre-segmentation and center loss function, we selected four commonly used feature extraction networks including AlexNet, Vgg-16, Resnet-18 and Desnet-101 as our benchmark network. All network models are implemented with the PyTorch framework. The hardware environment of the experiment is the same as the above segmentation experiment. If the dataset for tongue classification is much larger than the segmentation dataset, setting a too-small batch size will seriously affect the training efficiency. Thus, we decided to increase the batch size to 32. Four models are trained using standard SGD. Learning rate and momentum are set the same as the segmentation experiment. In the experiment, we use ten-fold cross-validation for each model and the pre-trained network is used for each model.

4.3.3. Experimental Results and Analysis

In this part, we first compare the results of different feature extraction networks including AlexNet, Vgg-16, ResNet-18 and DenseNet-101 under different values of parameters, and then discuss the contribution of segmentation and center loss function to the classification result. In addition, we compare the effects of different classifiers including Softmax, SVM and RF. Finally, we analyze the individual sensitivity of different methods under the best OA and visualize the learned features.

(1) Hyper parameter λ

Considering that there is a variable λ in the joint loss function, we first observe the effect of λ on overall accuracy. Figures 9–11 show the average overall accuracy of AlexNet, Vgg-16, ResNet-18 and DenseNet-101 with a hyper parameter λ (0.0001 to 1.0) controlling the weight between softmax loss and center loss. In addition, the results under softmax classifier, SVM classifier and RF classifier are also shown in Figures 9–11, respectively.

From the curves in Figures 9–11, we can see that Vgg-16 obtains the best average overall accuracy in these three datasets under different λ values. This indicates that Vgg-16 can extract high-level tongue color features efficiently. The ResNet's overall performance ranks second, and it achieves the second highest accuracy on Tongue-2040 and Tongue-2400 datasets. The results of DenseNet and AlexNet are similar, and their overall performance are worse than the ResNet and Vgg-16.

From the comparison, we can also conclude that the hyper parameter λ values are essential to the experiment result. If λ is small enough, the center loss makes little contribution to loss value. If λ is large enough, we will get a large loss value which will lead to gradient explosion. When λ is bigger than 1.0, gradient explosion occurs which results in a stop for training.

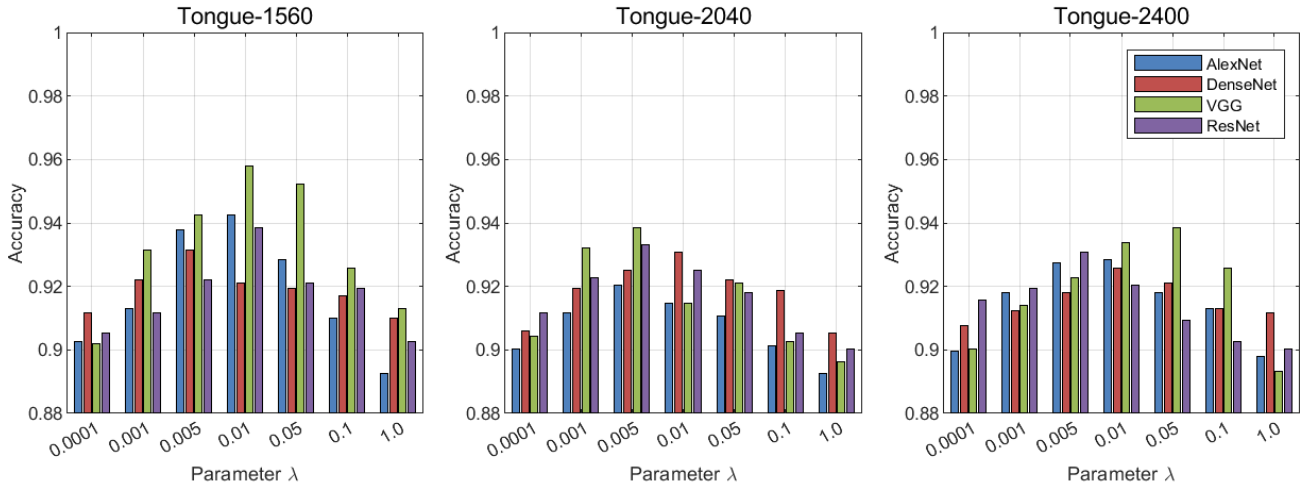

Figure 9. Comparison of mean overall accuracy under different values (Softmax classifier).

