

分类号\_\_\_\_\_ 密级\_\_\_\_\_

UDC \_\_\_\_\_

西華大學

# 硕士学位论文

(专业学位)

基于深度学习的战国竹简文字识别

作者姓名: 谢文达

学位类别: 电子信息硕士专业学位

学号: 212021085400033

校内导师: 蒋文波

校外导师: 何渝

完成日期: 2024年6月

Classified Index: \_\_\_\_\_

UDC: \_\_\_\_\_

**Xihua University**  
**Master Degree Thesis**

**Character recognition of Bamboo Slips in  
the Warring States Period based on deep  
learning**

**Candidate : Xie Wenda**

**Major : Electronic Information**

**Student ID: 212021085400033**

**Supervisor: Prof.Jiang Wenbo,  
Associate Prof. He Yu**

**June, 2024**

## 摘要

古文字识别工作对于文化遗产、历史研究、学术研究等方面都具有重要意义，有助于推动人类文明的传承和发展。但由于缺少公开的战国竹简文字识别数据集以及战国竹简文字本身存在的竹木纹路干扰、墨迹干扰、部分字体断裂和严重的样本数量不均衡等问题，现有的古文字识别方法对战国竹简文字的识别准确率不高。因此，针对目前存在的问题，有必要对战国竹简文字识别进行更加深入、更加具有针对性的研究。

在此背景下，论文围绕战国竹简文字识别上存在的问题开展研究，工作总结如下：

(1) 针对缺少战国竹简文字识别数据集的问题。本文以“清华简”为研究原型，整理搜集了 1001 种战国竹简文字，共 19042 张图像数据。首先，对图像数据进行标注；其次，根据图像样本数量将数据划分成 Many-shot 类、Medium-shot 类和 Few-shot 类数据子集；再次，对数据进行图片尺寸规范和标准化处理两种预处理操作；最后，对数据进行亮度增强、腐蚀增强和膨胀增强等增强操作，增强后共 26564 张图像数据，即完成战国竹简文字识别数据集 zgzt1001 的制作。

(2) 针对竹简上存在的竹木纹路、墨迹、部分字体断裂等干扰信息会对文字识别产生影响的问题，本文对 ResNet50 算法进行改进。首先，将空间注意力机制和自注意力机制进行特征融合，改进为空间自注意力机制 (SSA)，使模型更加关注竹简文字的主体信息，从而削弱了竹简上存在的干扰信息对文字识别带来的影响；其次，将其添加到 ResNet50 网络中的最后一个卷积层，完成了基于空间自注意力机制的战国竹简文字识别模型 (SSA-ResNet50) 的搭建；最后，进行模型训练和实验结果对比分析。实验结果表明，与改进前 ResNet50 模型相比，SSA-ResNet50 模型在数据集 zgzt1001 上平衡准确率提高了 1.67%，F1 分数提高了 1.34%。

(3) 针对战国竹简文字样本数量严重不均衡，会造成模型训练困难和识别性能差的问题，本文在 SSA-ResNet50 算法的基础上进行改进。首先，将 SSA-ResNet50 算法与自监督对比学习中的 SimSiam 算法结合设计了一个双分支网络；其次，完成了基于自监督双分支网络的战国竹简文字识别模型 (SBBN) 的搭建，提高了尾部类样本的识别准确率，从而削弱了数据集长尾分布造成的影响；最后，进行模型训练和实验结果对比分析。实验结果表明，SBBN 模型在 zgzt1001 上平衡准确率为 97.56%、F1 分数为 97.38%；在 Many-shot 类、Medium-shot 类和 Few-shot 类分别达到了 98.42%、97.45% 和 96.90% 的平衡准确率。与 SSA-ResNet50 模型相比，SBBN 模型在数据集 zgzt1001 上平衡准确率提高了 2.72%，F1 分数提高了 2.97%。在 Many-shot 类和 Medium-shot 类的平衡准确率分别提高了 0.82%、1.07%，在 Few-shot 类上平衡准确率提高了 5.58%。

