廣東工業大學

本科毕业设计（论文）

基于手机后置摄像和电脑屏幕显示的舌苔 AI 导诊系统

学　　院	自动化学院
专　　业	物联网工程
年级班别	2020 级（2）班
学　　号	3120001462
学生姓名	叶星华
指导教师	何昭水

2024 年 6 月

基于手机后置摄像和电脑屏幕显示的舌苔AI导诊系统

叶星华

自动化学院

摘 要

在中医诊断领域，舌诊作为一种简便、直观的诊断方法，历来受到医家的重视。对于舌苔的观察不仅能够反映人体内脏的功能状态，还能够揭示患者的体质特征。例如，某些舌象可能表明患者存在气血瘀滞、阴虚火旺、痰湿内蕴等问题。随着人工智能技术的快速发展，将 AI 技术应用于中医舌诊，实现智能化的舌苔分析和体质识别，已成为研究的新趋势。

本文旨在开发一款基于手机后置摄像和电脑屏幕显示的舌苔 AI 导诊系统。系统首先采用 DeepLab V3+模型对舌体进行精确分割，然后利用 ViT 模型对舌苔进行分类。为了实现用户界面的友好交互，本文使用 PyQt5 库构建了电脑端应用程序，并开发了相应的 Android 手机端 App。此外，通过 Flask 服务器端技术，系统能够实现图像的高效传输。最终，这些组件将被集成到一个完整的系统中，为用户提供中医体质特征的智能识别和用药指导服务。本文主要内容如下：

首先，本文介绍了中医舌诊的基本原理，突出舌象特征在中医诊断中的核心作用。接着，探讨卷积神经网络（CNN）和 Transformer 这两种深度学习模型的结构和关键原理，以及它们在图像分割和分类任务中的应用。通过比较基于 CNN 的图像分割模型和基于 Transformer 的图像分类模型的优缺点，本文阐明了选择 DeepLab V3+进行舌体分割和 ViT 进行舌苔分类的原因。

其次，本文对开源数据集中的图像进行筛选，剔除了质量较差的样本。然后，对 1360 张原始舌头图像进行了预处理，包括翻转和对比度调整，以构建用于舌体分割和舌苔分类的数据集。

在舌体分割任务中，本文采用了迁移学习策略，并选择了 DeepLab V3+模型，以 MobileNet V2 作为主干网络。该模型利用深度可分离卷积技术，有效降低了模型的参数规模和运算需求，实现了高效的舌体分割。实验结果表明，DeepLab V3+在本文数据集上的 MIoU 达到了 91.6%，显示了较高的分割精度。

在舌苔分类方面，本文采用了 ViT 模型。ViT 不依赖于卷积操作，而是完全基于自注意力机制来处理图像数据。这种机制使得 ViT 能够较为有效地识别舌苔的不同类型。实验结果表明，ViT 在本文数据集上的分类准确率达到了 72%，为构建一个高效的舌苔分类系统提供了支持。

最后，为了提供用户友好的交互体验，本文开发了基于 PyQt5 的电脑端可视化程序，以及一个移动端 Android App。通过 Flask 后端服务，本文实现了从手机端到电脑端的舌头图像文件传输功能。这一流程的设计使得智能中医舌诊服务方便快捷。

关键词：中医舌诊，深度学习，舌体分割，舌苔分类，可视化程序

Abstract

In the field of traditional Chinese medicine (TCM) diagnosis, tongue diagnosis has always been valued by practitioners as a simple and intuitive diagnostic method. Observation of the tongue coating can not only reflect the functional state of the internal organs but also reveal the patient's constitutional characteristics. For instance, certain tongue appearances may indicate issues such as Qi and blood stasis, yin deficiency with internal heat, or phlegm-dampness accumulation. With the rapid development of artificial intelligence technology, applying AI to TCM tongue diagnosis to achieve intelligent tongue coating analysis and constitutional identification has become a new trend in research.

This paper aims to develop a tongue coating AI diagnostic system based on the rear camera of a smartphone and the display of a computer screen. The system first uses the DeepLab V3+ model for precise segmentation of the tongue body, and then employs the ViT model for tongue coating classification. To achieve a user-friendly interface interaction, this paper uses the PyQt5 library to build a computer-side application and develops a corresponding Android mobile app. Additionally, through Flask server-side technology, the system is capable of efficient image transfer. Ultimately, these components will be integrated into a complete system to provide intelligent identification of TCM constitutional characteristics and guidance on medication usage. The main content of this paper is as follows:

Firstly, this paper introduces the basic principles of TCM tongue diagnosis, highlighting the core role of tongue appearance features in TCM diagnosis. Then, it discusses the structure and key principles of convolutional neural networks (CNN) and Transformer, two types of deep learning models, as well as their application in image segmentation and classification tasks. By comparing the advantages and disadvantages of CNN-based image segmentation models and Transformer-based image classification models, this paper clarifies the reasons for choosing DeepLab V3+ for tongue body segmentation and ViT for tongue coating classification.

Secondly, this paper filters the images in the open-source dataset, eliminating those of

poor quality. Then, 1360 original tongue images were preprocessed, including operations such as flipping and contrast adjustment, to construct a dataset for tongue body segmentation and tongue coating classification.

In the task of tongue body segmentation, this paper adopts a transfer learning strategy and selects the DeepLab V3+ model, with MobileNet V2 as the backbone network. The model utilizes depthwise separable convolution technology, effectively reducing the model's parameter scale and computational requirements, achieving efficient segmentation of the tongue body. Experimental results show that DeepLab V3+ achieved an MIoU of 91.6% on the dataset of this paper, demonstrating high segmentation accuracy.

In terms of tongue coating classification, this paper uses the ViT model. ViT does not rely on convolutional operations but is entirely based on the self-attention mechanism to process image data. This mechanism enables ViT to effectively identify different types of tongue coatings. Experimental results show that ViT achieved a classification accuracy of 72% on the dataset of this paper, providing support for building an efficient tongue coating classification system.

Finally, to provide a user-friendly interaction experience, this paper has developed a computer-side visualization program based on PyQt5, as well as a mobile Android app. Through Flask server-side technology, this paper has realized the transfer function of tongue image files from the mobile side to the computer side. The design of this process makes the intelligent TCM tongue diagnosis service convenient and fast.

Key words: TCM tongue diagnosis, deep learning, tongue image segmentation, tongue coating classification, graphical user interface program

目 录

第一章　绪论

1.1 研究背景及意义

在传统中医的诊断过程中，舌诊扮演着一个关键角色。这种方法主要通过观察舌头的颜色、形态和表面覆盖物等特征[1]，来辅助识别和区分人体的健康与疾病状态。中医舌诊的历史悠久，自古以来就有记录，历代医学家不断积累经验，但直到元代《敖氏伤寒金镜录》的出现，才有了系统的图谱和文字说明[2]，将舌象与病证联系起来，为舌诊的发展奠定了基础。近些年的多项研究[3-6]已经陆续揭示了舌象特征与多种常见疾病之间的潜在联系。这些疾病包括但不限于糖尿病、冠状动脉疾病、乳腺恶性肿瘤、肺部恶性肿瘤以及急性脑血液循环障碍等。

全球人口结构的老龄化及生活方式的演变使得慢性病和亚健康状态变得日益常见，公众对健康问题的关注和需求日益增长。在众多健康维护策略中，中医学凭借其独到的理论基础和临床实践经验，为个体健康提供了丰富的知识和技术支持。作为中医诊断核心组成部分的舌诊，通过分析舌头的外观来评估个体的健康状况及疾病进展，这种诊断方式因其无创性、直观性和便捷性而广受欢迎。

虽然传统中医舌诊是一种古老的诊断技术，但是其准确性常受到医生个人经验和主观意向的影响，于是可能导致诊断结果出现波动，甚至误诊。另外，面对各大城市快速的生活节奏，城市民众对医疗服务的效率和便捷性有更高的要求。相比之下，传统中医舌诊较为耗时，而且前往中医院看病存在包括预约困难、候诊时间长等问题。因此，提高中医舌诊的准确度、减少主观判断和增强其便捷性，是智能化中医舌诊的关键。

随着 AI 技术，特别是深度学习算法的迅速进步，智能化中医舌诊发展迎来了发展的关键时期。深度学习在多个领域，如图像识别、自然语言处理和数据分析等方面，已经取得了突破性的进展。利用深度学习对患者舌部进行智能化的图像分析和特征识别，不仅能够提高诊断的准确率，还能使用户便捷地在任何时间、任何地点进行舌象检查，从而获取相关的健康信息，这大大增强了中医舌诊的方便快捷。

近些年，智能手机的普及和后置摄像头技术的进步，使得利用手机后置摄像头进行舌头图像采集变得快捷容易。同时，深度学习技术的发展，尤其是图像分割模型和图像分类模型以及 CNN 和 Transformer 的应用，为舌苔自动识别和中医体质特征推断提供了强大的技术支撑。因此，结合这些技术，开发一款基于手机后置摄像和电脑屏幕显示的

舌苔 AI 导诊系统，将大大促进中医舌诊的智能化和便捷化发展。

综上所述，本文舌苔 AI 导诊系统的开发不仅有助于提高中医舌诊的准确率和便捷性，而且对于推动中医舌诊的智能化具有重要的现实意义，还可为快节奏生活的人们提供可靠的健康管理。

1.2 国内外研究现状

1.2.1 舌体分割研究现状

舌体分割是中医舌诊现代化研究的关键环节，它利用深度学习技术对舌体进行分析和识别，旨在实现舌象特征的客观化和量化研究。在一张含有舌体的图像中，只有舌体部分才是研究者们关心的感兴趣区域(Region of Interest, ROI)，其他部分如人脸、嘴唇等背景需要利用图像分割技术初步排除，进而得到只含舌体的图像。类似的处理步骤有助于减少非舌体部分的干扰，提高分类的准确性和鲁棒性，并且由于 ROI 远小于原始图像，还可以降低后续处理步骤的计算复杂性。

2017 年，Panling Qu、Hui Zhang 等研究者提出了一种基于亮度统计的图像质量评估方法来决定输入图像是否适合分割，并使用 SegNet 在专为舌体图像分割构建的两个数据集上进行训练，获得了用于自动分割的深度模型[7]。该方法在两个数据集上的平均交并比分别达到了 95.89%和 90.72%，相较于传统舌体图像分析技术，该深度学习策略省去了繁琐的特征手工提取步骤，同时在分割效率上展现出显著的优越性。2020 年，马龙祥，杨浩等研究者提出了一种基于高分辨率网络的分割算法[8]，通过区域定位网络生成建议框并利用高清晰度特征进行细致的舌象划分，成功地保留了舌象轮廓的细节，相较于 SegNet 和 Mask R-CNN，实现了更精细的分割效果。2023 年，研究者谭建聪、肖晓霞与邹北骥使用了 BlendMask 算法对采集的高分辨率舌面图像进行预裁剪和标注后，实现了舌体的自动分割。他们通过与分水岭算法、GrabCut 以及 Mask R-CNN 等现有技术的对比分析，BlendMask 在舌体定位的精确度上分别实现了 54.57%、29.25%和 6.40%的显著提升，最终达到了 99.77%的高准确率[9]，从而准确分割舌体并支持智能舌诊。

1.2.2 舌象识别与分析研究现状

将舌体从原始图像中检测并且分割出来后，就需要利用技术手段进行舌象识别与分

析，从而得到最终的诊断结果。舌象识别与分析是舌诊中的一个重要环节，它涉及对舌头的颜色、形状、舌苔等特征的观察和解读，以评估个体的健康状况。在现代医学研究中，为了提高舌诊的准确性和客观性，研究者们采用了多种技术和方法来进行舌象的自动识别与分析。

梁金鹏、杨浩等研究者采用多项式回归进行舌象图像颜色校正，利用内积阈值和颜色聚类分割舌体[10]，并通过引入权值改进的算法模型，优化了中医舌诊中 6 种舌质、舌苔类型的自动分类识别方法，提升了分类的准确性。瞿婷婷、夏春明等研究者提出了一种基于 Gabor 小波变换的舌苔腐腻识别方法[11]，通过弱化 Gabor 变换中的边缘效应以增强纹理描述，并提取均值和标准差作为纹理特征，对正常苔、腐苔和腻苔进行分类识别，实验结果显示该方法的平均准确率达到 91%。

以上研究者结合了传统的图像处理和机器学习方法来进行舌象的识别与分析，而以下研究者则采用了深度学习方法识别与分析舌象。

钟振通过改进的 U-Net 网络和分离注意网络 ResNeSt 模型[12]，结合统计学方法，成功实现了中医舌像的自动分割和舌色苔色的高精度识别，为舌诊客观化提供了强有力的技术支持。刘伟、陈锦明等研究者构建了名为 Tongue Color Classification Net (TCCNet) 的深度学习网络[13]，利用 Triplet-Loss 增强不同舌色类别间的嵌入分离，有效提升了舌色分类的性能。就在最近，董易杭、王建勋等研究者采用 U-net 网络进行舌体分割，基于 ResNet-34 构建分类模型[14]，并结合交叉熵损失和 Dice 损失进行优化，通过自动化分析舌象图像，成功区分阳虚质和阴虚质舌象。

1.3 论文主要内容及章节安排

本文的核心任务是融合深度学习技术与中医舌诊，深入探讨目标检测、图像分割和分类算法的优劣，选择合适的算法构建深度学习模型，并将其集成到 PyQt5 用户界面和 Android 应用中，旨在开发一个基于手机摄像头和电脑屏幕的舌苔 AI 导诊系统，以提升中医舌诊的精确性和客观性，同时提供便捷的中医舌诊服务。本文共分为五章，各章主要内容如下：

第一章 绪论。分析智能舌诊系统开发的研究背景及意义，结合实际情况提出舌苔 AI 导诊系统开发的必要性，并概述国内外对于舌体分割和舌象识别与分析的研究进展，

最后明确本文研究的主要内容及各章节的组织架构。

第二章 舌苔 AI 导诊系统相关技术分析及总体方案设计。首先概述卷积神经网络的主要组成部分和核心原理，然后分析基于卷积神经网络的图像分割模型与图像分类模型，并说明本文选择了 DeepLab V3+和 ResNet 作为分割与分类模型，最后介绍深度学习框架及选择。

第三章 利用 DeepLab V3+进行舌体分割。首先介绍舌体分割数据集，包括图像数据的来源和规模。随后，详细阐述数据预处理的步骤，包括图像标注和增强等。第二节介绍了迁移学习的策略。本章还详细介绍了 DeepLab V3+的模型结构，包括其编码器-解码器架构和特征融合策略，以及阐述了 DeepLab V3+在处理图像分割任务中的优势。最后，展示 DeepLab V3+模型在舌体分割任务上的实验结果，并使用评价指标 MIoU 对模型性能进行分析，还直观对比 DeepLab V3+与 U-Net 的分割效果。

第四章，利用 ViT 进行舌苔分类。本章首先介绍用于舌苔分类的数据集，说明数据集的构成。接着，介绍数据预处理的几个方法，包括图像翻转、亮度调整、饱和度调整、对比度调整以及锐度调节等操作。本章深入还分析 ViT 模型的核心原理和结构，特别是其自注意力机制。最后，展示 ViT 舌苔分类的实验结果。

第五章，舌苔 AI 导诊系统。本章详细介绍分别利用 PyQt5 开发电脑端可视化程序和 Android Studio 开发手机端 App 以及基于 Python 实现的 Flask 服务器端的相关开发技术以及技术细节，并展示系统的可视化界面。

第二章 舌苔 AI 导诊系统相关技术分析及总体方案设计

2.1 卷积神经网络概述

作为深度学习领域的一种先进架构，卷积神经网络（Convolutional Neural Network, CNN）在图像识别和分类任务中表现出色。CNN 的核心思想是通过卷积层来提取图像的局部细节，然后通过非线性激活函数、池化操作以及全连接层等来进行更高层次的学习和分类。CNN 在图像的自动分类、物体的精确定位与识别、场景的详细解析以及对人类语言的深入理解和生成等领域有着广泛的应用，是目前深度学习领域最强大的工具之一。以下为 CNN 的组成部分与核心原理的概述。

2.1.1 卷积层

在 CNN 中，卷积层处于核心地位，负责通过卷积运算来识别和提取输入图像的局部模式。卷积层包含一系列的滤波器（或称为卷积核），这些滤波器在输入数据上滑动以生成特征图。

作为深度前馈神经网络的一个关键层，卷积层拥有局部感知连接和参数共享的能力。卷积层由一系列的滤波器组成，常用的滤波器大小为 3×3，其会覆盖输入数据的一个 3×3 局部区域。卷积层通过将滤波器在输入数据上滑动来工作。在滑动过程中，滤波器的每个元素与输入数据的相应元素相乘，并将乘积求和，生成一个单一的输出值。这个过程对输入数据的每个局部区域重复进行，生成一个特征图。图 2.1 给出了二维卷积示例。

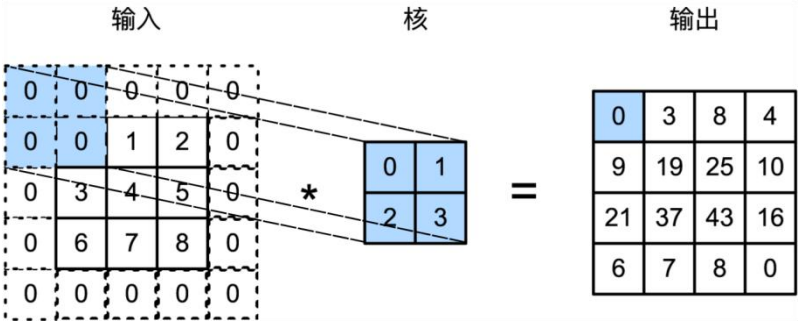


图 2.1 二维卷积示例

卷积神经网络的设计灵感源自生物神经系统中的感受区域原理。在此网络架构中，局部感受区域（Local Receptive Fields）的功能表现为卷积层内单个神经元仅对输入图像的特定局部范围或片段产生反应。这意味着每个神经元或滤波器只关注输入图像的一小部分，而不是整个图像。在卷积神经网络中，通过使用小的滤波器（例如 3×3 或 5×5

像素），可以在图像上滑动以覆盖不同的局部区域，从而提取局部特征。这种方法不仅减少了模型的计算负担，而且提高了网络对平移等变化的不变性。局部感受野的设计使得卷积神经网络特别适合于图像识别任务，因为图像中的许多重要特征都是局部的。

在卷积神经网络的设计中，参数共享是一个核心概念。它指的是在卷积过程中使用的滤波器或卷积核在整个输入图像上以相同的权重重复应用，这意味着无论滤波器在图像的哪个位置，其权重都保持不变。在卷积神经网络的架构中，通过在不同层次间复用相同的权重，实现了参数的一致性。这种策略显著降低了模型所需的总参数量，从而精简了网络结构。参数共享的另一个重要效果是增强了网络的平移不变性，即无论图像中的特征出现在哪个位置，网络都能够有效地识别出相同的特征。这种特性对于图像识别任务至关重要，因为目标对象在图像中的位置是多变的。