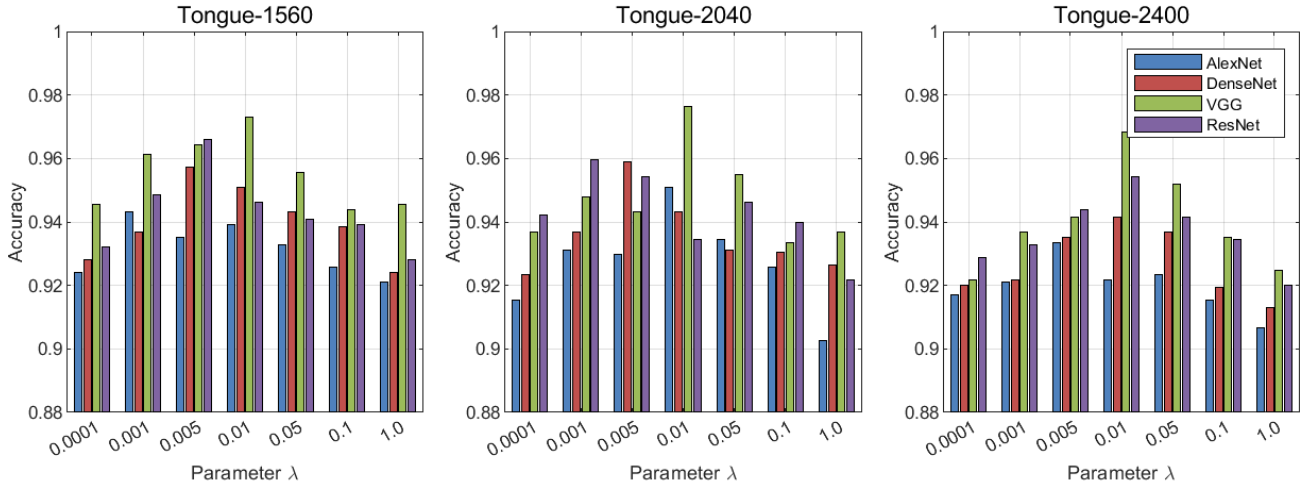

Figure 10. Comparison of mean overall accuracy under different values (SVM classifier).

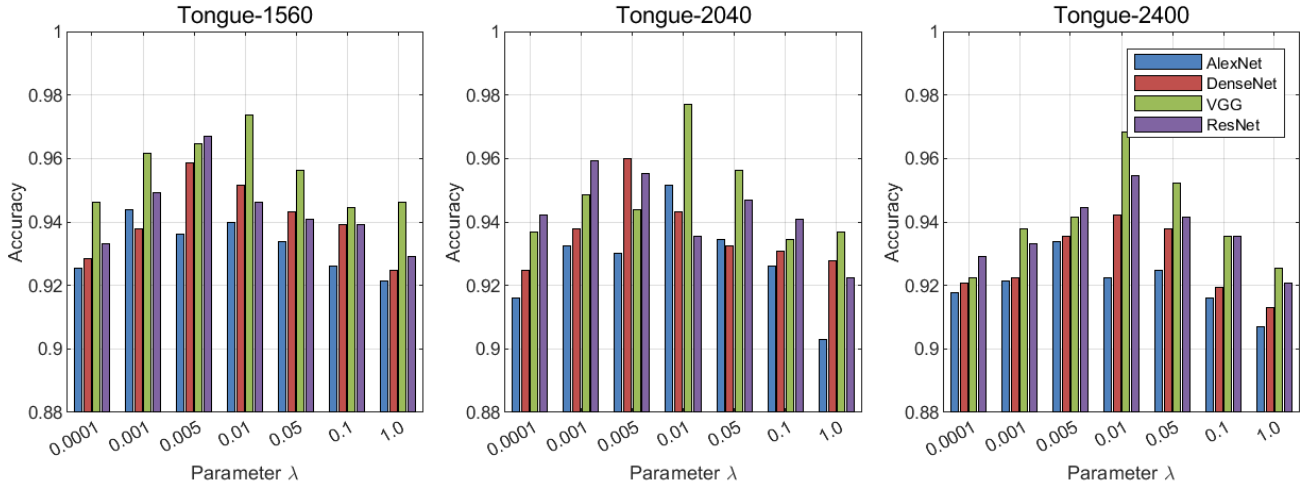

Figure 11. Comparison of mean overall accuracy under different values (RF classifier).

(2) Overall Accuracy

For further comparison, the best average overall accuracy of each feature extraction CNNs model is chosen to evaluate the performance. Table 2 compares the best average overall accuracy and the corresponding Kappa value of the original feature extraction models with those of the improved feature extraction models using segmentation and center loss under different base classifiers. Furthermore, the best result of the improved feature extraction models in Table 2 is obtained based on the optimal hyper parameter λ values in the center loss.