(4) 设计了战国竹简文字识别系统。首先，利用 HTML、CSS 和 JavaScript 设计了前端界面；其次，使用 Flask 框架来实现后端服务；最后，对系统进行了用户注册登录功能、图像预处理功能和战国竹简文字识别功能展示。结果表明该系统实现了对战国竹简文字的准确识别，能够满足用户对战国竹简文字识别任务的需求，具有一定的实用价值。

本文关于战国竹简文字识别的关键技术研究，提高了战国竹简文字的识别准确率，为战国竹简文字的整理破译工作提供了参考。

**关键词：**战国竹简文字识别；注意力机制；双分支网络；自监督学习；Flask 框架

## Abstract

The recognition of ancient characters is of great significance for cultural heritage, historical research, and academic research, and it helps to promote the inheritance and development of human civilization. However, due to the lack of publicly available datasets for the recognition of Bamboo Slips from the Warring States Period, as well as the interference of bamboo and wood patterns, ink interference, partial font breaks, and serious imbalance in sample quantity that exist in the Bamboo Slips themselves, the existing methods for the recognition of ancient characters have a low accuracy rate for the recognition of Bamboo Slips from the Warring States Period. Therefore, it is necessary to conduct more in-depth and targeted research on the recognition of Bamboo Slips in the Warring States Period in response to the current problems.

In this context, this thesis focuses on the problems in the character recognition of Bamboo Slips in the Warring States Period. The work summary is as follows:

(1) Aiming at the problem of lack of Warring States Period Bamboo Slips character recognition dataset. In the thesis, 1001 kinds of Bamboo Slips of the Warring States Period were collected, with a total of 19042 image data. Firstly, the image data was labeled, secondly, the data was divided into Many-shot class, Medium-shot class and Few-shot data subset according to the number of image samples, and thirdly, the data were preprocessed by image size specification and standardization processing, and finally, the data were enhanced by brightness enhancement technology, corrosion enhancement and expansion enhancement, and a total of 26564 image data were added after enhancement, that is, the production of the Warring States Bamboo Slips character recognition dataset zgzt1001 was completed.

(2) In order to solve the problem that the interference information such as bamboo and wood grain, ink, and some font fractures on Bamboo Slips will affect character recognition, this thesis improves the ResNet50 algorithm. Firstly, the spatial attention mechanism and the self-attention mechanism were fused to improve the spatial self-attention mechanism (SSA), so that the model paid more attention to the main information of the Bamboo Slips, so as to weaken the influence of the interference information on the Bamboo Slips on the character recognition, secondly, it was added to the last convolutional layer in the ResNet50 network to complete the Warring States Period Bamboo Slips character recognition model based on the spatial self-attention mechanism (SSA-ResNet50) Finally, the model training and experimental results were compared and analyzed. Experimental results show that compared with the improved ResNet50 model, the SSA-ResNet50 model improves the equilibrium accuracy and F1 score on zgzt1001 by 1.67% and 1.34%.

(3) In view of the serious imbalance in the number of Bamboo Slip characters in the Warring States period, which will cause difficult model training and poor recognition performance, this thesis improves the SSA-ResNet50 algorithm on the basis of the problem. Firstly, the SSA-ResNet50 algorithm was combined with the SimSiam algorithm in self-supervised contrastive learning to design a dual-branch network, secondly, the construction of the Warring States Bamboo Script Recognition Model (SBBN) based on the self-supervised double-branch network was completed, which improved the recognition accuracy of tail samples and weakened the influence of the long-tail distribution of the dataset. The experimental results show that the SBBN model has a balance accuracy of 97.56% and an F1 score of 97.38% on zgzt1001, and a balance accuracy of 98.42%, 97.45% and 96.90% in Many-shot, Medium-shot and Few-shot, respectively. Compared with the SSA-ResNet50 model, the SBBN model improved the balance accuracy by 2.72% and the F1 score by 2.97% on zgzt1001. The balancing accuracy of Many-shot and Medium-shot was increased by 0.82% and 1.07%, respectively, and the balancing accuracy was increased by 5.58% on Few-shot.