2.1.2 池化层

池化层可用于降低特征图空间维度，并且有助于提取图像的重要特征并增加对图像变化的不变性，常见的池化操作有最大池化和平均池化，如图 2.2 所示。

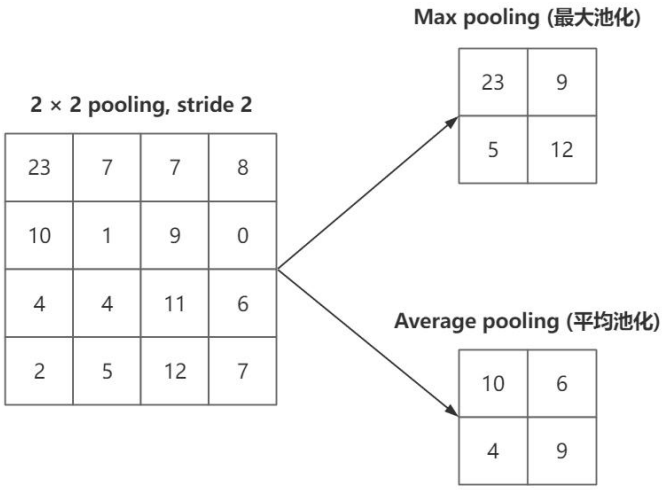


图 2.2 最大池化与平均池化

池化层的工作是在已提取出的特征图中的一个局部小区域，求最大值或者求平均值，使特征图变小，同时还能保留该区域中最重要的信息。这样做的好处是减少要处理的数据量，即将为。最大池化，就是选择每个小区域中的最大值，目的是突出那些特别明显的特征，而且即使特征的位置产生细微变动，也不会影响到结果。而平均池化，就是计算每个小区域里的所有数值的平均值，这样得到的特征图会更加平滑，有助于得到一个稳定的特征表示。池化操作在一个可滑动的窗口上执行，该窗口覆盖特征图的局部区域，

并通过步长控制窗口的移动，从而影响输出特征图的尺寸。在多通道特征图中，每个通道独立进行池化操作，然后合并结果，保持输出特征图的深度。通过降低特征的空间分辨率，池化层增强了网络对图像平移、缩放和旋转等变化的不变性[15]。池化使得特征图的尺寸变小、参数变少，有助于减少网络的过拟合风险。池化层通常位于卷积层之后，可以在 CNN 中多次使用，与卷积层搭配使用，这能有效提取原始数据中有用的特征。

2.1.3 全连接层

在神经网络架构中，全连接层承担着将卷积层或池化层传递的高级特征映射转换为模型输出的任务，通常是类别预测或连续值。

全连接层将从卷积层和池化层传递过来的局部细节，综合为可以代表整个图片的特征，用于执行分类或其他任务。全连接层的设计特点在于其内部的每个神经元均与前一层的全部神经元建立连接[16]，构成了一个密集的网络结构。正是这种全面的连接模式导致了全连接层拥有相对较多的参数数量。在这些参数中，权重和偏置是关键，它们与非线性激活函数（例如修正线性单元 ReLU）相结合，旨在提升网络处理复杂关系的能力。在用于分类的 CNN 架构里，最后的全连接层扮演着输出层的角色，其包含的神经元数目直接对应于待分类对象的种类总数。这一层的输出通过 Softmax 函数转换为类别概率分布。全连接层由于参数众多，可能会导致过拟合。为了减少这种风险，可以采用 dropout、L2 正则化等正则化技术。在某些深度学习架构中，全连接层可以堆叠多个，以增加网络的深度，提高其学习和表示复杂模式的能力。在训练阶段，网络利用反向传播机制，一步步调整全连接层的各种参数，目的是使模型预测结果与实际标签的差距尽可能小，进而提高模型在预测任务上的精确度。

2.1.4 非线性激活函数

非线性激活函数用于在网络的神经元或节点之间引入非线性[17]。常见的激活函数包括 ReLU、Sigmoid、Tanh 和 Softmax，这四种函数曲线如图 2.3 所示。

ReLU 因其计算简单、训练效率高而被广泛使用在各种深度学习模型中。ReLU 在正区间内是线性的，而在负区间内输出为零。这种操作不仅计算效率高，而且在模型训练期间不易使梯度下降较小，进而加快收敛速度。Sigmoid 常用于二分类，它将神经网络的输出转换到 0 和 1 之间的概率值。Tanh 函数图像形状与 Sigmoid 的相似，但它将输

出转换到－1 到 1，可使结果的平均值接近零，方便计算机处理。Softmax 则常用于多分类问题，它将任何数字输入转换为一个和为 1 的概率输出。

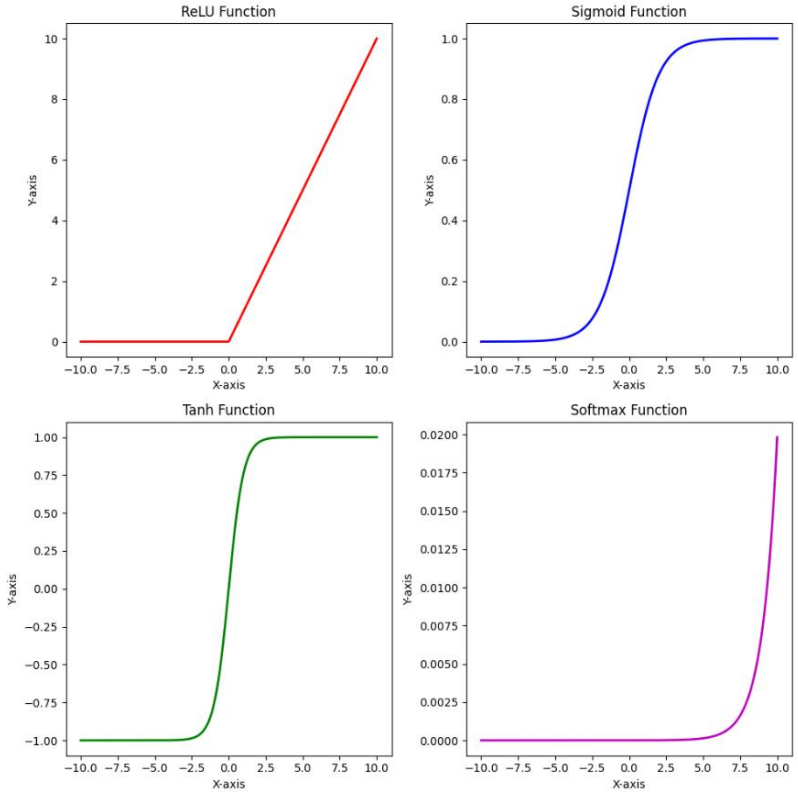


图 2.3 四种激活函数曲线图

但是这些函数也有缺陷。例如 Sigmoid 和 Tanh，当出现输入值过大或过小，可能会导致梯度消失，这会使网络的学习速度大大减慢。而 ReLU 在负区间的梯度为零，可以避免梯度消失，但在正区间内梯度恒定，可能导致梯度爆炸。所以，激活函数的选择需要根据具体任务和网络结构。以下为 ReLU、Sigmoid、Tanh 和 Softmax 的数学表达式。

ReLU 函数表达式为：

$$f(x) = ReLU(x) = \max(0, x) \tag{2.1}$$

Sigmoid 函数表达式为：

$$f(x) = \sigma(x) = \frac{1}{1+e^{-x}} \tag{2.2}$$

Tanh 函数表达式为：

$$f(x) = \tanh(x) = \frac{e^{x}+e^{-x}}{e^{x}+e^{-x}} \tag{2.3}$$

Softmax 函数通常用于多分类的最后一层。对于向量 $x = [x_1, x_2, ..., x_n]$，表达式为：

$$f(x)_i = soft\max(x)_i = \frac{\exp(x_i)}{\sum_{j=1}^{n}\exp(x_j)} \tag{2.4}$$

2.1.5 损失函数

损失函数，亦称代价函数，是用于量化机器学习模型预测输出与真实值之间的偏差[18]。损失函数会提供一个数值指标来评估模型的预测性能，研究者需要使该数值尽可能小，从而调整模型参数以提高预测准确性。下面简要列举了几种损失函数：

均方误差（MSE），通过取预测输出与实际观测值之差的平方，然后求取这些平方差的平均值来评估模型的预测性能[19]。均方误差的数学表达式为：

$$MSE = \frac{1}{m}\sum_{i=1}^{m}\left(y_i - \hat{y}_i\right)^2 \tag{2.5}$$

交叉熵（Cross-Entropy），常用于分类问题，通过比较模型预测的概率模式和实际发生事件的概率模式[20]，来量化两者之间的不一致性。对于多分类问题，离散概率分布 p 和预测概率分布 $\hat{p}$，交叉熵的数学表达式为：

$$H\left(p,\hat{p}\right) = -\sum_{k=1}^{K} p_k \log\left(\hat{p}_k\right) \tag{2.6}$$

对于二分类问题（逻辑回归），用伯努利交叉熵：

$$CE(y,\hat{y}) = -y\log(\hat{y}) - (1-y)\log(1-\hat{y}) \tag{2.7}$$

Hinge Loss，在支持向量机中使用，用于最大化样本间的间隔，其数学表达式为：

$$L = \max(0, 1-(y\cdot(x^T w))) \tag{2.8}$$

损失函数通过梯度下降等优化算法，调整模型权重，使损失函数的值最小化。在神经网络中，损失函数的梯度通过反向传播算法传递到网络的每一层，更新每个层的权重。设计损失函数时，目标是优化模型的性能，确保其在训练集上的准确度，并能泛化至训练集之外的数据，从而在未知数据上同样展现出优秀的预测效果。损失函数的选择须适合要解决的问题和模型类型，不同的问题可能需要不同的损失函数来抓取数据的重要部分。损失函数值通常用于判断模型是否表现良好，但一个模型是否优秀，还需要结合其他指标如准确率、召回率等综合评估。

2.1.6 反向传播

反向传播分为两个主要步骤：误差的反向传播和梯度的计算。首先，计算损失函在输出层的梯度。然后，这个误差梯度从输出层开始，一层层往回传至输入层。每传一层，就根据这个梯度来调整那一层的权重。在神经网络的每一层，反向传播通过应用链式法则来确定损失相对于权重的梯度。对于网络中的每一权重，必须计算两个关键的导数：一是损失函数相对于前一层激活输出的偏导数，二是激活函数相对于该权重的偏导数。计算完成后，可以采用梯度下降或其他优化技术来对网络权重进行调整。权重的更新规则通常是：

$$w_{new} = w_{old} - \alpha \cdot \frac{\partial Loss}{\partial w} \tag{2.9}$$

其中，w_{old} 是旧权重，w_{new} 是新权重，α 是学习率，而 $\frac{\partial Loss}{\partial w}$ 是损失函数关于该权重的梯度。反向传播算法是迭代进行的。每次迭代包括前向传播以确定预测值和损失，随后是反向传播以计算梯度，并据此更新权重[21]。此过程不断重复，直至网络权重稳定至损失函数最小化的状态。

2.2 基于卷积神经网络的图像分割模型分析

CNN 在图像分割领域的应用涵盖从自动驾驶到医学诊断等众多场景，它强大的特征提取能力，能够实现精确分割。图像分割的目的是将像素分类到图像的多个分割区域，流行的基于卷积神经网络的分割技术包括全卷积网络（FCN）、U-Net、SegNet 以及 DeepLab 系列。本文将重点探讨 U-Net 和 DeepLab 算法。

2.2.1 U-Net

U-Net 专门用于图像分割任务，特别是在医学图像分割领域表现出色。它的设计初衷是为了解决生物医学图像分割问题，尤其是当样本数量有限时。

U-Net 采用了对称的编码器-解码器结构，形状与字母“U”相似，因此得名“U-Net”，其结构如图 2.4 所示。编码器部分的工作是减少图像细节，但是能够从中找出重要的信息。它使用卷积层和池化层从图片里找到有用的特征。解码器部分的任务是把编码器提取的特征重组为一张图像，但是这张图像包含更多高级特征，能帮助模型更精确地分辨图像中的像素。

U-Net 的一个关键的特性是“跳跃连接”，它将编码器中得到的特征直接输入到解码

器的对应层，目的是在恢复图像细节时不易丢失重要的特征。由于跳跃连接的使用，U-Net 在处理图像时能够保留更多细节，从而在图像分割中实现更精确的边界识别。但是，由于 U-Net 的结构比较复杂，所以需要更多的计算资源。U-Net 在处理尺寸特别大或特别小的物体时能力有限，因为它主要是用来捕捉图片中的局部信息的。U-Net 的深度和宽度限制了其性能，对于非常深或非常宽的网络，可能需要调整架构。

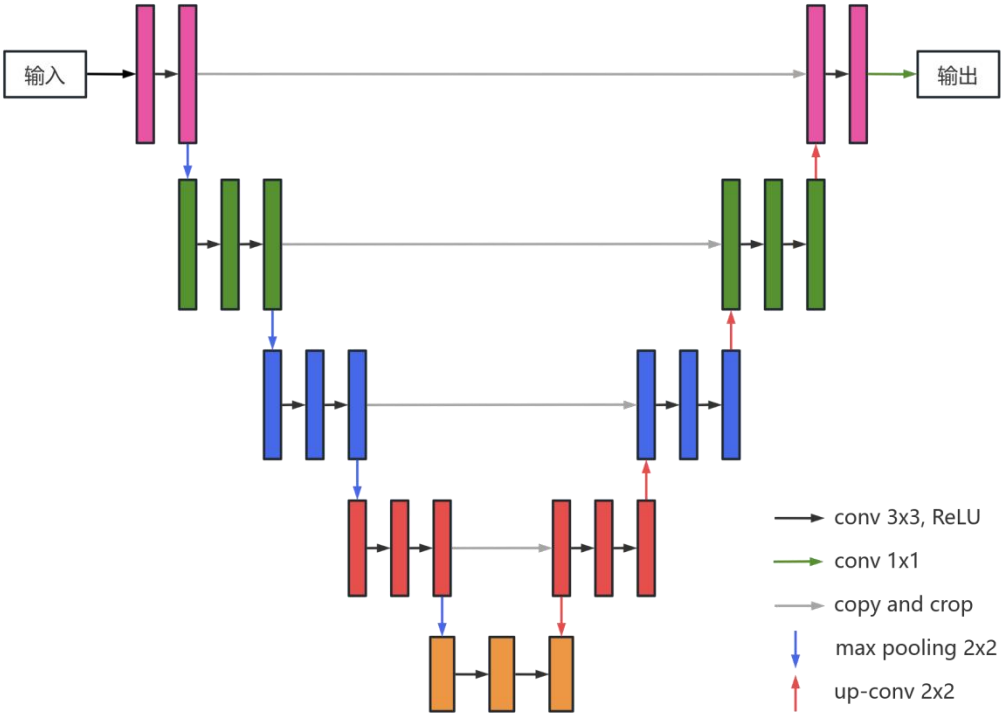


图 2.4 U-Net 网络结构

U-Net 主要适用于像素级图像分割任务，尤其是在医学图像分割领域，如细胞分割。除了医学图像，它也被用于其他图像分割任务，如分析卫星照片或在自动驾驶技术中识别路面。U-Net 很灵活，可以根据不同的任务进行调整，让分割的效果更佳。例如，通过加入注意力机制或 Transformer 结构，可以进一步提升 U-Net 的性能。总而言之，U-Net 是一个强大的图像分割模型，尤其适合于医学图像分割任务，但其计算成本和对特定类型物体分割的局限性也需要考虑。

2.2.2 DeepLab

DeepLab 是由谷歌团队提出的一种用于语义图像分割的深度学习算法系列。DeepLab 系列的首个版本 DeepLab V1 于 2014 年推出，随后在 2017 到 2018 年间，谷歌团队相继推出了 DeepLab V2、DeepLab V3 及 DeepLab V3+。DeepLab 系列算法在

PASCAL VOC2012 等数据集上取得了优异的成绩，成为语义分割领域的领先算法。DeepLab 系列算法的核心原理包括以下几个关键点：空洞卷积、空洞空间金字塔池化、条件随机场和编码器-解码器架构。下文主要介绍前三者的核心原理。

空洞卷积（Atrous Convolution），也称为扩张卷积（Dilated Convolution），是一种特殊类型的卷积运算，它通过在卷积核的元素之间添加间隔（空洞）来增大卷积核的覆盖范围，而不增加卷积核的参数数量或计算量[22]。这种方法允许网络捕获更广泛的上下文信息，同时保持较高的空间分辨率，这对于图像分割等任务特别有用。

在标准的卷积中，卷积核覆盖的输入特征图区域与卷积核大小成正比。而空洞卷积通过增加一个参数（空洞率或扩张率，dilation rate）来调整卷积核的覆盖范围，该参数定义了卷积核中元素之间的间距[23]。空洞卷积的等价卷积核大小可以通过以下公式计算：

$$K = k + (k-1)(r-1) \tag{2.10}$$

其中，k 是原始卷积核大小，r 是空洞率。

空洞空间金字塔池化（Atrous Spatial Pyramid Pooling, ASPP）是深度学习中用来帮助模型识别不同尺寸物体的技术。它通过使用不同空洞率的空洞卷积，让模型能够同时关注图像中的大物体和小细节。ASPP 由多个空洞卷积层组成，每个层具有不同的空洞率，通常包括一个没有空洞（普通卷积）的层。这些并行的空洞卷积层能够捕获不同尺度的特征，然后将它们合并，以提供更丰富的特征表示。因此，ASPP 能够提升模型对不同尺度物体的识别能力，尤其是在图像分割任务中。ASPP 的整体结构如图 2.5 所示。

条件随机场（Conditional Random Field, CRF）是一种统计模型，适合用于处理序列数据的标注任务。CRF 定义了一组条件概率分布，它考虑了观测数据（如图像分割中的特征图）和标注之间的复杂关系。CRF 的目的是找出最能解释观测数据的标注序列。在 CRF 的计算过程中，核心是确定和计算不同标注序列发生的概率。CFR 中使用的能量函数（Energy Function）计算公式如下：

$$E(y \mid x) = -\sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{k=1}^{K} \omega_k \delta(y_i - y_j, x_i - x_j) \tag{2.11}$$