The result from Table 2 shows that after using our method, the accuracy of the model has been improved by about 3 to 5 percentage points, which indicates that our method is really effective. Besides, we also find that after using our method, the Vgg-16 achieves the best classification results on all three datasets, which are 96.83%, 97.60% and 97.31%, respectively. Compared to the original method, it has increased by 5.95%, 5.66% and 5.71%, respectively.

Table 2. Best mean Overall Accuracy comparison of different methods.

Dataset	Method	SVM Classifier		RF Classifier		Softmax Classifier		Max
		OA(%)	Kappa	OA(%)	Kappa	OA(%)	Kappa	
Tongue-2400	AlexNet	90.53	0.8745	90.54	0.8739	88.25	0.8567	90.54
	A + SSN + CL	92.38	0.9117	93.08	0.9211	92.00	0.9067	94.08
	DenseNet	90.79	0.8902	88.50	0.8867	88.50	0.8867	91.79
	D + SSN + CL	94.17	0.9222	93.00	0.9067	92.75	0.9033	94.17
	Vgg-16	90.88	0.8917	89.04	0.8672	88.75	0.8633	91.88
	V + SSN + CL	96.83	0.9578	95.67	0.9556	94.00	0.9200	96.86
	ResNet-18	91.79	0.9039	88.79	0.8772	88.25	0.8700	92.79
	R + SSN + CL	95.42	0.9389	93.96	0.9328	93.25	0.9100	95.42
Tongue-2040	AlexNet	91.40	0.8987	91.11	0.8948	88.11	0.8549	92.40
	A + SSN + CL	95.05	0.9340	94.56	0.9275	92.06	0.8941	95.05
	DenseNet	91.47	0.8995	89.38	0.8987	88.88	0.8784	92.07
	D + SSN + CL	95.87	0.9450	94.54	0.9272	93.00	0.9067	95.87
	Vgg-16	91.94	0.9059	91.11	0.8948	88.59	0.8745	92.94
	V + SSN + CL	97.60	0.9680	96.50	0.9667	93.82	0.9176	97.60
	ResNet-18	93.85	0.9314	91.24	0.9098	88.29	0.8706	94.85
	R + SSN + CL	95.87	0.9450	94.62	0.9417	93.25	0.9100	95.87
Tongue-1560	AlexNet	91.03	0.8834	91.73	0.8897	89.62	0.8615	91.73
	A + SSN + CL	94.35	0.9248	94.42	0.9256	94.23	0.9231	94.42
	DenseNet	91.51	0.9052	89.78	0.9045	89.46	0.8861	92.91
	D + SSN + CL	95.77	0.9410	94.26	0.9368	93.08	0.9077	95.77
	Vgg-16	91.60	0.8880	89.47	0.8863	89.62	0.8615	91.60
	V + SSN + CL	97.31	0.9641	95.05	0.9607	94.77	0.9436	97.31
	ResNet-18	91.26	0.9034	89.76	0.9034	88.38	0.8718	92.76
	R + SSN + CL	96.60	0.9547	94.96	0.9462	93.85	0.9179	96.60

Next, we will discuss the effect of different classifiers on the classification results. The best OA values on three datasets are 96.83%, 97.60%, 97.31% when the SVM classifier is used. Compared to the RF classifier, it is improved by 1.16%, 1.10% and 2.26%, respectively. Compared to the Softmax classifier, it has increased by 2.83%, 3.78% and 2.54%, respectively. It can be seen that the SVM classifier has achieved better results than the RF classifier and Softmax classifier. This phenomenon may be caused by the characteristics of the SVM classifier. This classifier has a strong anti-noise ability and can use some negative information carried by some noise to help identify. Evidently, our method has better performance and the performance of segmentation and center loss is relatively stable under different classifiers.

(3) Kappa Coefficient

The Kappa coefficient is a reliable statistical indicator used for reliability testing. It is often used to detect the consistency of several judgments. Kappa values vary from −1 to 1, with −1 representing that the judgments are completely inconsistent, and 1 representing that the judgments are completely consistent. However, the kappa values less than 0 are unlikely in practice. In general, Cohen [48] suggested that the Kappa value can be interpreted as follows: above 0.90 means the agreement is almost perfect, 0.80–0.90 is strong, 0.60–0.79 is moderate, 0.40–0.59 is weak agreement and less than 0.40 is almost no agreement. In healthcare research, many research papers recommend the 80% agreement rate as the minimum acceptable agreement.

The result from Table 2 shows that Kappa values keep consistent with OA. With our proposed method, the Kappa value has been improved on all models and classifiers. It proves that the proposed method is reliable as it increases the agreement to the highest confidence interval of kappa.