其中，E 是能量函数，x 是观测数据，y 是标签序列，N 是序列长度，K 是状态（标签）数量，ω_k 是权重参数，δ 是核函数，通常用于计算两个标签之间的差异。

条件随机场概率表示给定观测序列 x 下标签序列 y 的条件概率分布[24]，计算公式如下：

$$P(y \mid x) = \frac{e^{-E(y|x)}}{Z(x)} \tag{2.12}$$

其中，$Z(x)$是归一化因子（也称为配分函数），确保所有可能的 y 的概率之和为 1。

配分函数 $Z(x)$用于归一化 CRF 模型，计算所有可能的标签序列的能量的指数和，计算公式如下：

$$Z(x) = \sum_{y} e^{-E(y|x)} \tag{2.13}$$

以上三种技术在图像分割任务中相互配合，空洞卷积和 ASPP 用于特征提取，而 CRF 用于后处理，共同提升了分割的准确度和稳定性。

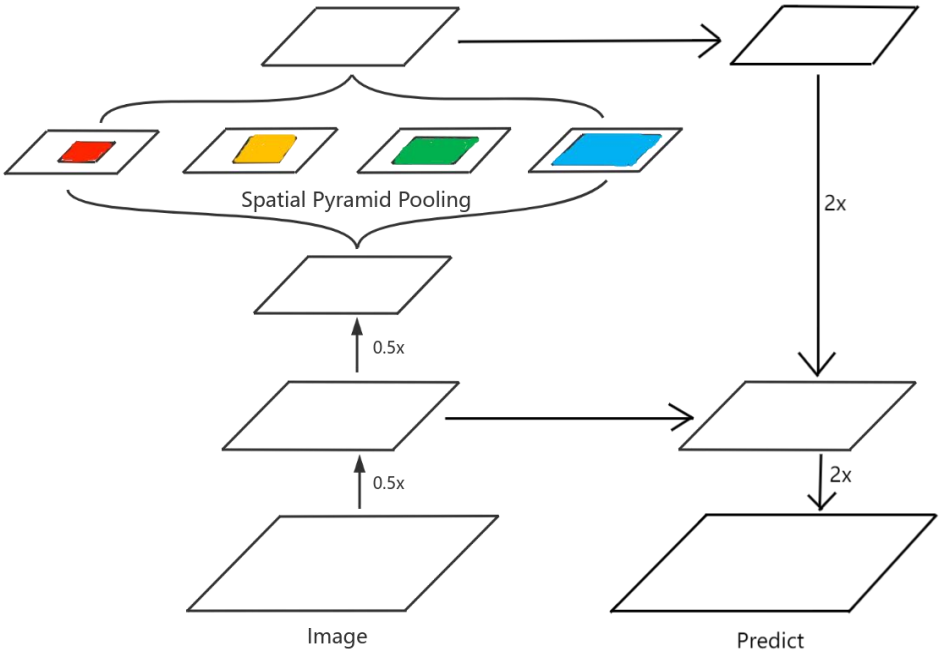


图 2.5 空间金字塔池化

DeepLab 通过 CRF 后处理和编码器-解码器架构，能更精确地定位对象边缘,并且 ASPP 结构使网络能够同时处理不同尺度的特征，提高了对多尺度物体的分割性能。除此之外，空洞卷积允许网络在不增加计算量的情况下捕获更广的上下文信息。因此，DeepLab 系列特别适合处理高分辨率的图像分割任务。以上是 DeepLab 系列的长处，以下是它们的局限或者说短处。虽然空洞卷积减少了计算量，但是 DeepLab v3+的编码器-解码器架构仍然需要较高的计算资源。

DeepLab 系列算法在进行需要非常精确的图像分割任务时表现出色，特别是在处理高清晰度的图片时。它们在多个领域都有应用，如自动驾驶领域（帮助自动驾驶车辆更准确地识别道路上的其他车辆和行人）。DeepLab 系列算法以其在图像分割任务中的高

性能和创新性，成为了计算机视觉领域的一个重要里程碑。

2.2.3 舌体分割模型选择

舌体分割的目标是从给定的舌部图像中准确地识别和定位舌头的区域，这样做是为了进一步的分析，如舌质、舌苔的颜色和形态分析等。分割任务需要高精度地识别舌头的轮廓，包括可能的不规则边缘；舌体分割算法应能处理不同个体、不同光照条件、不同成像设备和不同健康状况下的舌部图像。所以，本文的舌体分割任务需要把作为前景的舌体从原始图像的背景中分割出来，而原始图像的背景都是人脸、嘴唇、口腔、牙齿等部位，最关键的是，口腔内部、嘴唇和人脸的颜色与舌体十分相似，甚至有些舌体的颜色在光照强度较弱的情况下连人眼都难以区分出。因此，舌头的颜色可能与口腔内部的其他区域相似，使得区分舌头和背景变得困难，而且舌部图像可能存在模糊、光照不均、颜色变化等问题，这些都会影响分割的准确性。

为了解决上述问题，本文采用基于深度卷积神经网络的 DeepLab V3+模型。虽然 DeepLab V3+也采用了与 U-Net 一致的编码器-解码器架构。但是它具有的空洞卷积方法，可以扩大感受野，这使得网络能够在不同的尺度上捕捉特征，有助于处理不同大小的舌体。

2.3 Transformer 概述

Transformer 最初在 2017 年由 Google 的研究者在论文《Attention Is All You Need》中提出，主要用于解决自然语言处理（Natural Language Processing, NLP）中的序列到序列（seq2seq）问题，如机器翻译。随后，Transformer 架构也被成功应用于其他领域，包括图像处理和语音识别。

在 Transformer 模型问世之前，循环神经网络（Recurrent Neural Network, RNN）和长短期记忆网络（Long Short-Term Memory, LSTM）是处理序列数据的主流方法。但 hi 是这些模型存在一些局限，如处理长距离依赖的能力有限，以及在训练时难以难以实现有效的并行计算。为了克服这些限制，研究者提出了 Transformer 模型，它使用自注意力机制来捕捉序列内的长距离依赖关系[25]。以下为 Transformer 的组成部分与核心原理的概述。

2.3.1 自注意力层

自注意力层可使模型在处理序列时，不仅能“看到眼前的信息”，还能“留意”到序列中其他位置的信息。这样，不管这些元素之间距离多远，模型都能够理解序列中各个元素之间的关系。自注意力机制允许序列中的每个元素都直接与序列中的其他元素交互，而不是仅依赖于邻近的几个元素。简单来说，自注意力层让模型在处理序列时，能看得更远、听得更多，从而更好地理解整个序列的意思。

Transformer 中的自注意力机制，能让模型在处理一串序列时，是计算序列中每个元素对其他所有元素的关注度，然后模型会用这些注意力权重重新表示这个序列，简单理解为是把每个元素的重要性按照它对其他元素的关注度重新标出来。以下是自注意力机制的基本计算步骤和公式。

给定一个输入序列 X，其中 X 的维度是 $m \times d_{Model}$（m 是序列的长度，d_{Model} 是模型的维度），自注意力机制首先通过三个线性变换生成查询（Query, Q）、键（Key, K）和值（Value, V）[26]：

$$\begin{aligned} Q &= XW^{Q} \\ K &= XW^{K} \\ V &= XW^{V} \end{aligned} \tag{2.14}$$

其中，W^{Q}，W^{K}，W^{V} 是可学习的权重矩阵。接着，计算查询 Q 和所有键 K 之间的注意力分数，然后应用 Softmax 函数来获得权重：

$$Attention\_Scores = \frac{QK^{T}}{\sqrt{d_k}} \tag{2.15}$$

其中，K^{T} 表示 K 的转置，d_k 是键的维度，通常取 d_{Model}/h，其中 h 是注意力头的数量。除以 $\sqrt{d_k}$ 是一种缩放操作，有助于稳定梯度。

$$Attention\_Weights = Soft\max(Attention\_Scores) \tag{2.16}$$

最后，使用注意力权重来加权求和值 V，得到最终的自注意力输出：

$$Output = Attention\_Weights \cdot V \tag{2.17}$$

在多头注意力（Multi-Head Attention）的情况下，上述过程会被复制 h 次，每个头使用不同的权重矩阵 W^{Q}，W^{K}，W^{V}。每个头的输出然后被拼接起来，并通过另一个线

性变换来产生最终的输出：

$$Multi-Head\_Output = Concat(Head_1, Head_2, ..., Head_h)W^O \tag{2.18}$$

其中，*Concat* 表示将所有头的输出拼接起来，W^O 是最终的输出权重矩阵。

这样，自注意力机制能使模型感知到序列中每个元素对其他所有元素的影响，从而更好地理解整个序列的完整意思，这是 Transformer 架构能够有效处理序列数据的关键所在。

2.3.2 前馈神经网络

前馈神经网络（Feed-Forward Neural Network, FFNN）在自注意力层处理完序列信息后，对其输出的信息进一步提炼，帮助模型更深入地理解和表示序列数据。FFNN 是 Transformer 中不可或缺的一部分，它与自注意力层一起，为模型提供了强大的序列处理能力。

FFNN 在 Transformer 中首先接收输入的数据，然后这些数据会通过一个全连接层，用来从这些数据中找出重要的细节和特征。但是仅仅找出特征还不够，因为问题一般很复杂，需要模型能够理解和处理这种复杂性，这时候需要激活函数的加入。通过激活函数给模型加入非直线的、有弹性的思考能力，模型不仅能处理直线型问题，还能学习和理解更复杂的模式。通过这两个步骤，FFNN 能够帮助 Transformer 更深入地理解和处理输入的数据，使它在处理语言或者图像等任务时更加聪明和准确。

Transformer 中的 FFNN 通常与残差连接和层归一化结合使用。残差连接能使网络训练得更深，通过将 FFNN 的输入直接添加到输出上，有助于解决梯度消失问题。层归一化则对每个层次的输出都做一个归一化处理，让模型训练的过程更加稳定，同时也让模型在面对新数据时，表现得更好，即泛化能力。在 Transformer 中，FFNN 的参数是在开始训练之前随机设置的，然后在训练过程中，通过反向传播算法不断地调整这些参数，让模型越学越聪明。而且，FFNN 的输出和输入的大小是相同的，这能让它很轻易融入到 Transformer 的其他部分一起协同工作。

2.3.3 编码器和解码器

整个 Transformer 由多个相同的编码器层组成，解码器层则与编码器层结构相似，

但加入了对编码器输出的注意力机制，以实现信息的双向流动。编解码的流程如图 2.6 所示。

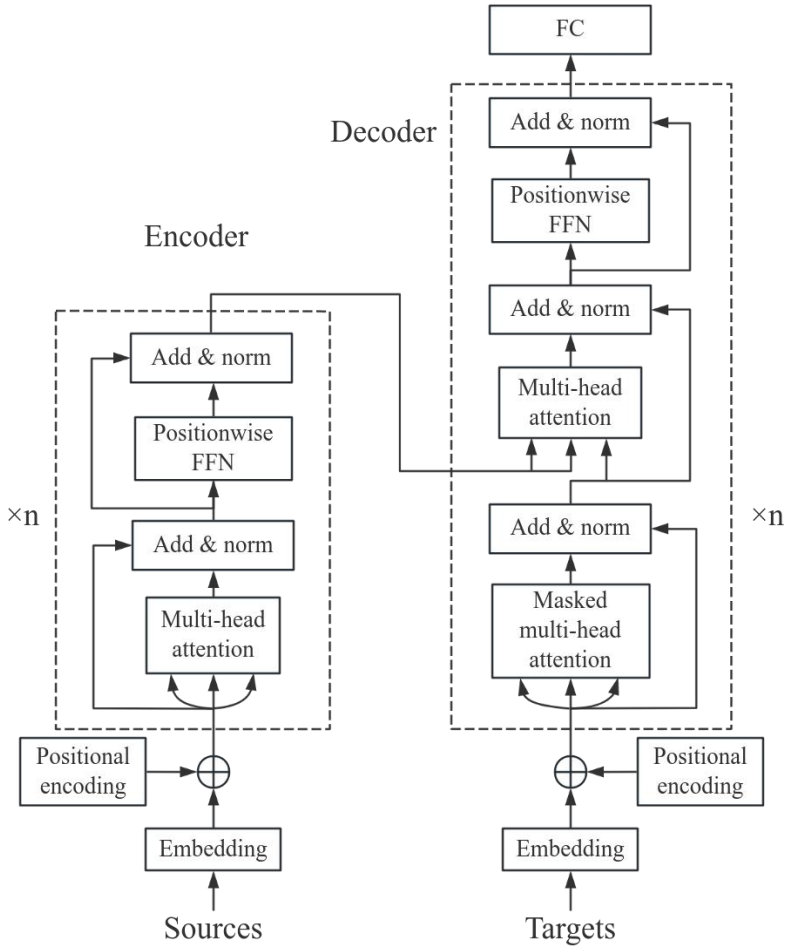


图 2.6 编解码流程

编码器部分的主要任务是将输入序列（如用一种语言写的句子），转换成一个连续的、能够包含整个句子主要意思和上下文的向量。编码器由很多层相同的结构堆叠而成，一般有 6 层。每一层都包含两个重要的部分：自注意力层和 FFNN。自注意力层让模型能够关注序列中的每个元素，并理解它们之间的关系。FFNN 则进一步处理这些信息，提取出更重要的特征。在这两层之间，模型会使用残差连接把每层的输入和输出加起来，同时模型还会进行层归一化。由于 Transformer 不像 RNN 那样可以记忆序列中每个元素的顺序，所以它需要一个额外的位置编码来告诉模型每个元素在序列中的位置，比如每个单词在句子中的位置。这样，模型就可以更好地理解序列中的上下文信息。

解码器部分的工作是把编码器理解的内容转换成另一个序列（如把一种语言翻译成另一种语言）。解码器也是由很多层相同的结构组成的，每一层里有三个重要的部分：自注意力层、编码器-解码器注意力层和 FFNN。

2.4 基于 Transformer 的图像分类模型分析

基于 Transformer 的图像分类模型，是利用 Transformer 架构来进行图像分类任务的深度学习模型。Transformer 最初是在自然语言处理领域提出的，用于处理序列数据，特别是在机器翻译任务中取得了巨大成功。近年来，Transformer 也被引入到计算机视觉领域，尤其是在图像分类任务中，展现了其强大的性能。

2.4.1 Vision Transformer

Vision Transformer（ViT）是由 Google 团队在 2020 年提出的一种图像分类模型，它将自然语言处理中广泛使用的 Transformer 架构成功应用于视觉领域[27]。此前，卷积神经网络在图像识别任务中一直占据主导地位，而 ViT 的提出标志着 Transformer 架构在计算机视觉领域的突破性应用。以下将简述 ViT 的核心原理和适用场景，其模型结构将在第四章 4.2 节详细说明。

ViT 的核心原理是将图像分割成多个小块（patches），然后将这些小块视为序列元素，送入 Transformer 的编码器进行处理。ViT 不依赖于卷积操作，完全基于自注意力机制来处理图像数据。

ViT 在多种视觉任务中展现出优异的性能，包括但不限于：

（1）图像分类：ViT 在大规模数据集上进行预训练，然后在较小的数据集上进行微调，以适应特定的图像分类任务。

（2）目标检测：ViT 也被用于目标检测任务，如 Facebook 提出的 DETR 算法，该算法完全基于 Transformer 进行端到端的目标检测。

中医舌诊需要对舌头整体进行全面细致观察。具体来讲，中医通过观察舌头的几个关键方面来进行诊断，包括检查舌质的质地、舌苔的状况，以及舌头颜色。舌质为舌头的肌肉部分，主要观察其颜色和形态；舌苔为覆盖在舌质上的一层薄薄的黏膜，观察其颜色、厚度、湿度等。舌色为舌头的色泽，可以反映身体的寒热状态。中医舌诊的这种全面性表明，在进行诊断时不能仅仅关注舌头的某一部分，而应该综合考虑舌头的整体状态[28]。

由于每个人的舌头状态和形态各异，颜色多变，因此在进行舌苔分类时，需要对整个舌头进行全面的诊断，而不能仅仅关注舌苔或舌色。否则，诊断结果可能会不够客观或出现误判。因此，本文选择使用 ViT 作为舌苔分类的模型。ViT 能够捕捉到舌苔图像的全局信息，这对于理解舌头图像的整体内容来说非常有用。

2.4.2 Swin Transformer

Swin Transformer 是由微软亚洲研究院设计出的一种新型 Transformer 模型，它主要是为了解决传统 Transformer 在处理视觉任务，比如分析图片时遇到的一些难题。这些难题包括处理分辨率特别高的图像、计算起来太复杂、以及怎样让模型更通用和能够适应不同大小的图像等。以下将简述 Swin Transformer 的核心原理、网络结构和适用场景。

Swin Transformer 的核心原理包括层次化特征表示、局部性与全局性的结合和高效计算。层次化特征表示指的是，通过逐层降低分辨率和增加模型的感受野，生成多尺度的特征表示；局部性与全局性的结合指的是，在局部窗口内计算自注意力[29]，同时通过移位操作保持全局信息的交互；高效的计算指的是通过限制自注意力在局部窗口内，显著降低了计算复杂度，使其与图像大小呈线性关系[30]。

Swin Transformer 的网络结构是一种层次化的设计，它结合了 Transformer 的自注意力机制和 CNN 的局部感知能力，特别适用于计算机视觉任务[31]。以下是 Swin Transformer 模型结构（如图 2.7 所示）的关键组成部分和特点：

（1）Patch Embedding：输入图像首先被分割成固定大小的小块（patches），这些小块被视为序列元素。在 Swin Transformer 中，每个 patch 代表图片中的一小块区域，而这个 patch 的特征，就是这个区域原始像素的 RGB 值的连接。通过一个线性嵌入层，将这些 patch 的特征投影到一个较高的维度空间。

（2）多层次结构（Stages）：Swin Transformer 模型包含多个阶段（Stage），每个阶段都会缩小输入特征图的分辨率，类似于 CNN 中的逐层抽象。在每个阶段，特征图的尺寸会减半，同时通道数翻倍，这样的层次化设计使得模型能够捕捉不同尺度的特征。

（3）Patch Merging：在除了第一个阶段之外的每个阶段开始时，都会使用 Patch Merging 操作。这个方法是把图片中紧邻的两个小块合并成一个更大的块，这样做的目的是让特征图的分辨率降低，也就是说，图片的细节程度变少了，但是能看到更宏观的特征。这种合并小块的方式有助于模型建立起一种层次化的理解，类似于先看整张图片的大结构，然后再慢慢关注细节。同时，因为图片的分辨率降低了，所以模型处理图片时需要的计算量也会减少，这样就能更快地完成计算，提高效率。

（4）Swin Transformer Blocks：每个阶段由多个 Swin Transformer Blocks 组成，这些 blocks 可以是标准的 Transformer Block，也可以是具有局部自注意力机制的变种。这些 blocks 让模型能够专注于图像的局部区域，使得模型在局部感受野内能捕捉到更多细

节和信息。

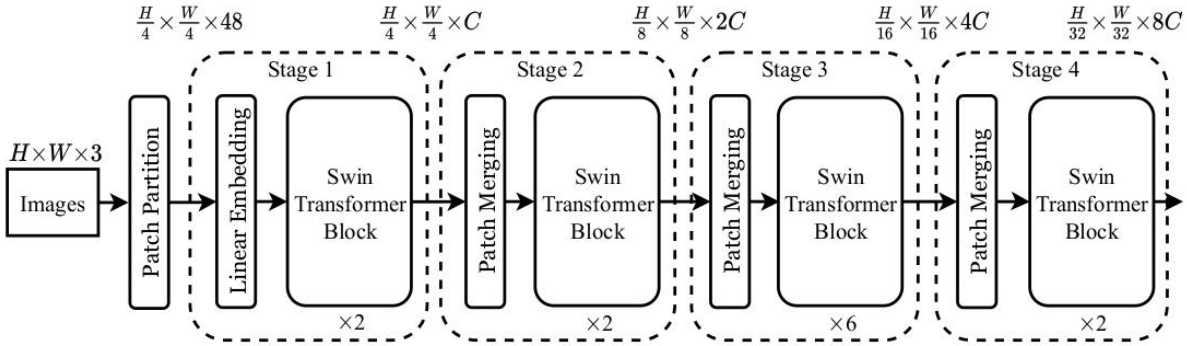


图 2.7 Swin Transformer 模型结构

Swin Transformer 适用于以下计算机视觉任务：

（1）图像分类：Swin Transformer 能够深入理解图片的内容，因为它可以捕捉到图片中不同层次的特征。

（2）目标检测和实例分割：Swin Transformer 能够融合不同尺度的特征，因此可以检测不同尺寸的目标。

（3）语义分割：Swin Transformer 的层次化特征表示有助于理解图像中每个区域的语义信息。

2.4.3 舌苔图像分类模型选择

在对比 ViT 和 Swin Transformer 两种模型后，本文选择 ViT 作为舌苔图像分类算法的原因如下：

（1）与 Swin Transformer 相比，尽管 Swin Transformer 通过局部感受野和层间移动窗口机制来提升全局信息的捕捉能力，ViT 通过自注意力机制能够直接捕捉图像的全局信息，这在舌苔图像分类中尤为重要，因为舌苔图像往往需要对图像的整体上下文有更深入的理解。

（2）ViT 比 Swin Transformer 更加高效，因为它不需要计算局部感受野或者层间的移动窗口，这样可以减少模型的参数数量和计算量。

（3）ViT 的自注意力机制让它在硬件上能够实现更快的训练和推理速度，这一点比 Swin Transformer 的层间移动窗口机制更容易进行并行处理。

（4）ViT 的模型结构比较简单和统一，这使得它更容易扩展到不同的任务和数据集上，而 Swin Transformer 的复杂性可能会在小数据集上导致过拟合。

综上所述，考虑到 ViT 在全局信息捕捉、参数效率、计算复杂度等优势，本文选择了 ViT 作为处理舌苔图像分类任务的算法。

2.5 深度学习框架概述与选择

深度学习框架是设计、训练和部署深度学习模型的关键工具。选择一个合适的框架对于提升开发效率、模型性能和代码的可维护性非常关键。以下是三种流行的深度学习框架的概述。

TensorFlow 是一个由 Google 开发的开源机器学习框架，适用于研究和生产环境。它拥有强大的社区支持和丰富的开源资源。TensorFlow 的灵活性体现在其数据流图的设计上，图中的节点代表操作，边代表数据流动。另外，TensorFlow 2.0 版本集成了 Keras，使得构建和训练模型变得更加简单直观。

Caffe 则专注于图像处理任务，它的设计在于模块化和可扩展性。Caffe 允许用户通过配置文件来定义模型架构，这种方式非常灵活，可以轻松添加新的层或者功能。Caffe 提供了多种语言的接口，包括 C++、Python 和 MATLAB，以及方便的命令行工具，这些都大大简化了模型的训练和部署工作。

PyTorch 是另一个流行的开源机器学习库，由 Facebook 的 AI 研究团队开发。PyTorch 的 API 与 Python 语言类似，易于上手，特别适合熟悉 Python 的开发者。它提供了高度的灵活性，适合自定义复杂网络架构，适合研究和原型设计。

对比以上三个深度学习框架后，考虑到 PyTorch 的 API 设计简洁、灵活性高、对 NVIDIA CUDA 的良好支持，故选择 PyTorch 作为实现 DeepLab V3+模型和 ViT 模型的框架。PyTorch 的这些特性为快速原型设计、实验和模型训练提供了便利，特别适合深度学习研究和开发。

2.6 舌苔 AI 导诊系统总体方案设计

本文研究的内容是利用图像处理技术来实现中医舌诊的智能化，主要工作具体分为图像预处理、舌体分割模型和舌苔分类模型的搭建。本文所提出的舌苔 AI 导诊系统的整体架构如图 2.8 所示，其中虚线框标注的部分为本文的核心部分——舌体分割和舌苔分类。

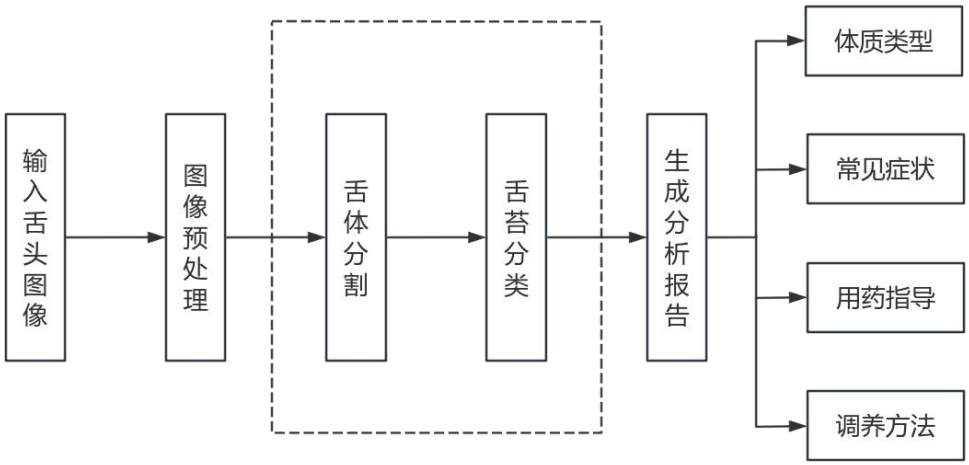


图 2.8 舌苔 AI 导诊系统的总体框架图

本文将创建两个专门的数据集：一个用于舌体分割的数据集，另一个用于舌苔分类的数据集。这两个数据集将分别用于训练 DeepLab V3+分割网络模型和 ViT 分类网络模型。DeepLab V3+能使舌苔 AI 导诊系统准确地从舌头图像中识别并分割出舌体，为舌苔分类提供准确的输入图像。最终，利用 ViT 模型，系统将根据舌面的特征信息进行分析和分类，从而预测出对应的中医体质特征，并提供相应的用药指导、体质分析报告，以辅助医生进行更准确的病情诊断。

2.7 本章小结

本章首先分别对 CNN 和 Transformer 进行了核心原理和网络结构的概述。随后对几种主流的基于 CNN 的图像分割模型和基于 Transformer 的分类模型进行了深入的分析和比较。最终选择 DeepLab V3+和 ViT 分别作为舌体分割模型和舌苔分类模型。还对几个流行的深度学习框架进行了综合评估，考量了它们各自的优势和局限，基于此选择 PyTorch 作为本文的深度学习框架。最后给出了本文的总体设计方案，大致介绍了系统的运行流程。

第三章 利用 DeepLab V3+进行舌体分割

3.1 舌体分割数据集与数据预处理

医学图像数据一般都难以获取，其设计个人健康信息，属于敏感数据并且医学图像的收集和使用通常需要通过伦理审查，可能存在伦理上的限制，导致数据集不能随意分发，所以很多论文作者为了保护患者个人隐私，数据集都不会公开分享。舌头图像属于医学图像数据的一部分，因此从以往研究者获取几乎是不可能的，只能从公共数据集去搜寻。不可回避的是，高质量的舌头图像数据集非常稀缺，数据集的研究者可能需要投入了大量时间和资源来收集和标注。经过漫长的搜寻，本文最终从飞桨 AI Studio 星河社区的公共数据集中获取了 1472 张原始舌头图像，图像文件概览如图 3.1 所示，包含了五种中医体质类别，即 Mirror-Approximated、Thin-White、White-Greasy、Yellow-Greasy、Grey-Black（镜像近似、薄白、白油腻、黄油腻、灰黑）。

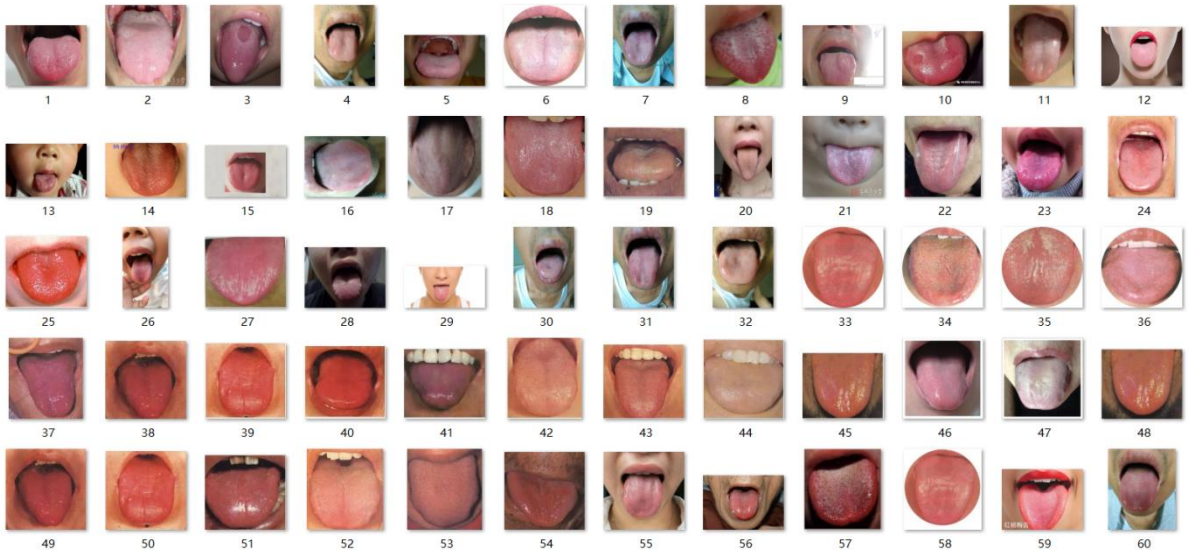


图 3.1 部分原始舌头图像概览图

舌体分割数据集首先需要对原始舌头图像中的舌体进行标注。本文使用 Labelme 数据标注工具进行快速准确的舌体标注，将整个舌头部位从背景中框选出来。在进行图像分割任务时，原始输入图像应该被调整到模型预期的输入尺寸。DeepLab V3+通常接受高和宽都是 224 或 256 像素的图像。标签为 JSON 格式文件，需要将 JSON 格式转换成二值掩码图像。标签图中的像素值是整数，表示不同的类别，舌体被标记为 255——白色，而背景是 0——黑色，标注图和二值掩码图如图 3.2 所示。标签图通常保存为 PNG 格式，因为 PNG 格式支持保存标签值而不丢失信息。

(a)标注图　　(b)二值掩码图

图 3.2 标注图和二值掩码图

使用上述方法，本文对公开获取的 1472 张舌部图像进行了逐一的精确标注，创建了相应的二值掩码图像，从而完成了舌体分割数据集的标注工作，如图 3.3 所示。

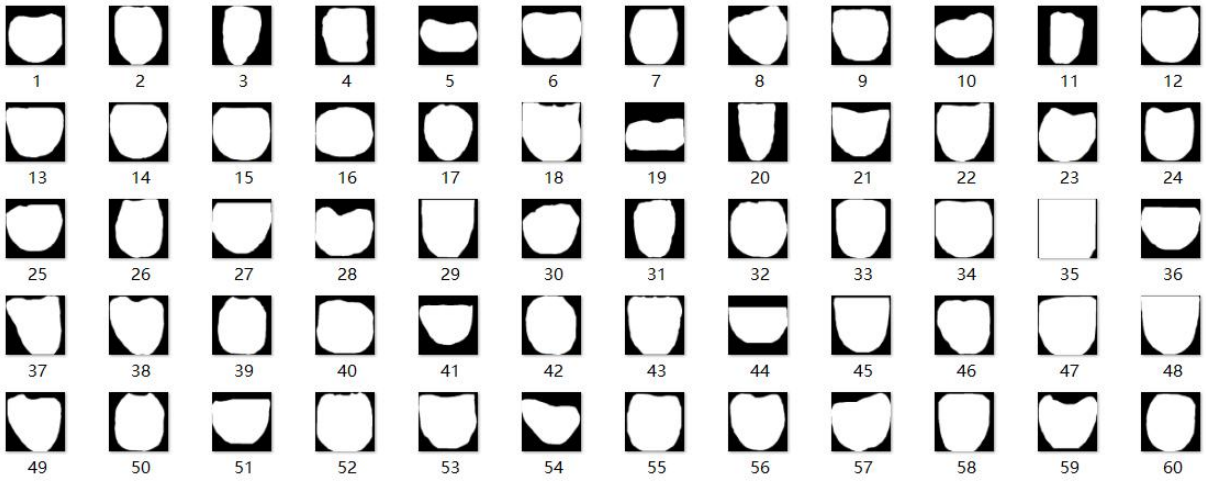


图 3.3 舌体分割数据集标签部分

进行 DeepLab V3+模型训练前，需要进行数据预处理。原始图像尺寸不一，而模型要求输入尺寸一致，因此需要将原始图像使用 OpenCV 开源计算机视觉库进行图像处理。利用库函数对所有原始图像进行不失真的重新调整尺寸操作——resize。具体而言是根据指定的输入尺寸比例——256×256，计算需要在图像的上下左右各填充多少像素，以使填充后的图像尺寸与指定的尺寸成比例。最后，应用边界填充，然后调整图像大小以匹配网络的输入尺寸，如图 3.4 所示。

(a)原始图像　　(b)边界填充后的图像

图 3.4 原始图像和边界填充后的图像

3.2 迁移学习

CNN 进行特征学习时需要依赖大量的数据样本来优化网络。然而，由于本文中舌体分割数据集样本量有限，直接训练可能会导致模型出现过拟合现象，进而影响模型的泛化性能[32]。为了解决这一问题，本文采取了迁移学习策略，利用已有的知识或者模型，帮助模型更快地学会新的知识。该策略不仅有助于减轻过拟合问题，还能加速模型的训练进程[33]。

迁移学习可以通过两种主要方式实施：首先是微调，这涉及到对预先训练好的模型进行细微的参数调整，以便其适应新的应用场景。其次是冻结训练，即先保留模型前面几层的设置不变，只对后面几层做一些训练，让模型逐渐适应新的任务。

本文所用的 DeepLab V3+舌体分割模型使用了原始模型的主干结构和参数设置，这为应用迁移学习策略提供了便利。由于不同图像间存在共同的基本视觉元素，如边缘、颜色和纹理等，这些共性特征可以通过网络的早期卷积层来捕获，从而使得迁移学习在舌体分割任务中成为可能。

本文迁移学习的具体过程包括：第一阶段，冻结预训练模型主干部分，即特征提取网络，此时模型的主干被冻结，特征提取网络的参数在训练时不发生改变，随即进行 50 epochs 的训练和验证过程，该过程占用的显存较小，仅对网络进行微调[34]；第二阶段，解冻阶段，即把进行了冻结训练的预训练模型的主干部分进行解除冻结操作，此时模型的主干解冻，特征提取网络的参数在训练时会发生改变[35]，随即进行另外 50 epochs 的训练和验证过程，该过程占用的显存较大，网络所有的参数都会发生改变。迁移学习的流程如图 3.5、图 3.6 所示。

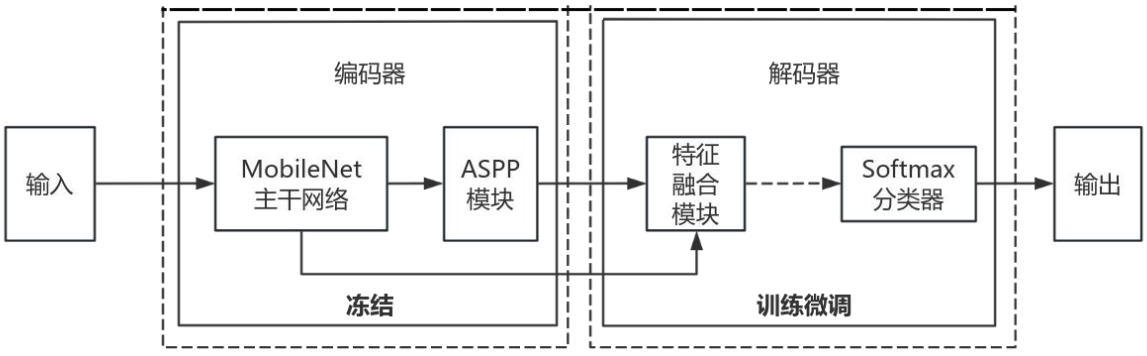


图 3.5 前 50 epochs 的训练过程

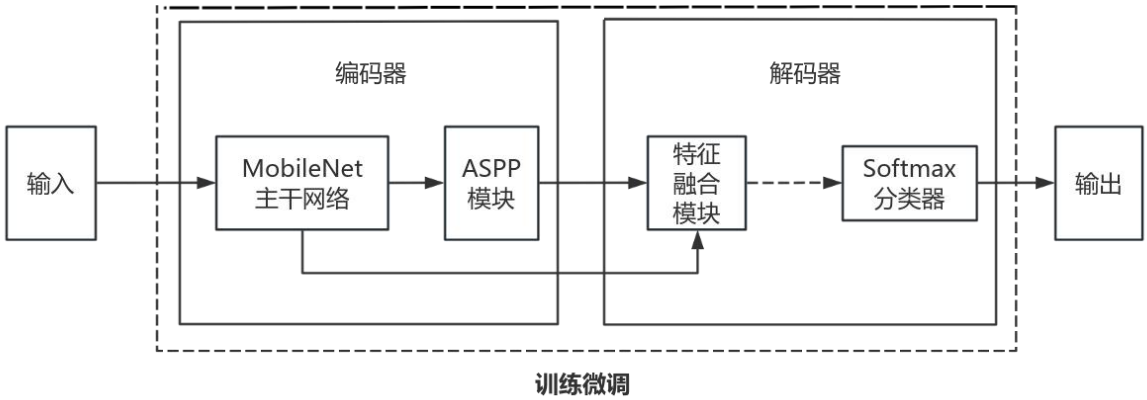


图 3.6 后 50 epochs 的训练过程

3.3 DeepLab V3+模型结构

舌体分割任务目的是，从包含复杂背景如面部、口腔和牙齿的图像中精确地区分出舌体区域。实现精确分割的核心挑战在于能否有效地捕捉到区分舌体与背景的特征。鉴于卷积神经网络在特征提取方面相较于传统方法所表现出的显著优势，本文采用基于卷积神经网络的 DeepLab V3+进行舌体分割，其模型结构如图 3.7 所示。