(4) Individual Sensitivity

The individual sensitivity of different methods under the best OA is provided in Table 3. As can be seen from the table, the Sens of all classes has been improved after using our recommended method, which shows that our method is effective. In addition, we also find an interesting phenomenon that no matter what method and classifier are used, models are able to deal with class 0, class 2 and class 3 better. This may be due to the fact that the tongue images with the label 1 are light red, representing a healthy tongue. Therefore, the class with label 1 has the largest number of samples in the total datasets which makes features widely distributed in feature space and their features may intersect with other classification features, especially the pale white and red class. For this reason, the classification effect of the model on class 1 may be a little worse.

Table 3. Sensitivity comparison of different methods under best mean overall accuracy.

Dataset	Method	SVM Classifier				RF Classifier				Softmax Classifier			
		Sens (%)				Sens (%)				Sens (%)			
		0	1	2	3	0	1	2	3	0	1	2	3
Tongue-2400	AlexNet	91.25	90.32	92.32	92.14	91.50	90.25	92.13	90.98	89.25	88.98	89.96	89.45
	A + SSN + CL	93.38	91.23	94.23	93.98	94.08	92.30	94.25	94.09	92.67	90.58	93.25	92.36
	DenseNet	92.17	90.28	92.15	93.09	91.50	89.59	91.23	92.45	91.50	90.56	92.54	91.35
	D + SSN + CL	94.17	92.15	93.48	94.89	93.00	91.25	92.56	93.96	92.69	92.03	92.98	92.21
	Vgg-16	91.88	91.05	92.36	92.12	90.04	88.45	89.17	90.69	89.25	88.54	89.30	89.14
	V + SSN + CL	96.83	95.36	96.41	96.98	96.67	95.36	96.18	96.98	93.99	92.54	94.32	94.82
	ResNet-18	92.79	90.56	91.87	93.65	90.79	89.21	90.68	91.54	92.00	91.02	92.36	92.47
	R + SSN + CL	95.42	93.57	96.35	96.45	94.96	92.65	93.54	93.47	93.28	91.52	92.47	92.58
Tongue-2040	AlexNet	92.40	91.54	92.74	92.68	92.11	92.01	93.47	92.73	89.12	88.54	91.58	90.75
	A + SSN + CL	95.05	93.47	94.12	96.36	94.56	93.57	95.18	94.85	92.08	91.58	90.54	93.58
	DenseNet	92.47	91.05	92.85	92.83	92.08	91.58	93.47	92.36	90.88	89.25	91.58	92.84
	D + SSN + CL	95.88	94.35	95.98	96.02	94.54	93.69	94.58	95.12	91.67	89.21	92.50	91.58
	Vgg-16	92.94	91.29	92.14	93.05	92.02	91.58	93.88	94.25	90.59	90.25	91.54	90.87
	V + SSN + CL	97.60	95.20	96.34	97.18	97.50	96.35	98.25	97.14	93.79	90.25	94.58	94.85
	ResNet-18	94.85	92.68	94.18	94.78	93.24	93.02	92.48	93.69	90.29	88.95	91.47	92.58
	R + SSN + CL	95.89	94.57	96.35	96.85	95.63	94.27	94.50	95.96	93.24	91.58	92.64	92.90
Tongue-1560	AlexNet	91.03	90.25	91.47	91.08	91.73	90.45	92.14	91.81	89.62	88.21	89.64	90.53
	A + SSN + CL	94.36	93.56	94.20	93.05	94.42	93.12	94.82	94.62	94.63	93.58	93.47	94.18
	DenseNet	92.93	91.05	94.84	93.67	92.78	91.58	92.69	93.14	92.46	90.51	92.82	92.64
	D + SSN + CL	95.58	94.05	96.77	96.14	95.26	94.25	94.58	95.84	93.56	92.84	93.64	94.76
	Vgg-16	91.60	92.14	91.30	91.25	91.48	90.54	91.87	92.54	89.62	88.65	91.54	92.47
	V + SSN + CL	97.31	95.36	97.36	96.48	97.05	96.48	98.25	98.02	95.93	93.47	94.14	94.61
	ResNet-18	92.76	90.14	93.91	92.86	92.76	91.25	92.34	92.87	90.38	89.08	90.07	91.25
	R + SSN + CL	96.60	95.35	94.56	96.84	95.96	92.15	96.54	94.15	93.99	92.84	93.14	92.24