3.3.1 编码器

在编码器部分，主要包括了主干网络（MobileNet V2）和 ASPP 两部分。MobileNet V2 从输入图片中提取出有用的特征，为后面的分析做准备。ASPP 模块紧跟在 MobileNet V2 后面，其使用一个 1×1 的卷积、三个 3×3 的空洞卷积和一个全局池化来对 MobileNet V2 的输出进行处理。然后再将其结果都连接起来并用一个 1×1 的卷积 来缩减通道数。具体如下：

（1）主干网络把图像的重要特征分为两部分：一部分是图像的高级特征，这部分会送到 ASPP 模块进行进一步处理；另一部分是图像的低级特征，这部分会直接送到解码器模块。

（2）ASPP 模块接收到主干网络的高级特征后，会用四种不同膨胀率的空洞卷积和一个全局平均池化来处理这些特征，得到五组不同的特征图。然后，这些特征图会被拼接在一起，再通过一个筛选器（1×1 卷积块，包括卷积、批量归一化、激活和 Dropout 层）进行筛选，最终形成一组更加精准的特征图，这些特征图会被送到解码器模块进行下一步处理。

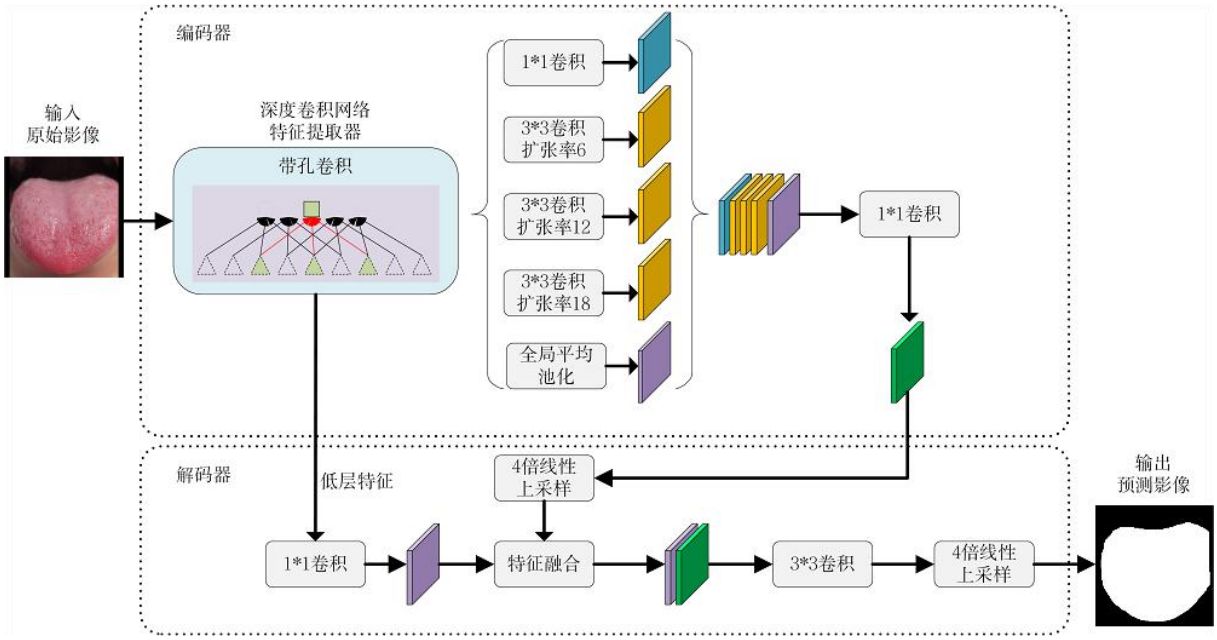


图 3.7 DeepLab V3+模型结构

空洞卷积的优点如下：

（1）与传统卷积操作相比，空洞卷积显著降低了所需的计算量。其计算量可降至普通卷积的九分之一，这主要得益于空洞卷积在不增加额外的参数来增大感受野。

（2）为了增强神经网络对输入图像的感知范围，网络的深度被增加，从而扩展了单个像素的感受野。然而，这种深度的增加导致了特征图的空间尺寸减小，进而降低了网络的空间分辨率。为了解决这一问题，引入了空洞卷积技术。空洞卷积通过扩大卷积核的覆盖范围，有效地增加了网络的感受野，使其能够检测和分割较大的目标对象。

（3）为了捕获多尺度的上下文信息，空洞卷积在卷积核的权重之间插入特定的零填充。对于一个给定的卷积核，其连续的权重值之间会根据设定的参数 $r-1$（其中 r 为空洞率或膨胀率）插入相应数量的零。通过这种方式，感受野在不增加额外参数的前提下被有效地扩大。较小的 r 值适用于捕获图像的局部细节，而较大的 r 值则有助于网络从更广阔的范围内提取信息，从而实现对更大尺度结构的识别。

3.3.2 主干网络

本文的 DeepLab V3+采用了 MobileNet V2 作为其主干网络（Backbone），该网络相比于 Xception 的参数数量较少，且在保持可接受精度的同时，能够以更快的速度运行，这对于本文构建的基于 PyQt5 的舌苔 AI 导诊系统程序非常有利。一个简化的 MobileNet

V2 网络结构如图 3.8 所示。

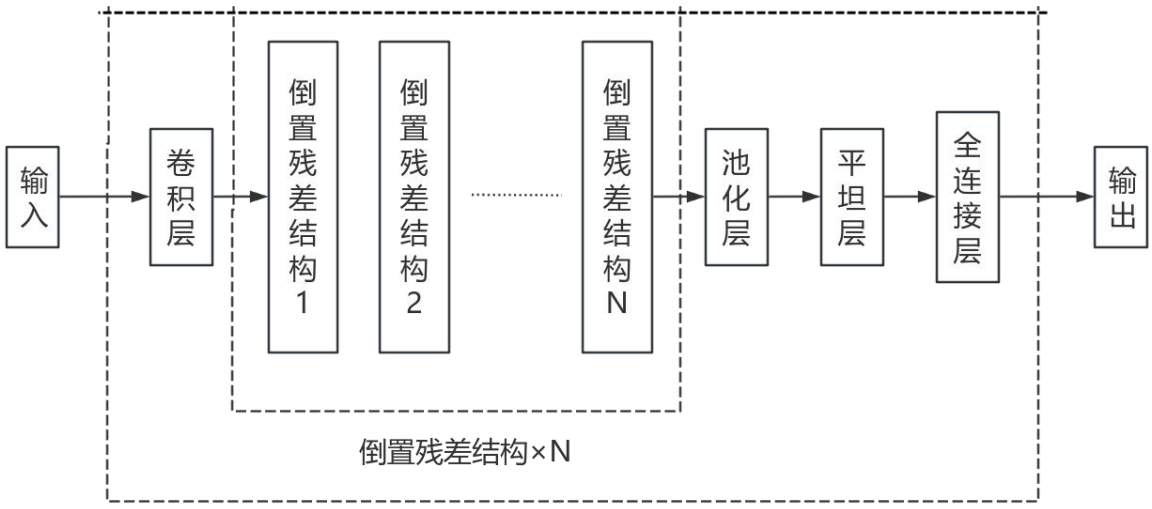


图 3.8 简化的 MobileNet V2 网络结构

以下是 MobileNet V2 的主要网络结构特点：

（1）倒置残差结构（Inverted Residual Blocks）：MobileNet V2 使用了一系列倒置残差结构，这些结构首先通过深度可分离卷积（Depthwise Convolution followed by a Pointwise Convolution）来扩展输入的通道数，然后通过一系列的相同或减少通道数的深度可分离卷积层来进一步提取特征。

（2）线性瓶颈（Linear Bottlenecks）：在倒置残差结构中，网络首先通过 1×1 的点积卷积（Pointwise Convolution）减少通道数，形成瓶颈，然后再通过深度卷积（Depthwise Convolution）和点积卷积进一步处理特征。

（3）扩展比（Expansion Ratio）：在倒置残差结构中，扩展比控制了点积卷积后中间特征图的宽度。如果扩展比是 2，那么输出的通道数将是输入通道数的两倍。扩展比越大，中间特征图的通道数越多，模型的参数也越多。

（4）深度可分离卷积（Depthwise Separable Convolution）：它将传统的卷积操作分解成深度卷积和点积卷积。在深度卷积中，每个输入通道使用一个独立的滤波器进行卷积操作。点积卷积将深度卷积的输出作为输入，使用 1×1 的滤波器进行卷积。1×1 卷积的主要作用是将深度卷积的输出通道进行组合和重组，以产生最终的输出特征图。

（5）网络架构：MobileNet V2 的架构通常开始于一个或多个标准的卷积层，然后是一系列倒置残差结构，最后是池化层、平坦层和全连接层。

3.3.3 解码器

解码器部分，处理的是两组特征图：一组是从主干网络中间层得到的低级特征图，另一组是从 ASPP 模块得到的高级特征图。首先，用 1×1 的卷积对低级特征图进行处理，把通道数从 256 减少到 48。然后，对 ASPP 的输出特征图进行插值上采样，使它的尺寸与低级特征图的相同。接着，把经过通道降维的低级特征图和上采样后的 ASPP 特征图合在一起。这两张图一张提供了粗糙的结构信息，一张提供了细节信息，合在一起能得到更完整的信息。然后，用 3×3 的卷积块对拼接后的大特征图进行处理。最后，再次进行上采样，让特征图的尺寸和原始的输入图片一样大。这样，得到了最终的预测图。

3.4 DeepLab V3+舌体分割实验结果与分析

3.4.1 实验环境配置

支持本章实验运行的环境主要由硬件和软件两部分组成，硬件使用了 Intel(R) Xeon(R) Platinum 8352V 的 CPU 和 RTX 4090 的 GPU，软件使用了 cuDNN、CUDA，在 Python 的基础上使用了 PyTorch 作为深度学习框架来构建网络模型。详细的配置和环境如表 3.1 和表 3.2 所示。

实验在 AutoDL 算力云远程服务器的 ubuntu 系统上进行。AutoDL 是一个提供 GPU 云计算资源的平台，允许用户租用 GPU 服务器进行深度学习任务，它具有以下优点：

（1）用户可以根据需要租用不同配置的 GPU 服务器，而无需购买和维护物理硬件。

（2）它提供多种 GPU 型号和计算能力的选择，适应不同的计算需求。

（3）GPU 服务器预装了流行的深度学习框架和库，方便用户直接开始工作。

本文所有脚本和代码的编程语言均为 Python，集成开发环境（IDE）选用 PyCharm 专业版。因为 PyCharm 专业版提供对远程开发的支持。通过 SSH（Secure Shell）连接，可以将 PyCharm 作为远程开发环境，连接到一个运行在另一台计算机上的服务器。本文使用 PyCharm 专业版通过 SSH 进行远程开发的步骤如下：

（1）首先需要在 PyCharm 中配置一个 SSH 连接，这包括主机名（或 IP 地址）、用户名和 SSH 密钥。

（2）然后设置一个远程 Python 解释器，PyCharm 会将项目文件同步到远程服务器，并在远程服务器上运行代码。

（3）接着使用 Filezilla 软件，通过 SFTP（SSH File Transfer Protocol）进行个人 PC 和远程服务器之间的文件传输。

（4）最后在 PyCharm 上进行调试，它会在远程环境中运行程序，并在本地 IDE 中显示调试信息。

表 3.1 计算机硬件配置

硬件名称	型号及容量
GPU	RTX 4090
CPU	16 vCPU Intel(R) Xeon(R) Platinum 8352V
显存	24GB
内存	120GB

表 3.2 DeepLab V3+舌体分割实验软件环境

软件名称	软件版本
操作系统	ubuntu 20.04
Anaconda	Anaconda 3.21.8
Python 解释器	Python 3.8
集成开发环境（IDE）	PyCharm 专业版 2023.3.5
CUDA	CUDA 11.8
cuDNN	cuDNN 8.9.0
深度学习框架	PyTorch 2.0.0

CUDA（Compute Unified Device Architecture）是由 NVIDIA 公司推出的一种并行计算平台和编程模型，它可以使程序利用 GPU 的大量核心进行并行处理，这可以显著加快深度学习的计算速度。cuDNN（CUDA Deep Neural Network library）是一个用于深度学习的 GPU 加速库，它是 NVIDIA CUDA 平台的一部分，专门针对深度神经网络的高效训练和推理进行了优化。

3.4.2 舌体分割模型评价指标

像素精度（Pixel Accuracy, PA）、平均交并比（Mean Intersection over Union, MIoU）和帧率（Frames per Second, FPS）是评估图像分割模型性能的重要指标。本文采用 MIoU

作为舌体分割任务的评价指标。

像素精度用来衡量分割结果中每个像素被正确分类的比例[36]。像素精度的计算公式如下：

$$PA = \frac{\sum_{i=0}^{k} p_{ii}}{\sum_{i=0}^{k}\sum_{j=0}^{k} p_{ij}} \tag{3.1}$$

其中，k 为前景的类别数，p_{ii} 为正确分类的像素数，p_{ij} 为第 i 类被预测为第 j 类的像素数。像素精度是一个有用的度量，但通常还会结合其他指标，如交并比或平均交并比。

交并比（Intersection over Union, IoU）用来衡量预测框和实际标注框的重叠程度，通常用于衡量分割结果的好坏。对于一个特定的类别，IoU 计算公式如下：

$$IoU = \frac{\sum_{i=0}^{k} p_{ii}}{\sum_{i=0}^{k}(\sum_{j=0}^{k} p_{ji} + \sum_{j=0}^{k} p_{ij} - p_{ii})} \tag{3.2}$$

其中，p_{ji} 是第 j 类被预测为第 i 类的像素数。平均交并比则是计算所有类别的 IoU 后取平均值得到的，它为每个类别分别计算 IoU 值，然后求这些 IoU 值的平均数，其计算公式如下：

$$MIoU = \frac{1}{N}\sum_{i=1}^{N} IoU_i \tag{3.3}$$

其中，N 是类别总数，IoU_i 是第 i 个类别的 IoU 值。MIoU 越高，表示分割模型的性能越好，因此本文用它来作为舌体分割任务的评价指标。

帧率用于衡量视频播放或实时视频分析的速度，它表示每秒钟可以处理和显示的帧数，其计算公式如下：

$$FPS = \frac{Frames}{Second} \tag{3.4}$$

其中，*Frames* 表示总帧数，*Second* 表示总秒数。

3.4.3 实验结果分析

本文的舌体分割实验选择 MobileNet V2 作为主干网络，其轻量化设计优于 Xception 的网络结构，并且也能维持较高的准确率。经过前 50 epochs 的冻结和后 50 epochs 的训练微调策略，得到图 3.9 的训练、验证损失曲线和图 3.10 的 MIoU 曲线。

在训练过程中，训练集和验证集的损失函数在前十个训练批次中均呈现出迅速下降

的趋势，实际最终分别降到了 0.6 和 0.8 左右，这表明模型快速地收敛于一个最优解。这表明了所选模型架构和超参数对于舌体分割是十分有效的。

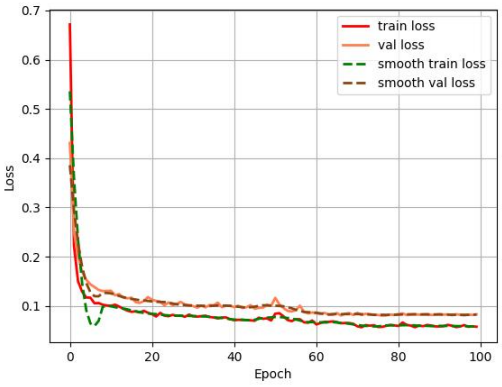


图 3.9 训练、验证损失曲线

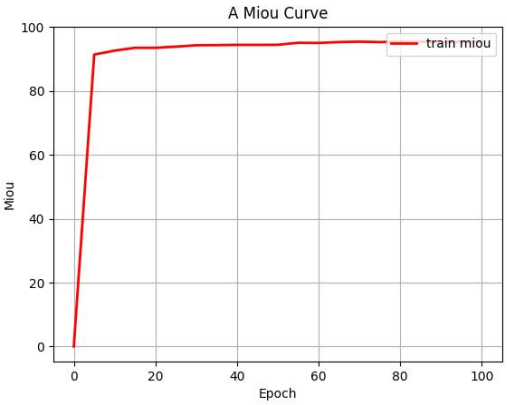


图 3.10 MIoU 曲线

同时，MIoU 指标在前十个 epoch 中也迅速上升，说明模型在提升划分舌体前景像素和背景像素的性能。随着训练的进行，训练和验证损失曲线最终趋于稳定，这说明模型已达到稳态，进一步的训练迭代不会带来显著的改进。同样，MIoU 在达到峰值 92%后也随之稳定，这表明模型已经有效地学习到了舌体和背景之间的差异，并且具有较高的保真度。以上分析可以表明，模型对未见数据有十分优秀的泛化能力。

实验抽取了一张舌头图片进行分割预测，结果如图 3.11 所示。

(a)原始舌头图像　　(b)分割后的图像

图 3.11 左图为原始舌头图像，右图为分割后的图像

经肉眼对于两张图像的对比，可以明显得知 DeepLab V3+的分割效果非常出色。尽管舌体与嘴唇、脸部颜色十分相似，甚至边缘部分融合进其他部分中，但是模型仍然几乎完整地分割出了舌体。原始图像与分割后图像的直观对比表明，DeepLab V3+在保持边缘细节的同时，准确地区分了各个语义区域，实现了对舌体的精确识别与定位。综上所述，DeepLab V3+满足了本文对高准确度舌体图像分割的要求。

3.4.4 DeepLab V3+与 U-Net 分割性能对比

本文实验还对原始舌头图像进行了分割实验对比，比较了 U-Net 和 DeepLab V3+两种先进的深度学习模型。实验结果表明，虽然 U-Net 在图像分割领域性能良好，但 DeepLab V3+在舌体图像分割任务中的表现更为优秀。实验结果如图 3.12 所示。

(a)原始图像　　(b)U-Net 分割图像　　(c)DeepLab V3+分割图像

图 3.12 实验结果图

通过对两种模型分割结果的细致对比分析，DeepLab V3+在舌体边缘细节的保留、噪声抑制以及整体分割精度上均展现出了显著的优势。特别是在舌体轮廓和舌脸交界处的处理上，DeepLab V3+采用的深度可分离卷积和特征融合技术，有效地提升了分割的准确性和可靠性。表 3.3 为定量评估指标如 MIoU、PA 和 FPS，综合验证了 DeepLab V3+在分割质量上的优势。

表 3.3 DeepLab V3+和 U-Net 分割性能对比

模型	MIoU	PA	FPS
DeepLab V3+	91.6%	92.7%	25.1
U-Net	87.3%	88.7%	22.4

综上所述，DeepLab V3+在舌体图像分割的应用中，不仅在视觉直观上提供了更为精细的分割效果，而且在定量分析上也达到了更高的评价标准，满足了本文对高精准度分割效果的严格要求。因此，DeepLab V3+被认定为更适合于本文中舌体图像分割的模型选择。

3.5 本章小结

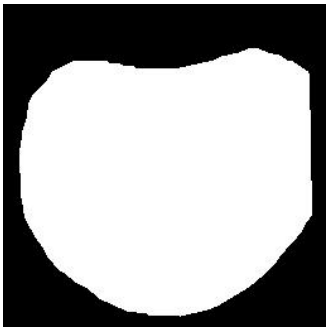
本章通过对 DeepLab V3+舌体分割实验的全过程和结果分析，以及与 U-Net 的分割性能对比，明确指出了 DeepLab V3+作为舌苔 AI 导诊系统舌体分割模型的优越性能。首先，本章详细说明了舌头图像数据的收集和数据预处理；其次，介绍了舌体分割模型的训练策略；然后，分析了 DeepLab V3+的网络结构，指出其先进的深度可分离卷积和

特征融合策略对于图像分割任务具有明显优势；最后，通过评估 DeepLab V3+和 U-Net 两种图像分割模型在舌体分割任务中的效果，进一步有效地证明了 DeepLab V3+是本次舌体分割实验最适配的模型。

第四章 利用 ViT 进行舌苔分类

4.1 舌苔分类数据集与数据预处理

在本文中，为了构建一个精确的舌苔分类数据集。首先，利用 Labelme 工具对舌体图像进行了细致的像素级分割。Labelme 将标注数据导出为 JSON 格式，其中包含了舌体区域的边界坐标信息及类别标签。随后，实施了一系列数据处理步骤，将 JSON 文件转换为掩码图像（Mask）。具体而言，编写了解析算法，提取 JSON 中的舌体轮廓坐标，并根据这些坐标在图像上生成了一个二值化掩码，其中舌体区域被标记为 255，而非舌体区域则标记为 0。最终，为了得到一个具有黑色背景的舌体图像，本文进一步处理了掩码图像。通过将原始舌体图像与掩码进行元素-wise 乘法操作，将舌体区域从原始图像中分离出来，并将背景替换为黑色。这一操作保留了舌体的原始颜色和纹理信息，同时为后续的舌苔分类任务提供了无干扰的舌体图像数据集。黑白掩码图像和黑色背景舌体图像如图 4.1 所示。



(a)黑白掩码图像

(b)黑色背景舌体图像

图 4.1 黑白掩码图像和黑色背景舌体图像

在进行舌苔图像的分类前，本文先剔除原始数据集中 Mirror-Approximated 类别，因为该类别数据量少且很多图像都与 Thin-White 类别重复，若不剔除，将会大大影响模型的拟合效果。其次，对剩余的 1360 张图像采用一系列的图像预处理技术，旨在增强图像特征，提高后续 ViT 模型的泛化能力和分类精度。部分图像经过包括图像翻转、亮度调整、饱和度调整、对比度调整以及锐度调节等预处理操作，生成了多组增强图像。在处理舌头图像数据集时，预处理步骤需要特别小心，因为舌苔分类对舌色和苔色的细节非常敏感。例如，对比度调整可以增强图像中明暗区域之间的差异。然而，在舌苔图像中，对比度的大幅度调整可能导致舌色和苔色的细微差别丢失。还有，亮度调整会影响整个图像的明暗程度。因此需要对舌头图像进行亮度、饱和度等微调。

首先，对原始图像进行左右翻转，生成镜像图像，以增强模型对图像左右翻转不变性的鲁棒性。随后，通过亮度调节操作，对原始图像进行亮度增强和减弱处理，包括提升至原来的 1.1 倍和 1.2 倍以及降低至原来的 0.9 倍和 0.8 倍，模拟不同光照条件下的图像。此外，利用饱和度调整技术，对图像的色度进行增强和减弱，包括增强至 1.1 倍和 1.2 倍以及减弱至 0.9 倍和 0.8 倍，以提升模型对色彩变化的识别能力。对比度调整则通过提高或降低图像的对比度，包括提高至 1.1 倍和 1.2 倍以及降低至 0.9 倍和 0.8 倍，使得图像中的特征更加明显，有助于模型更好地区分不同舌苔类别。最后，通过锐度调节，包括提高至 1.1 倍和 1.2 倍以及降低至 0.9 倍和 0.8 倍，提高模型对边缘信息细节的识别能力。原始图像和预处理后的图像如图 4.2 所示。

图 4.2 原始图像和预处理后的图像

通过上述增强策略，原始数据集从 1360 张图像扩充至 2800 张。最终扩充后的各类舌苔数据统计如表 4.1 所示。这一扩充增加了数据集的多样性，为模型训练提供了更为丰富的样本。在本文中，图像增强的参数选择主要基于经验判断，模拟了不同拍摄条件下的图像特征，以及确保图像增强后的数据能够有效提升模型性能，同时避免过拟合。所有预处理操作均通过 Python 编程实现，利用 PIL 库中的 Image 及其子模块 ImageEnhance 进行操作。预处理后的图像数据集为 ViT 模型的训练和验证提供了丰富的数据，为实现精准的舌苔分类奠定了坚实的基础。

表 4.1 各类舌苔数据统计

舌苔类别	属于该类别的图像数量
Thin-White	707
White-Greasy	699
Yellow-Greasy	700
Grey-Black	702
	共计 2808

4.2 图像输入 ViT 的整体流程与 ViT 模型结构

传统的 Transformer 结构主要用于处理自然语言，也就是人们说的话和写的文字。在处理语言时，它会用一种特殊的方式，叫做“词向量”，来表示每个词或句子词向量与传统图像数据的区别在于，词向量通常是一维向量按顺序排列，而图像则是像素排列成行和列。多头注意力机制在处理一维词向量的堆叠时会提取词向量之间的联系也就是上下文语义，这使得 Transformer 在自然语言处理领域非常好用，而二维图像矩阵如何与一维词向量进行转化就成为了 Transformer 进军图像处理领域的一个小门槛。

在 ViT 模型中，处理图像的方式是将输入图像分割成许多小块，这些小块在 ViT 中被称为“patch”。具体来说，就是把每个通道分成 16×16 的小格子，每个格子就是一个 patch。这个过程可以通过卷积操作来自动完成，也可以手动进行划分，但使用卷积操作的好处是它不仅可以完成划分，还能同时对数据进行预处理。以一个 224×224 像素的图像为例。

首先，通过卷积操作，图像会被划分成 16×16 个 patch。由于卷积操作通常会有一定的边缘效应，所以实际上每个 patch 的大小是 14×14 像素，而不是 16×16。这样，ViT 就可以把这些小的图像块当作序列数据来处理，利用 Transformer 模型强大的处理序列的能力来进行图像识别和分类的任务。

接下来，每个 patch 被“展平”成一个一维向量。二维的像素矩阵被转换成了一个长的一维列表。这些一维向量随后被堆叠起来，模拟了自然语言处理中词向量的操作。在这个例子中，每个 14×14 的 patch 转换成了一个长度为 196 的一维向量。通过以上方式，ViT 模型能够将图像数据转换成一系列一维向量，这些向量可以被送入 Transformer

模型进行处理。这种方法允许模型利用其自注意力机制来理解图像中不同区域之间的关系，从而进行有效的图像识别和分类。

之后，在每个 patch 的一维向量前，加入一个类别编码（class_embedding），这个编码通常是向量的第一位。这个类别编码是一个可学习的参数，它的值在模型训练过程中会不断调整。加入类别编码后，原本 196 维的向量变成了 197 维。类别编码的作用是帮助模型区分不同的 patches 属于同一个图像的不同部分，并且最终模型会使用这个编码的第一个维度来确定图像的分类。

除了类别编码，每个 patch 的向量还会加入位置编码（pos_embedding），这是另一组可学习的参数。位置编码为模型提供了关于 patches 在原始图像中相对位置的信息。位置编码的作用类似于在全连接网络或卷积网络中加入的偏置项，它帮助模型理解数据的空间结构。

通过加入位置编码，每个 patch 的向量长度变为 197 维，模型最终会基于这些 197 维的向量来进行图像的分类。以上图像输入 ViT 的整体流程如图 4.3 所示。

总之，整个图像输入 ViT 的流程通过这些步骤，将原始的二维图像数据转换成一系列一维向量，这些向量包含了图像的空间和类别信息，为后续的 Transformer 模型处理提供了基础。

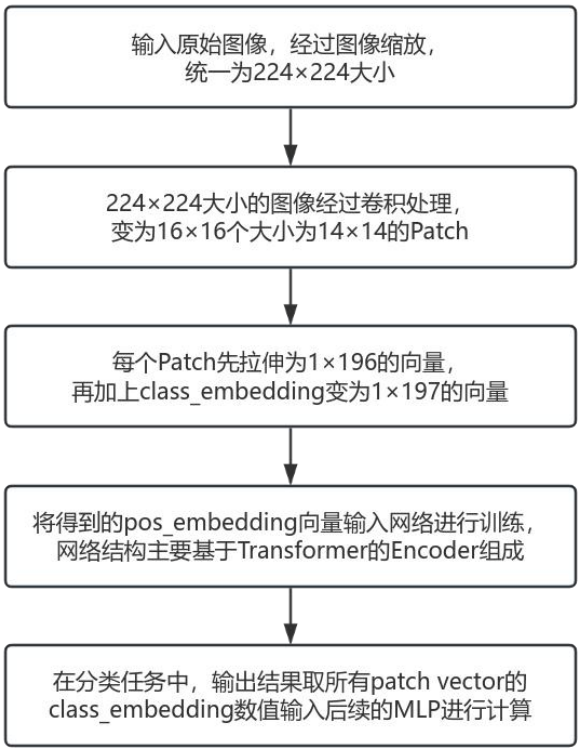


图 4.3 图像输入 ViT 整体流程

ViT 的模型结构（如图 4.4 所示）分为以下几个主要部分：

（1）图像块嵌入（Patch Embeddings）：ViT 模型的第一步是将输入图像划分成等大小的小块，比如 16×16 像素的方块。然后，每个图像块被转换成一个固定维度的向量，这个过程称为嵌入，它为后续的 Transformer 编码器提供了序列化的数据表示。

（2）位置嵌入（Positional Embeddings）：因为 Transformer 模型本身不包含任何关于序列中元素顺序的信息，ViT 通过添加位置嵌入来补充这一部分缺失的信息。位置嵌入为每个图像块提供了其在原始图像中位置的表示。

（3）Transformer 编码器：这是 ViT 模型的核心，由多个相同的层构成，每层都包括自注意力机制和 FFNN。自注意力机制允许模型在处理序列的每个元素时，考虑到序列中的所有其他元素，而 FFNN 进一步处理这些信息，使得模型能理解复杂的关系。

（4）分类器：在 Transformer 编码器的输出基础上，ViT 通常会使用一个全连接层或者特定的类别嵌入向量来进行最终的图像分类。这个分类器根据 Transformer 编码器学习到的特征，预测输入图像的类别。

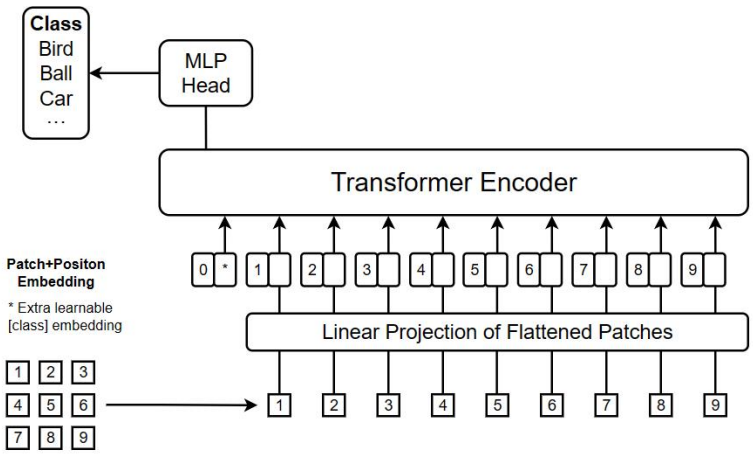


图 4.4 ViT 模型结构

4.3 ViT 舌苔分类实验结果与分析

支持本章实验的实验环境配置与上一章舌体分割实验的相同，这里不再赘述。在开始训练 ViT 模型之前，需要设置损失函数，优化器以及各个参数等。训练一个 ViT 模型通常要花很长时间，在本文中训练了 430 个 epochs。当正常输出每个 epoch 的 step 信息时，意味着训练正在进行，通过模型输出可以查看当前训练的 loss 值和时间等指标。如果未使用预训练模型参数，则需要更多的 epoch 来训练。

通过多次实验，找到较佳参数，如表 4.2 所示，其中，权重衰减（weight decay）防止模型过拟合；学习率（learning rate）决定模型的权重按照梯度的方向调整的量；动量（Momentum）帮助模型更快地收敛，并减少在训练过程中的震荡。

在本次舌苔分类实验中，本文采用了 Top_1_Accuracy 作为衡量模型表现的标准。Top_1_Accuracy 是指模型预测出的概率最高的类别恰好是正确类别的概率。实验结果显示，模型的 Top_1_Accuracy 达到了 72%，这是一个相对较高的水平，部分得益于我们使用了预训练模型参数。然而，需要注意的是，舌苔分类模型训练和验证所需的数据集属于医学领域，获取高质量的图像本身就具有一定的挑战性。此外，本次实验所用的数据集并未经过资深中医师的检验，因此，最终的实验结果可能存在一定的偏差。

表 4.2 实验参数设置

参数名称	参数
权重衰减	5×10^{-4}
最小学习率	0.0001
动量	0.9
学习率下降方式	Cosine Annealing
优化器	SGD
Batch Size	32
Epoch	430

本文将同属于 Transformer 架构的 ViT 与 Swin Transformer 进行舌苔分类训练和验证，分别得到如图 4.5 和图 4.6 的 ViT 损失曲线和 Swin Transformer 损失曲线。

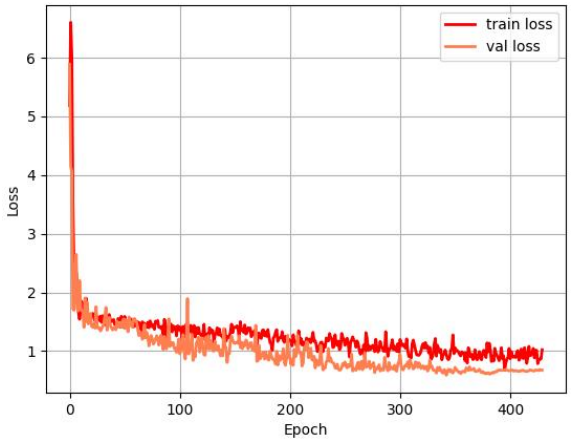


图 4.5 ViT 训练、验证损失

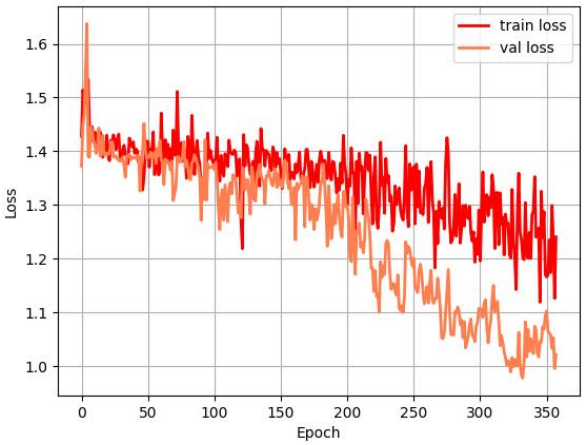


图 4.6 Swin Transformer 训练、验证损失

根据 Swin Transformer 损失曲线可以得知，Swin Transformer 的训练损失和验证损失出现了更多“锯齿”（即图中出现很多尖锐的峰和谷），并且损失曲线也没有迅速下降，甚至在中后期波动越来越大，说明该模型对于舌苔图像分类任务的拟合较差，并不是适合该任务的模型。而 ViT 损失曲线表明，随着训练的进行，损失曲线逐渐趋于稳定，下降速度变慢。这表明模型开始捕捉数据中的复杂关系，并且损失函数的最小值正在被逼近。在训练的后期，损失曲线趋于平缓，表明 ViT 模型已经接近其最优性能。此时，损失函数的值变化不大，模型参数的更新也变得较慢。

本文采用了一张 Thin-White 类别的舌头图片作为推理图片来测试模型表现，分别测试 ViT 和 Swin Transformer 是否可以给出正确的预测结果，如图 4.7 和图 4.8 所示。

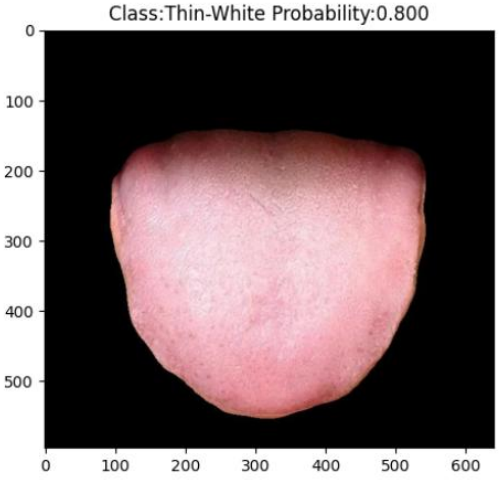


图 4.7 ViT 预测结果

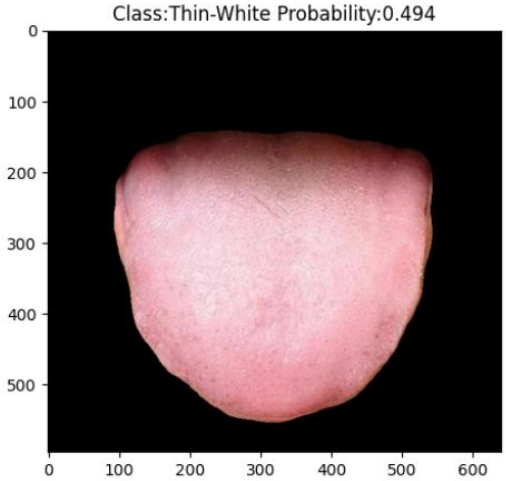


图 4.8 Swin Transformer 预测结果

显而易见，虽然两个模型都能成功预测该舌头照片的舌苔类别为 Thin-White，但是 ViT 的预测概率为 80%，而 Swin Transformer 的预测概率仅为 49.4%，与 ViT 预测效果相差甚远。因此，本文采取 ViT 作为舌苔图像分类模型是合理的。