(5) The change of loss

In order to observe the effect of the center loss function in the training process, we visualized the change of loss in the training process in Figure 12. Figure 12a shows the changes of loss value when the center loss is not used, and the Figure 12b shows the training process with center loss. As can be seen from the figure, when the center loss is not used, the loss on the validation set is much larger than that on the training set. When using center loss, the loss on the validation set and training set is almost the same. Kawaguchi [49] points out that the generalization gap can be approximately regarded as the difference between the loss on the training set and the loss on the validation set. The smaller the difference is, the stronger the generalization ability of the model has. Therefore, the result is strong evidence that the CNNs models with center loss can learn the common features of different classes and have a strong generalization ability, and if it is used to make predictions for unknown samples, it can achieve more reliable results. In addition, the training process without center loss occurs overfitting, as the loss of a training set is smaller than that of a validation set in the whole process, which means the learned model cannot predict an unknown sample well.

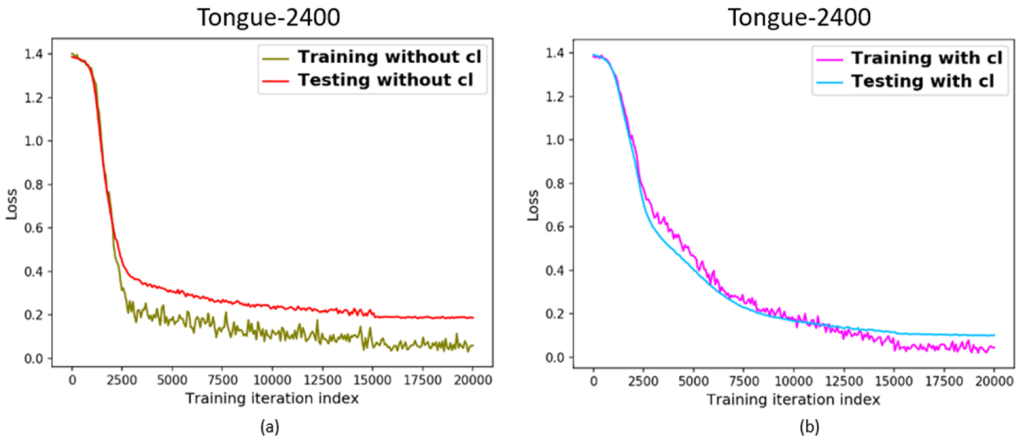

Figure 12. Influence of center loss in training phase: (**a**) without center loss; (**b**) with center loss.

(6) Feature visualization

Further, in order to observe the feature distribution after using the center loss function more intuitively, the distribution of deeply learned features under the supervision of joint loss function during the training process is shown in Figure 13. When only the softmax loss function is used, we can get a feature distribution on the training set and validation set as shown in Figure 13a,b. If the two functions are combined to supervise the CNNs jointly, we can get a feature distribution as shown in Figure 13c,d. From the figure, we can easily see that when the center loss is not used, the deeply learned features are not separable enough and many samples with Label 1 are mixed with Label 0. Besides, the points of each class are too scattered, which makes it difficult for the model to distinguish each class and reduce the generalization ability of the model. However, after using our method, we can observe that the point of each class is close to the center point of its own class, and it is easy to distinguish each class. Moreover, the phenomenon that the points of class 0 and the points of class 1 are mixed with each other is also significantly reduced, indicating that our method enhances the distinguishing ability of the model. Last but not least, the result also shows that the center loss can compress the feature distribution to a smaller feature space, which realizes the efficient storage of features, and it is also helpful to solve the overfitting problem.