4.4 本章小结

本章深入分析了 ViT 在舌苔分类实验中的全过程和结果，并与 Swin Transformer 进行了效果对比，从而明确指出 ViT 作为舌苔 AI 导诊系统分类模型的优越性能。首先，本章详细介绍了舌苔图像数据集的构建和数据预处理方法。其次，分析了图像输入到 ViT 的整个流程，以及 ViT 的网络结构，指出 ViT 即使不依赖卷积操作，也能在图像分类任务中取得优异的效果。最后，通过对比评估 ViT 和 Swin Transformer 两种模型在舌苔分类任务中的表现，进一步证明了 ViT 是本次实验中最适配的模型。

第五章 开发舌苔 AI 导诊系统

在第三章和第四章中，本文分别对图像分割模型 DeepLab V3+和图像分类模型 ViT 进行了详细的介绍和分析。基于前两章的理论和实验基础，本章设计并实现了一个舌苔 AI 导诊系统。针对该系统的实际需求和使用场景，本章对系统的前端和后端进行了详细的设计和实现。具体来说，系统前端采用了 PyQt5 框架进行开发，后端则使用了 Flask 框架。此外，为了满足移动设备用户的需求，我们还开发了一个手机端 App，采用 Android 技术实现。

5.1 应用场景

本文开发的舌苔 AI 导诊系统是一种融合了人工智能技术的诊断工具，通过分析舌苔图像来辅助中医诊断过程。在中医看病过程中，望诊是是一个非常重要的环节，特别是观察舌头的情况，对于判断一个人的健康状况尤为关键。舌苔 AI 导诊系统的开发目的是，通过舌部特征提取和分析分类，提升舌诊的准确率与效率以及提高医疗服务的便捷性。以下是舌苔 AI 导诊系统可能的应用场景：

（1）中医诊疗机构：在中医诊所或者大医院的中医科，该 AI 系统可以作为一个助手，帮助医生更快、更准确地进行诊断。

（2）远程医疗服务：通过远程医疗的平台，患者可以拍一张自己的舌头照片，然后上传到 AI 系统进行初步分析，之后由专业的中医师在网上进行复核。

（3）移动健康应用：在一些健康管理或者中医养生的手机应用里，用户可以上传自己的舌头照片，系统会立刻给出健康状态的评价和一些个人化的建议。

（4）公共卫生监测工具：AI 舌苔导诊系统可用于大规模人群健康状态的监测，尤其在疫情或流行病爆发期间，快速识别病症特征。

5.2 舌诊依据

在中医学中，舌诊通过观察舌头的舌质和苔色等特征来诊断疾病。中医认为舌为心之苗，脾之外候，能准确反映人体的健康状况。舌质和苔色的变化不仅是病理变化的外在表现，也是中医辨证施治的重要依据。

舌质，又称舌体，主要指舌头的肌肉组织和血液充盈情况。在中医理论中，舌质的

颜色和形态变化与人体的气血、阴阳状态密切相关。例如，舌色淡白多为提示气血两虚或阳虚；舌色红绛常与热病或阴虚火旺有关。

舌苔是指覆盖在舌体上的一层薄白膜，其由胃气所生，反映脾胃的生理和病理状态。舌苔的颜色、厚度、干湿等特征在中医诊断中具有重要意义，例如，白苔通常与表证、寒证相关，如外感风寒或脾胃虚寒。

在中医临床实践中，舌质与舌苔常结合分析，以获得更全面的病理信息。例如，舌质淡白且苔白滑，可能提示阳虚内寒；舌质红而苔黄燥，则可能指示热盛伤津。舌苔与舌质的观察是中医舌诊的核心，它们为中医提供了丰富的病理信息。表 5.1 表示舌苔与人体中医体质的对应关系，来自于中国中医药出版社出版的教材《中医诊断学》与多名中医医师的诊断经验。

表 5.1 舌苔与人体中医体质的对应关系

舌质与苔色	中医诊断体质
淡红舌、薄白苔	脾气虚
淡白舌、白厚苔	脾虚兼寒湿气
红舌、黄厚苔	脾虚湿热
青紫舌、黄厚苔/灰黑苔	血淤

5.3 舌苔 AI 导诊系统实现

5.3.1 电脑端 PyQt5 可视化程序

5.3.1.1 相关技术

（1）PyQt5 框架

PyQt5 是能让开发者创建可以在不同操作系统上运行的图形用户界面（GUI）应用程序的功能强大的工具集，它提供了一套丰富的 API，允许开发者使用 Qt 框架下的 C++ 核心功能。PyQt5 是 Qt for Python 的官方版本，它使用 Qt5 的 LGPL 授权。PyQt5 不仅支持事件驱动编程，还提供了信号和槽机制，这对于开发响应式用户界面很重要。

在舌苔 AI 导诊系统开发中，PyQt5 被选作开发舌苔 AI 导诊系统的电脑端界面。该系统可以提供一个直观、交互性强的平台，使用户能够方便地上传舌苔图像，并接收 AI 系统的诊断结果。本文首先利用 PyQt5 的 Qt Designer 和相关布局管理器，设计了一

个用户友好的界面，包括图像上传区域、诊断结果显示区以及必要的交互控件。通过PyQt5 的信号和槽，实现了用户操作（如上传图像）与后台处理（AI 诊断）之间的同步。系统还实现了包括错误处理、用户提示和反馈机制在内的用户交互功能。

通过 PyQt5 开发的舌苔 AI 导诊系统电脑端程序，不仅提高了中医诊断的可访问性和便捷性，而且通过集成先进的 AI 技术，增强了诊断的可交互性和效率。

（2）PostgreSQL 14

PostgreSQL 是一个高度可扩展的开源对象关系数据库系统，它十分稳定、拥有强大的功能，以及遵守 SQL 标准。PostgreSQL 14 作为该系列的最新版本，引入了多项改进和新特性，使其成为存储和管理舌苔 AI 导诊系统数据的理想选择。

在本文开发的舌苔 AI 导诊系统中，PostgreSQL 14 作为后端数据库用来存储、检索和管理与用户健康相关的数据。PostgreSQL 14 的应用为系统带来了以下三个好处：

1）数据存储与管理：系统使用 PostgreSQL 14 来保存用户的舌苔图像、个人信息、诊断记录和用户的反馈。

2）复杂查询支持：PostgreSQL 14 提供了强大的 SQL 查询功能，让系统能够执行复杂的数据搜索和分析。这意味着用户可以快速找到自己的病例数据。

3）数据完整性保障：PostgreSQL 14 通过外键、触发器和各种数据约束等特性，帮助确保存储在数据库中的信息是准确和一致的。

本文通过 PostgreSQL 14 的集成，舌苔 AI 导诊系统能够将使用者的登录帐号和密码以及就诊记录存入 PostgreSQL 数据库中，能够在相应的功能中查询数据库中的数据，从而显示在可视化程序中。在登录界面中，程序会根据使用者输入的帐号和密码，与数据库中存储的所有帐号和密码进行一一比对（即数据库的读取操作，使用 SELECT 语句），确保两者均输入正确，才允许使用者进入功能界面；在查询诊断记录界面中，程序同样使用数据库读取操作，查询该使用者的所有数据记录，并显示出来；在注册界面中，程序会将使用者输入的帐号和密码录入 PostgreSQL 数据库中（即添加新的数据记录，使用 INSERT 语句）；在修改密码界面中，程序要求使用者正确输入两次密码，才允许其修改密码，并将新密码更新到数据库中该用户的数据记录对应项（即数据库的更新操作，使用 UPDATE 语句）；在注销帐号界面中，程序会将该使用者在数据库中的所有数据记录删除（即数据库删除操作，使用 DELETE 语句）。

创建（Create）、读取（Read）、更新（Update）和删除（Delete），这些操作统称为数据库的增删查改（CRUD），是数据库管理的核心[37]。而在设计数据库应用程序时，PostgreSQL 能使 CRUD 操作实现安全性、完整性和高效管理，为舌苔 AI 诊断提供了一个坚实的数据基础。

5.3.1.2 系统设计

（1）系统功能设计

通过该系统的运行，用户可以轻松快捷地完成对舌苔识别和分类的诊断，提高中医舌诊效率，可以让博大精深的中医舌诊能够快速便捷地惠及每一位百姓。系统功能的主要有 AI 舌诊、历史记录查询、帐号的登录和注册、密码修改等功能。系统功能设计图如图 5.1 所示。

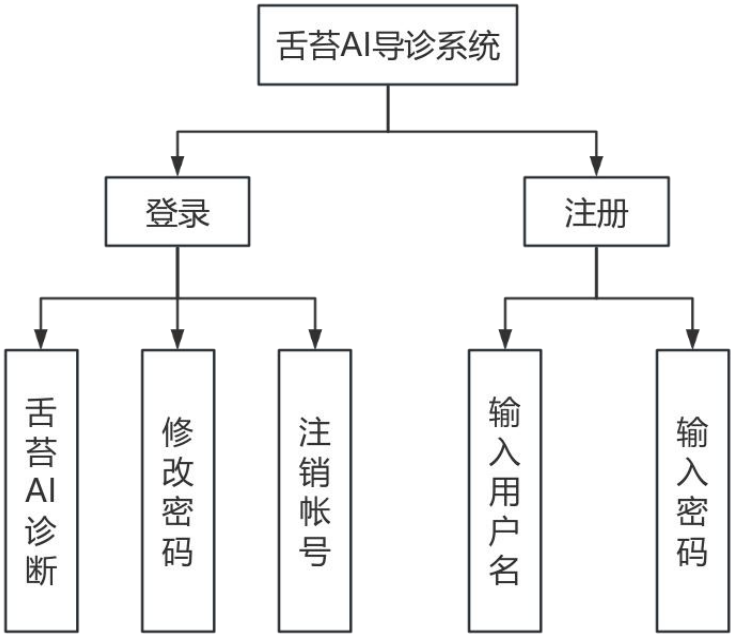


图 5.1 系统功能设计图

（2）系统流程设计

1）登录：输入用户名和密码，程序根据用户输入的用户名和密码，到数据库查询，若正确则登录成功，并保存用户名以备后用。若卡号存在但密码输入不正确，或无该卡信息，则系统提示密码不符，清空用户名输入栏和密码输入栏，等待用户的重新输入。登录流程图如图 5.2 所示。

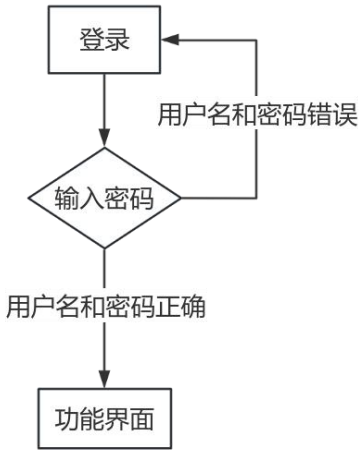


图 5.2 登录流程图

2）注册：要求用户输入用户名，以及两次相同的密码。如果第二次输入的密码与第一次输入的密码不相同，系统则清空用户名输入栏和密码输入栏并要求其重新输入。若两次密码输入相同，则系统会将该用户注册的用户名和密码存储到数据库。注册流程如图 5.3 所示。

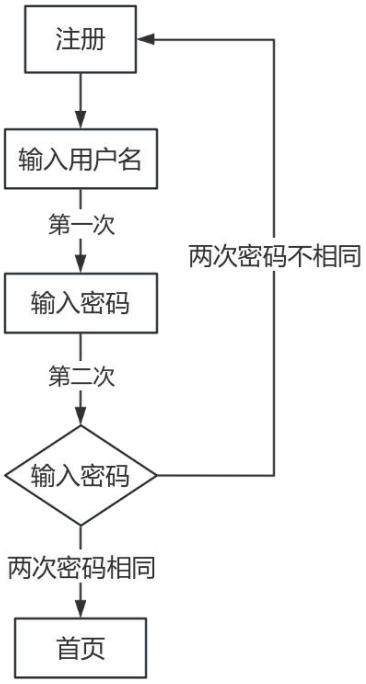


图 5.3 注册流程图

3）舌苔 AI 诊断：用户通过手机 App 扫描界面上的二维码，传输手机相册的舌头图像或者调用手机后置摄像头进行实时拍摄图片，并上传该图片。点击“诊断”按钮，系

统则会根据第三章的DeepLab V3+模型自动分割图像中的舌体及第四章的ViT模型将舌苔分类，并且向用户反馈中医体质特征、用药指导等。舌苔AI诊断流程如图5.4所示。

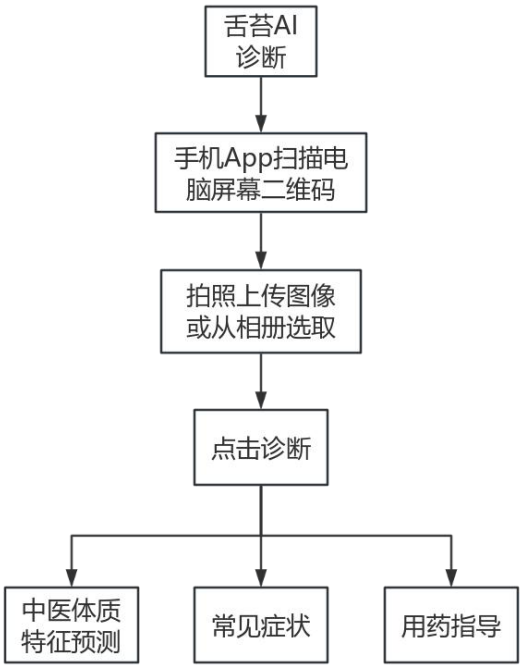


图5.4 舌苔AI诊断流程图

4）就诊记录查询：用户可通过该功能查看过往的就诊记录，包括诊断时间、上传的舌头图像、中医体质特征、用药指导等。就诊记录查询流程如图5.5所示。

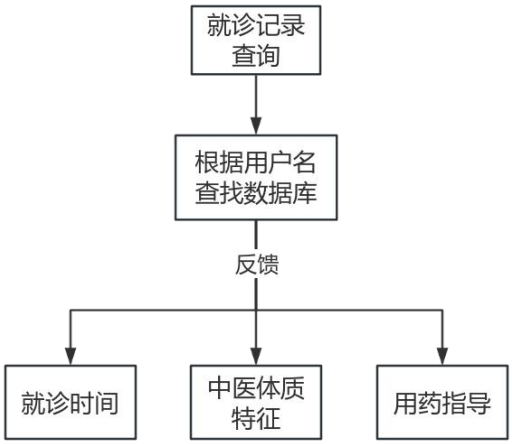


图5.5 就诊记录查询流程图

5）密码修改：用户需要输入两次相同的新密码，才能成功修改密码。此时，系统会将新密码安全地存储到数据库中，新密码会覆盖旧密码，旧密码失效。若两次输入的新密码不相同，系统则清空用户名输入栏和密码输入栏并要求其重新输入。密码修改流

程如图 5.6 所示。

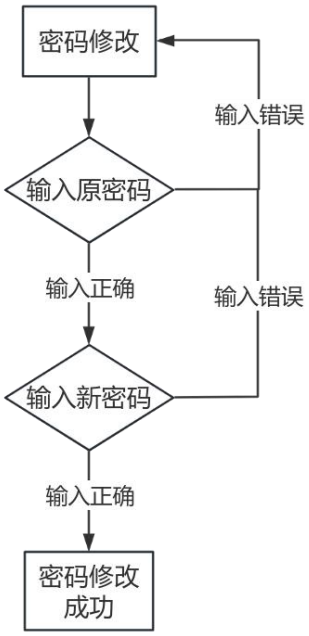


图 5.6 密码修改流程图

6）帐号注销：用户需要输入两次相同的密码，才成功注销帐号。此时，系统会安全地删除数据库中该用户的所有信息，包括用户名、密码和就诊记录，该用户的用户名和密码失效。若两次输入到密码不相同，系统则清空用户名输入栏和密码栏。帐号注销流程如图 5.7 所示。

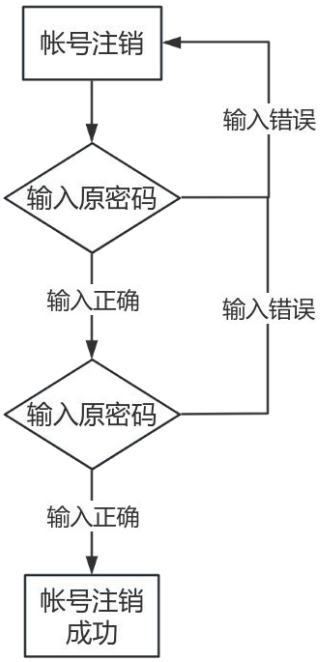


图 5.7 帐号注销流程图

5.3.1.3 PyQt5 可视化程序开发

本文开发了一个基于 PyQt5 的舌苔图像分类与分析的可视化程序。该程序旨在提供一个用户友好的界面，使非专业用户也能轻松地上传舌苔图像，并接收 AI 系统的诊断反馈。程序的设计遵循了直观性、交互性和可扩展性的原则。

用户界面(User Interface, UI)是程序与用户交互的窗口。本文利用 PyQt5 的 QWidget、QFrame、QPushButton、QLabel 和相关布局管理器等，设计了一个直观的 UI，包括图像上传区域、诊断结果显示区以及必要的交互控件。程序的核心是集成的舌苔图像分类 AI 模型。本文采用了第三、四章经过相应数据集训练的深度学习模型 DeepLab V3+和 ViT，并利用相应函数接口进行模型拼接，以适应输入的舌苔图像数据。模型的集成允许程序自动从上传的图像中提取特征，并进行分类。诊断结果通过可视化模块以图形化的方式展示给用户，包括舌苔的分类标签和置信度。此外，程序还提供了就诊记录查询功能，方便用户查看过往就诊记录。

PyQt5 可视化程序为舌苔图像的分类与分析提供了一个高效、准确且用户友好的工具。系统的各个界面如图 5.8、5.9、5.10、5.11、5.12、5.13 所示。

图 5.8 登录界面

图 5.9 注册界面

图 5.10 上传舌头图像界面

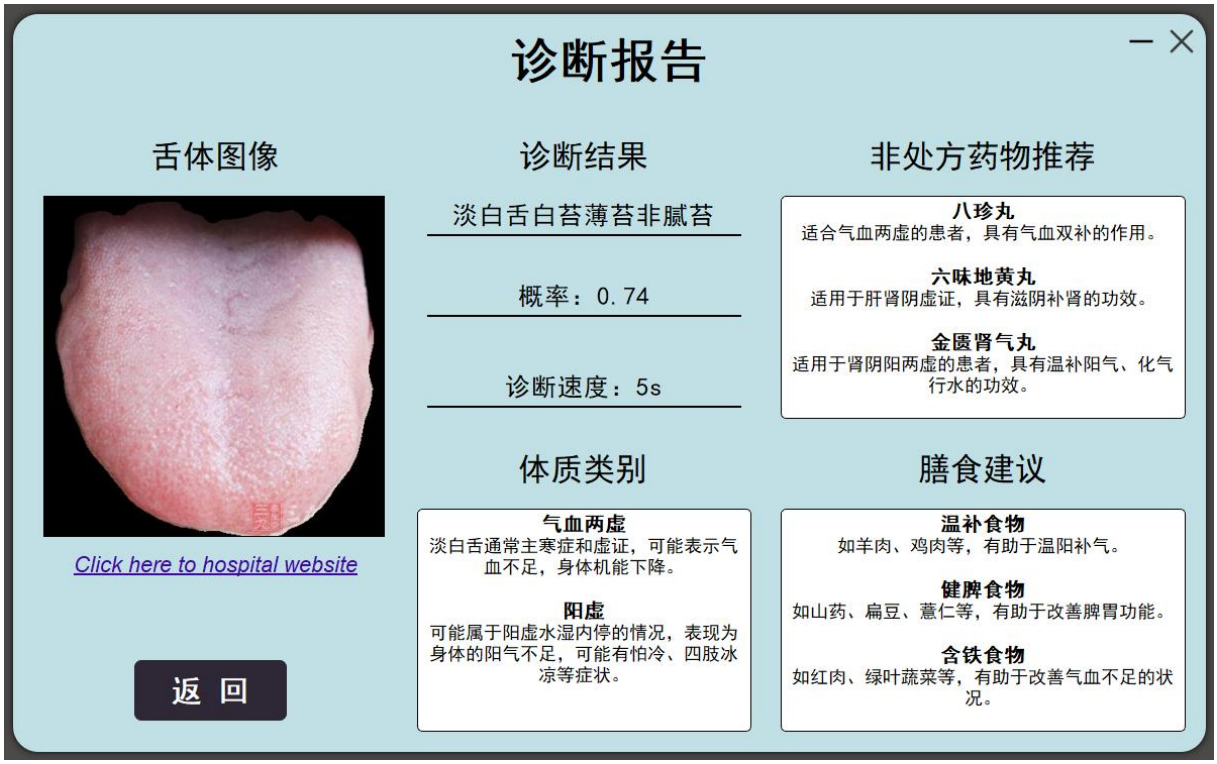


图 5.11 诊断报告界面

图 5.12 密码修改界面

图 5.13 帐号注销界面

5.3.2 手机端 Android App

5.3.2.1 相关技术

本文开发了一款 Android 手机端 App，该 App 能够扫描由电脑端舌苔 AI 导诊系统生成的二维码，并支持图像的上传功能，能让用户便捷地传输舌舌部图像。

应用程序采用 Android Studio 作为开发环境，利用 Java 语言进行编程。Android App 使用了华为移动服务（Huawei Mobile Services, HMS）中的 Scan Kit 开发工具包来实现 Android 应用中的二维码扫描功能。相比于以往常用的 ZXing 开源库，Scan Kit 有如下四点优势：

1）Scan Kit 为了应对各种复杂的扫码环境，比如会遇到反光、光线不足、条码脏污、图像模糊或者在柱面物体上的情况，都做了特别的优化。这让它的扫码成功率更高，用户用起来也更舒服。而相比之下，ZXing 在某些特定的设备或者环境下，可能就需要一些额外的调整，才能发挥出最好的效果。

2）Scan Kit 提供多种调用模式，包括 Default View Mode、Customized View Mode、Bitmap Mode 和 MultiProcessor Mode，允许开发者根据自己的需求快速集成扫码功能。

ZXing 则需要开发者进行更多的集成和自定义工作。

3）Scan Kit 在华为手机上使用增强识别模型，提供优秀的识别能力。ZXing 作为一个广泛使用的库，其性能在不同设备和场景下可能会有所差异。

4）Scan Kit 允许开发者通过较少的代码量实现扫码功能。ZXing 则可能需要更多的代码和配置来实现相同的功能。

本文分别用 Scan Kit 和 ZXing 实现的扫码功能，如图 5.14 所示。肉眼可以明显看到使用 Scan Kit 扫码时二维码并没有失真，扫码体验较好，而使用 ZXing 时二维码明显失真，效果较差，因此本文选择 Scan Kit 来实现 Android App 的扫码功能。

(a)Scan Kit 扫码　　(b)ZXing 扫码

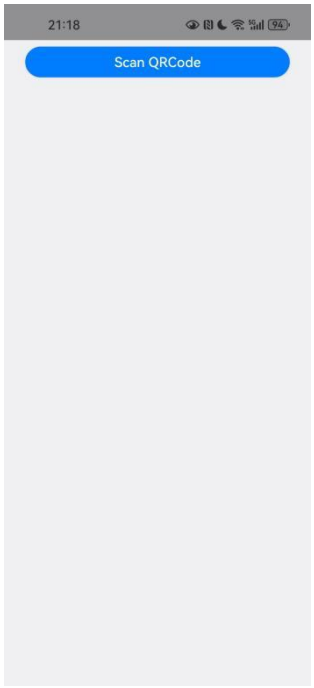
图 5.14 Scan Kit 与 ZXing 扫码对比图

除了扫描二维码以外，应用程序允许用户从手机相册选择图片。选择的图像会经过预处理，如调整大小、压缩和格式转换，以适应网络传输的要求。应用实现了 HTTP 或 HTTPS 协议的网络请求，以将图像数据安全地传输至 PyQt5 舌苔 AI 导诊系统的服务器。

开发的 Android 应用程序为 PyQt5 舌苔 AI 导诊系统提供了一个移动访问端，使用户能够更加便捷地使用舌苔图像诊断服务。通过集成先进的图像处理和网络通信技术，该应用程序不仅提高了诊断的可访问性，而且增强了用户交互的友好性。

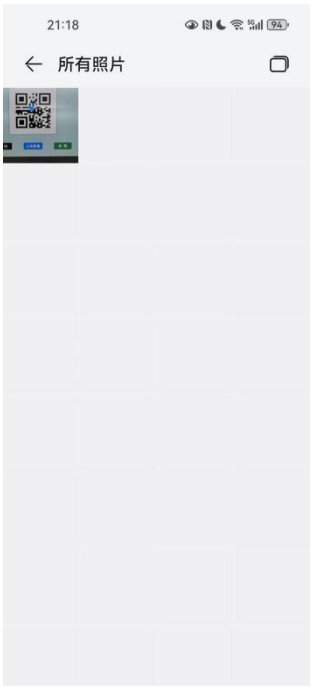
5.3.2.2 Android App 界面设计

本文设计的 Android App 界面如图 5.15 所示，从左到右依次为主页面，扫码页面，选择相册照片页面。



(a)主页面

(b)扫码页面



(c)选择文件页面

图 5.15 Android App 界面

使用者打开 Android 应用程序后，可以看到主页面，点击“Scan QRCode”，程序将使用华为的 Scan Kit 打开扫码界面，并且会调用手机后置摄像头对可见范围内的二维码进行扫描。值得一提的是，Scan Kit 能够实现高效的扫码服务，在两米范围内，都能让使用者的手机轻松识别二维码，并且识别速度非常快，基本 1 秒内即可识别成功。扫码成功后，程序会访问手机图片文件，选中一张舌头图片，即可上传至 Flask 服务器端，再由 PyQt5 客户端通过网络请求从 Flask 服务器请求图片数据，由此即可将获取的舌头图片数据加载到 PyQt5 程序中，并在舌诊界面的 QLabel 控件上显示。

5. 3. 3 Flask 服务器端

5.3.3.1 相关技术

本文采用 Flask 作为 Android App 和 PyQt5 客户端的图片传输媒介。Flask 是一个用 Python 编写的轻量级 Web 应用框架。它被设计为易于使用和扩展，是构建简单网站到复杂的、动态的 web 应用的理想选择。以下是 Flask 的一些关键特点：

（1）核心简单：Flask 的核心非常简单，便于初学者使用。

（2）扩展性：Flask 提供了大量的扩展库，可以方便地添加各种功能，如用户身份验证、数据库集成、表单处理等。

（3）灵活性：Flask 不强迫开发者遵循特定的项目结构，而是可以根据开发者的喜好来组织代码。

由于本文构建的舌苔 AI 导诊系统是一个面向简单需求、项目周期短的小应用，故本文选择了 Flask 作为服务器。

5.3.3.2 舌头图像传输逻辑

Android App 上传图片到 Flask 服务器，再由 Flask 传输到 PyQt5 程序的整个过程可以分为以下几个步骤。

（1）Android App 端

1）使用 Scan Kit 扫描 PyQt5 客户端上的二维码，选择要发送的舌头图片。

2）利用 OkHttp 网络请求库，将图片转换成二进制数据，然后发送到 Flask 服务器。

（2）Flask 服务器端

1）接收到 Android App 传来的图片数据。

2）把图片数据存储在服务器的文件系统里。

（3）PyQt5 客户端

1）通过网络请求向 Flask 服务器索取图片数据。

2）接收到图片数据后，将其加载进 PyQt5 程序，并在用户界面上展示出来。

5.4 本章小结

首先，本章阐述舌苔 AI 导诊系统的应用场景及中医舌诊依据。其次，本章分别对 PyQt5 客户端、Android 手机端和 Flask 服务器端的相关技术与相应的系统设计进行详细说明，包括 PyQt5 的可视化程序设计、Android App 的扫码与传输图像文件界面设计和 Flask 处理舌头图像传输的逻辑。

总 结

本文旨在实现一个舌苔 AI 导诊系统，专注于中医舌诊的现代化和智能化。为了提升舌诊的准确性和客观性，本文引入了深度学习技术。针对智能化中医舌诊的舌体分割和舌苔分类，采用了两个先进的深度学习模型：一个是用于舌体分割的基于 CNN 的 DeepLab V3+模型，另一个是用于舌苔分类的基于 Transformer 的 ViT 模型。为了便于用户操作和结果的可视化展示，本文使用 PyQt5 开发了电脑端的可视化应用程序，并利用 Android Studio 设计了移动端的 App，并将 DeepLab V3+和 ViT 模型集成其中，使得用户可以在界面上直观地进行舌诊图像的上传和诊断结果的查看。此外，本文通过 Flask 服务器端技术，实现手机端到电脑端的图像传输。这样，用户在手机端拍摄的舌头图像可以快速传输到电脑端，进行深度学习模型的分析和处理，最终实现一个完整的端到端的 AI 诊断流程，为中医舌诊的数字化和智能化提供了一个有效的解决方案。本文主要内容如下：

（1）系统设计包括两个核心组件：基于 CNN 的 DeepLab V3+舌体分割模型和基于 Transformer 的 ViT 舌苔分类模型。通过这两个模型的结合，系统能够有效地实现对舌体的精确分割和舌苔的准确分类。训练 DeepLab V3+模型时采用 MobileNet V2 作为主干网络以及迁移学习的策略，加上其编解码结构和 ASPP 模块，使其拥有强大的语义分割能力，实现了对舌头图像中舌体和背景的精确区分。ViT 模型利用自注意力机制，对舌苔图像进行分类，有效地捕捉了图像的全局特征，达到了良好的舌苔分类效果。通过 DeepLab V3+与 U-Net 对比实验以及 ViT 与 Swin Transformer 对比试验，验证了模型的有效性和实用性。实验结果表明，本文在舌体分割和舌苔分类任务上均取得了较为满意的性能，DeepLab V3+对舌体图像分割任务的 MIoU 达到 91.6%，ViT 对舌苔图像分类任务的准确率达到 72%。

（2）系统还包括了基于 PyQt5 的电脑端可视化程序和 Android Studio 开发的移动端 App，以及通过 Flask 服务器端实现的图像传输功能，最终开发了跨平台的可视化应用程序，实现了用户友好的交互界面。

（3）存在一些局限性，本文舌头图像数据集质量参差不齐、规模有限，并且没有经过经验丰富的中医师进行检验，大大影响了模型的泛化能力。

参考文献

[1]陈涌铨.基于人工智能的舌象辅助诊断系统[J].移动信息,2023,45(8): 210-212.

[2]梁嵘.《敖氏伤寒金镜录》的作者及版本流传[J].中华医史杂志,2017,47(2): 115-120.

[3]朱化珍.对中医舌诊现代研究的再认识[J].吉林中药,2008,28(5): 328-329.

[4]曾凤,赵莺.中医舌诊与糖尿病的相关性研究概述[J].甘肃中医,2008,21(3): 52-54.

[5]Jiang Zhihao,Zhu Kai,Lu Xiaozuo,et al. Analysis of tongue information in coronary artery disease[C/OL]//2008 IEEE International Symposium on IT in Medicine and Education, Xiamen, China. 2008: 287-290.

[6]姚嫱.乳腺癌和肺癌患者舌象特点及客观量化的研究[J].辽宁中医杂志,2008,35(2): 163-164.

[7]Qu P,Zhang H,Zhuo L,et al. Automatic Tongue Image Segmentation for Traditional Chinese Medicine Using Deep Neural Network[M/OL]//Intelligent Computing Theories and Application,Lecture Notes in Computer Science. 2017: 247-259.

[8]马龙祥.基于高分辨率特征的舌象分割算法研究[J].计算机工程,2020,46 (10): 248-252.

[9]谭建聪,肖晓霞,邹北骥.一种基于实例分割的舌体分割方法[J].中国卫生信息管理杂志,2023,20(03): 459-464.

[10]梁金鹏,杨浩,张海英.基于颜色特征的常见舌质舌苔分类识别[J].微型机与应用,2017,36(17): 102-105.

[11]瞿婷婷,夏春明,王忆勤.基于 Gabor 小波变换的舌苔腐腻识别[J].计算机应用与软件,2016,33(10): 162-166.

[12]钟振.基于深度学习的中医舌像分割与识别实验研究[D].福州:福建中医药大学,2021.

[13]刘伟,陈锦明,刘波.基于深度学习的舌体图像分割和舌色分类研究[J].Digital Chinese Medicine,2022,5(03): 253-263.

[14]董易杭,王建勋,王晶,等.基于深度学习的阳虚质与阴虚质舌象分类研究[J/OL].中华中医药学刊: 1-12.

[15]张居正.基于深度学习的端到端车道线检测方法[D].长沙: 湖南大学,2019.

[16]黄潇峰.基于深度学习的移动机器人实时目标检测算法研究与应用[D].成都: 成都理工大学,2020.

[17]申润业.SDN 中时间序列智能异常检测[D].北京: 北京邮电大学,2021.
[18]朱海龙.时间序列健康数据的分析与预测[D].杭州: 杭州师范大学,2018.
[19]贾佳,公茂盛,赵一男.基于深度学习算法的地震动重要持时预测模型[J]: 振动与冲击,2023,42(19): 249-259.
[20]姜光杰.基于 FastReID 哈希编码的图像检索研究[D].南昌: 南昌大学,2022.
[21]张亚琼.基于深度迁移学习的书法字体与文字内容同步识别方法研究[D].西安: 西安电子科技大学,2020.
[22]石钏.基于深度学习的太赫兹人体安检图像处理算法研究[D].西安: 西京学院,2022.
[23]兰朝凤,王顺博,郭小霞,等.基于 DCNN 和 BiLSTM 的单通道视听融合语音分离方法研究[J].电子学报,2023,51(04): 914-921.
[24]王明宇.基于统计特征和单样本分类的中文新词发现[D].厦门: 厦门大学,2020.
[25]Vaswani A,Shazeer N,Parmar N,et al. Attention is All you Need[J]. Neural Information Processing Systems,Neural Information Processing Systems, 2017.
[26]周家鑫.基于机器学习的态势感知系统的设计和实现[D].北京: 北京邮电大学,2021.
[27]赵继樑.基于双支路神经网络的多分辨率遥感图像融合分类[D].西安: 西安电子科技大学,2021.
[28]唐智贤,李萍,曾雅楠,等.计算机辅助舌象分析诊断研究进展[J].医学信息学杂志,2022,43(06): 36-39+71.
[29]Li Y,Zhang K,Cao J,et al. LocalViT: Bringing Locality to Vision Transformers[J]. arXiv: Computer Vision and Pattern Recognition,arXiv: Computer Vision and Pattern Recognition, 2021.
[30]李旻谕.基于深度学习的遥感图像目标检测算法研究[D].成都: 电子科技大学,2022.
[31]于嘉山.基于深度学习的超分辨率重建技术[D].成都: 电子科技大学,2022.
[32]李茂莹,杨柳,胡清华.同构迁移学习理论和算法研究进展[J].南京信息工程大学学报(自然科学版), 2019, 11(3): 269-277.
[33]张雪松,庄严,闫飞,等.基于迁移学习的类别级物体识别与检测研究与进展[J].自动化学报,2019,45(7): 1224-1243.
[34]黄丽华.水下图像增强与水下生物目标检测研究[D].上海: 上海海洋大学,2022.
[35]白静亚.基于机器视觉的棉田害虫图像采集与识别系统研究与改进[D].石河子: 石河

子大学,2022.
[36]罗凌云.基于嵌入式平台的图像语义分割研究与实现[D].成都: 电子科技大学,2021.
[37]刘超,赵建平,穆星,等.RBAC 权限管理在实验室管理系统中的应用[J].电子技术,2016,45(01): 40-43+37.

致 谢

随着这篇毕业论文的最后一个句号的落定，我四年的本科旅程也即将画上圆满的句号。在此，我想对那些在这段旅程中给予我支持与帮助的每一个人表达我的深深谢意。

首先，我要特别感谢我的指导老师何昭水教授。在整个毕业设计和论文撰写期间，何教授不仅以其深厚的专业知识和严谨的学术态度给予了我悉心的指导，而且以其丰富的人生经验和智慧给予了我许多宝贵的建议和鼓励。我还要感谢自动化学院的全体老师和同学们。在学习和生活中，他们提供了一个充满活力和互助的学习环境，使我能够不断进步和成长。特别感谢我的舍友们，他们在我需要帮助时总是无私地伸出援手，一起讨论问题、分享知识，让我在学术探索的道路上从未感到孤单。母校广东工业大学提供了一个充满活力、鼓励创新和自由探索的学术环境。在母校学习的这四年里，母校不仅为我提供了丰富的学习资源，还为我的成长和发展提供了宝贵的机会，我表示衷心的感谢，这些支持极大地促进了我的学术研究和个人发展。最后，我要感谢我的家人，他们一直是我学习和生活中的坚强后盾。感谢他们对我的无条件支持和鼓励，无论我遇到多大的困难和挑战，总是给予我力量和勇气。没有家人的爱和奉献，我不可能取得今天的成就。

随着毕业的临近，我满怀感激地回望过去，同时充满期待地展望未来。我相信，这段经历将成为我人生中宝贵的财富，伴随我走向更加广阔的天地。再次感谢所有给予我帮助和启发的人，是你们让这段旅程变得如此丰富和有意义。