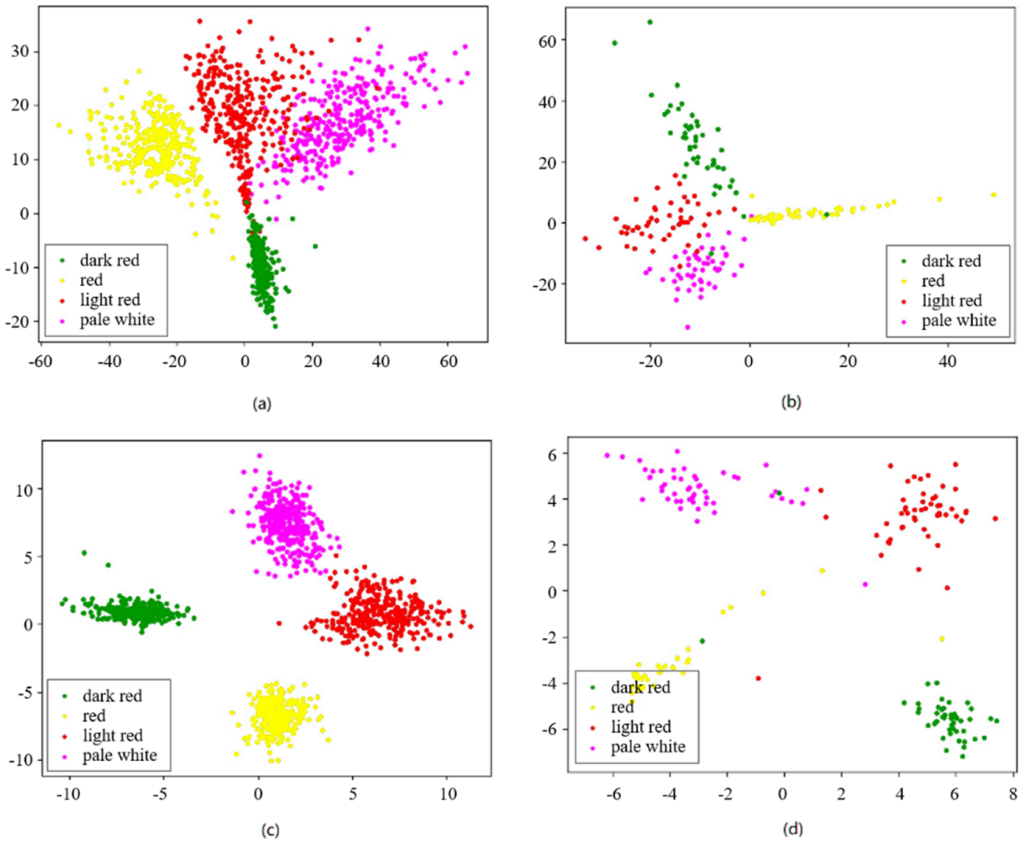

Figure 13. The distribution of learned features: (**a**) the training set without center loss; (**b**) the validation set without center loss; (**c**) the training set with center loss; (**d**) the validation set with center loss.

5. Conclusions

Tongue segmentation and color classification are challenging in TCM. A computer-aided tongue diagnosis system contains tongue segmentation, feature extraction and classification. The traditional methods have shortcomings which limit their usage. In order to improve the performance of the tongue diagnosis automation, we construct the dataset and use CNNs to solve segmentation and classification problems.

This paper proposes a novel tongue color classification method using semantic-based CNNs called SegTongue to effectively segment the tongue body from the background and trains robust tongue color feature extraction models combined with softmax loss and center loss. Semantic-based CNNs with atrous convolution and multiple ASPP modules can both compute feature maps densely and capture image context at multiple scales, which leads to a precise tongue segmentation. Besides, four CNNs models including AlexNet, Vgg-16, ResNet-18 and DenseNet-101 are used to extract features of tongue images and classify tongue-color features using different classifiers such as softmax, SVM and RF. In order to make tongue features more separable and discriminative during the training phase, center loss is applied to compute the loss value within each class.

To validate the performance of the proposed method, we conducted experiments based on different indicators. The results show that our method can not only achieve highly accurate tongue color classification, but also has a strong generalization performance. Thus, our proposed method can help doctors to achieve automated tongue diagnosis. In addition, our method is time-efficient and meets the simultaneous access of multiple internet devices, which has great value in practice.

Author Contributions: Algorithm design, programming and software development, B.Y.; Writing—original draft, S.Z.; Algorithm design, writing—review, editing and proofreading, Z.Y.; Parameter optimization H.S.; Information collection and charting, H.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the Natural Sciences and Engineering Research Council of Canada (NSERC).

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to maintaining privacy.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Tang, J.; Liu, B.; Ma, K. Traditional Chinese medicine. *Lancet* **2008**, *372*, 1938–1940. [CrossRef]
2. Normile, D. The New Face of Traditional Chinese Medicine. *Science* **2003**, *299*, 188–190. [CrossRef] [PubMed]
3. Zeng, X.; Zhang, Q.; Chen, J.; Zhang, G.; Zhou, A.; Wang, Y. Boundary Guidance Hierarchical Network for Real-Time Tongue Segmentation. *arXiv* **2020**, arXiv:2003.06529.
4. Zhang, Q.; Shang, H.; Zhu, J.; Jin, M.; Wang, W.; Kong, Q. A new tongue diagnosis application on Android platform. In Proceedings of the 2013 IEEE International Conference on Bioinformatics and Biomedicine, Shanghai, China, 18–21 December 2013.
5. Pang, B.; Zhang, D.; Li, N.; Wang, K. Computerized tongue diagnosis based on Bayesian networks. *IEEE Trans. Biomed. Eng.* **2004**, *51*, 1803–1810. [CrossRef] [PubMed]
6. Minaee, S.; Boykov, Y.; Porikli, F.; Plaza, A.; Kehtarnavaz, N.; Terzopoulos, D. Image Segmentation Using Deep Learning: A Survey. *IEEE Trans. Pattern. Anal. Mach. Intell.* **2022**, *44*, 3523–3542. [CrossRef]
7. Wei, Y. Tongue Image Segmentation Method Based on Adaptive Thresholds. *Comput. Technol. Dev.* **2011**, *9*, 63–65.
8. Fachrurrozi, M.; Erwin, S. Tongue Image Segmentation using Hybrid Multilevel Otsu Thresholding and Harmony Search Algorithm. *J. Phys. Conf. Ser.* **2019**, *1196*, 2072. [CrossRef]
9. Fu, Z. Tongue Image Segmentation Based on Snake Model and Radial Edge Detection. *J. Image. Graph.* **2009**, *4*, 688–693.
10. Zhang, D.; Zhang, H.; Zhang, B. A Snake-Based Approach to Automated Tongue Image Segmentation. In *Tongue Image Analysis*; Springer: Hong Kong, China, 2017; pp. 71–88.
11. Wei, Y.; Fan, P. Application of improved GrabCut method in tongue diagnosis system. *Trans. Microsyst. Technol.* **2014**, *10*, 157–160.
12. Chen, S.; Fu, H. Application of improved graph theory image segmentation algorithm in tongue image segmentation. *Comput. Eng. Appl.* **2012**, *5*, 201–203.
13. Guo, J.; Yang, Y.; Wu, Q. Adaptive active contour model based automatic tongue image segmentation. In Proceedings of the 9th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI 2016), Datong, China, 15–17 October 2016.
14. Shi, M.; Li, G.; Li, F. C2G2FSnake: Automatic tongue image segmentation utilizing prior knowledge. *Sci. China Inf. Sci.* **2013**, *9*, 1–14. [CrossRef]
15. Li, Q.; Liu, Z. Tongue color analysis and discrimination based on hyperspectral images. *Comput. Med. Imaging Graph.* **2009**, *5*, 217–221. [CrossRef] [PubMed]
16. Cao, G.; Ding, J.; Duan, Y. Classification of tongue images based on doublet and color space dictionary. In Proceedings of the 2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Shenzhen, China, 15–18 December 2016.
17. Diyana, K.; Yee, O. A Fast SVM-Based Tongue's Colour Classification Aided by k-Means Clustering Identifiers and Colour Attributes as Computer-Assisted Tool for Tongue Diagnosis. *J. Healthc. Eng.* **2017**, *2017*, 7460168.
18. Ding, J.; Cao, G.; Meng, D. Classification of Tongue Images Based on Doublet SVM. In Proceedings of the International Symposium on System and Software Reliability, Shanghai, China, 29–30 October 2016.
19. Li, Z.; Pei, Z.; Bo, C. Automatic tongue color analysis of traditional Chinese medicine based on image retrieval. In Proceedings of the International Conference on Control Automation Robotics and Vision, Singapore, 20–22 May 2015.
20. Niu, G.; Wang, C.; Yan, B.; Pan, Y. Tongue Color Classification Based on Convolutional Neural Network. In *Advances in Information and Communication, Proceedings of the 2021 Future of Information and Communication Conference, Vancouver, BC, Canada, 29–30 April 2021*; Springer: Cham, Switzerland, 2021.
21. Wen, Y.; Zhang, K.; Li, Z. A Discriminative Feature Learning Approach for Deep Face Recognition. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016.
22. Guo, A.J.X.; Zhu, F. Spectral-Spatial Feature Extraction and Classification by ANN Supervised With Center Loss in Hyperspectral Imagery. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 1755–1767. [CrossRef]
23. Lu, Y.; Li, X.; Zhang, H.; Zhang, J.; Zhuo, L. Review on Tongue Image Segmentation Technologies for Traditional Chinese Medicine: Methodologies, Performances and Prospects. *Acta Autom. Sin.* **2021**, *47*, 1005–1016.
24. Gao, S.; Guo, N.; Mao, D. LSM-SEC: Tongue Segmentation by the Level Set Model with Symmetry and Edge Constraints. *Comput. Intell. Neurosci.* **2021**, *2021*, 6370526. [CrossRef] [PubMed]

25. Huang, Z.; Miao, J.; Song, H.; Yang, S.; Zhong, Y.; Xu, Q.; Tan, Y.; Wen, C.; Guo, J. A novel tongue segmentation method based on improved U-Net. *Neurocomputing* **2022**, *500*, 73–89. [CrossRef]
26. Qu, P.; Zhang, H.; Zhuo, L. Automatic Tongue Image Segmentation for Traditional Chinese Medicine Using Deep Neural Network. In Proceedings of the Intelligent Computing Theories and Application, Liverpool, UK, 7–10 August 2017.
27. Lin, B.; Xie, J.; Li, C. Deeptongue: Tongue segmentation via resnet. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Seoul, Republic of Korea, 22–28 April 2018.
28. Zhou, J.; Zhang, Q.; Zhang, B.; Chen, X. TongueNet: A Precise and Fast Tongue Segmentation System Using U-Net with a Morphological Processing Layer. *Appl. Sci.* **2019**, *9*, 3128. [CrossRef]
29. Mozaffari, M.; Lee, W. Encoder-Decoder CNN Models for Automatic Tracking of Tongue Contours in Real-time Ultrasound Data. *Methods* **2020**, *179*, 26–36. [CrossRef] [PubMed]
30. Hu, Y.; Wen, G.; Liao, H. Automatic construction of Chinese herbal prescription from tongue image via CNNs and auxiliary latent therapy topics. *IEEE Trans. Cybern.* **2018**, *10*, 708–721.
31. Ma, J.; Wen, G.; Hu, Y. Complexity perception classification method for tongue constitution recognition. *Artif. Intell. Med.* **2019**, *96*, 123–133. [CrossRef] [PubMed]
32. Lu, Y.; Li, X.; Gong, Z.; Zhuo, L. TDCCN: A Two-Phase Deep Color Correction Network for Traditional Chinese Medicine Tongue Images. *Appl. Sci.* **2020**, *10*, 1784. [CrossRef]
33. Zhang, J.; Xu, J.; Hu, X.; Chen, Q.; Tu, L.; Huang, J.; Cui, J. Diagnostic Method of Diabetes Based on Support Vector Machine and Tongue Images. *Biomed. Res. Int.* **2017**, *2017*, 7961494. [CrossRef] [PubMed]
34. Mansour, R.F.; Althobaiti, M.M.; Ashour, A.A. Internet of Things and Synergic Deep Learning Based Biomedical Tongue Color Image Analysis for Disease Diagnosis and Classification. *IEEE Access* **2021**, *9*, 94769–94779. [CrossRef]
35. Li, J.; Zhang, Z.; Zhu, X.; Zhao, Y.; Ma, Y.; Zang, J.; Li, B.; Cao, X.; Xue, C. Automatic Classification Framework of Tongue Feature Based on Convolutional Neural Networks. *Micromachines* **2022**, *13*, 501. [CrossRef]
36. Holschneider, M.; Morlet, J. A Real-Time Algorithm for Signal Analysis with the Help of the Wavelet Transform. In *Wavelets: Time Frequency Methods and Phase Space*; Springer: NewYork, NY, USA, 1989; pp. 286–297.
37. Krizhevsky, A.; Sutskever, I.; Hinton, G. ImageNet Classification with Deep Convolutional Neural Networks. In Proceedings of the 26th Neural Information Proceeding Systems(NIPS), Nevada City, NV, USA, 3–6 December 2012.
38. Szegedy, C.; Liu, W.; Jia, Y. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015.
39. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In Proceedings of the 3rd International Conference on Learning Representations, San Diego, CA, USA, 7–9 May 2015.
40. He, K.; Zhang, X.; Ren, S. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016.
41. Huang, G.; Liu, Z. Densely Connected Convolutional Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 22–25 July 2017.
42. He, K.; Zhang, X.; Ren, S. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *37*, 1904–1916. [CrossRef] [PubMed]
43. Chen, C.; Papandreou, G.; Kokkinos, I. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 838–848. [CrossRef] [PubMed]
44. Long, J.; Shelhamer, E.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *39*, 640–651.
45. Buda, M.; Maki, A. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Netw.* **2018**, *106*, 249–259. [CrossRef]
46. Ling, C.; Li, C. Data mining for direct marketing: Problems and solutions. In Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 27–31 August 1998.
47. Kingma, D.; Ba, J. Adam: A Method for Stochastic Optimization. In Proceedings of the 3rd International Conference for Learning Representations, San Diego, CA, USA, 7–9 May 2015.
48. Cohen, J. Weighted kappa: Nominal scale agreement with provision for scaled disagreement or partial credit. *Psychol. Bull.* **1968**, *70*, 213–220. [CrossRef]
49. Kawaguchi, K.; Kaelbling, L.; Bengio, Y. Generalization in Deep Learning. *arXiv* **2017**, arXiv:1710.05468.