(4) The character recognition system of Bamboo Slips in the Warring States Period was designed. Firstly, the front-end interface is designed by using HTML, CSS and JavaScript, secondly, the back-end service is realized by using the Flask framework, and finally, the user registration and login function, image preprocessing function and Warring States Bamboo Slips character recognition function are demonstrated. The demonstration results show that the system can accurately recognize the characters of the Warring States Period Bamboo Slips, which can meet the needs of users for the task of recognizing the characters of the Warring States Period and has certain practical value. The research on the recognition algorithm of the Warring States Bamboo Slips in this paper improves the recognition accuracy of the Warring States Bamboo Slips and provides a reference for the collation and decipherment of the Warring States Bamboo Slips.

In the thesis, the key technologies of the Warring States Bamboo Slips are researched, which improves the recognition accuracy of the Warring States Bamboo Slips and provides a reference for the collation and decipherment of the Warring States Bamboo Slips.

**Key Words:** Character recognition of Bamboo Slips in the Warring States Period; Attention mechanism; Dual-branch network; Self-supervised learning; Flask framework

## 目 录

|                       |     |
|-----------------------|-----|
| 摘 要.....              | I   |
| Abstract .....        | III |
| 1 绪论.....             | 1   |
| 1.1 研究背景与意义.....      | 1   |
| 1.2 国内外研究现状.....      | 2   |
| 1.2.1 古文字识别研究现状.....  | 2   |
| 1.2.2 战国竹简文字研究现状..... | 4   |
| 1.3 论文的主要工作与章节安排..... | 8   |
| 1.3.1 主要工作.....       | 8   |
| 1.3.2 章节安排.....       | 9   |
| 2 深度学习相关技术.....       | 11  |
| 2.1 深度残差网络.....       | 11  |
| 2.2 注意力机制.....        | 13  |
| 2.2.1 空间注意力机制.....    | 13  |
| 2.2.2 自注意力机制.....     | 14  |
| 2.3 长尾图像识别方法.....     | 14  |
| 2.3.1 重采样.....        | 15  |
| 2.3.2 重加权.....        | 16  |
| 2.3.3 迁移学习.....       | 17  |
| 2.3.4 解耦训练.....       | 17  |
| 2.4 实验环境.....         | 20  |
| 2.5 本章小结.....         | 20  |
| 3 构建战国竹简文字识别数据集.....  | 21  |
| 3.1 数据集制作.....        | 21  |
| 3.1.1 数据集标注.....      | 21  |
| 3.1.2 数据集划分.....      | 22  |
| 3.1.3 数据预处理.....      | 22  |
| 3.1.4 数据增强.....       | 23  |
| 3.2 数据集属性统计.....      | 24  |
| 3.3 识别难点.....         | 25  |
| 3.4 本章小结.....         | 26  |

|                                   |    |
|-----------------------------------|----|
| 4 基于空间自注意力机制的战国竹简文字识别算法研究 .....   | 28 |
| 4.1 改进空间自注意力模块 .....              | 28 |
| 4.1.1 空间注意力部分 .....               | 29 |
| 4.1.2 自注意力机制部分 .....              | 29 |
| 4.1.3 特征融合 .....                  | 30 |
| 4.2 改进网络模型主体框架 .....              | 30 |
| 4.3 模型训练及参数设置 .....               | 31 |
| 4.3.1 数据划分 .....                  | 31 |
| 4.3.2 评价指标 .....                  | 31 |
| 4.3.3 训练参数设置 .....                | 33 |
| 4.3.4 模型训练 .....                  | 33 |
| 4.4 实验结果及对比分析 .....               | 35 |
| 4.4.1 ResNet 系列模型的对比分析与预测结果 ..... | 35 |
| 4.4.2 改进模型的客观评价指标对比分析 .....       | 39 |
| 4.4.3 不同模型的客观评价指标对比分析 .....       | 40 |
| 4.5 本章小结 .....                    | 41 |
| 5 基于自监督双分支网络的战国竹简文字识别算法研究 .....   | 42 |
| 5.1 自监督对比学习 .....                 | 42 |
| 5.2 改进网络模型主体框架 .....              | 43 |
| 5.2.1 卷积网络学习分支 .....              | 43 |
| 5.2.2 自监督对比学习分支 .....             | 44 |
| 5.2.3 损失函数计算 .....                | 47 |
| 5.3 模型训练及参数设置 .....               | 48 |
| 5.3.1 评价指标 .....                  | 48 |
| 5.3.2 训练参数设置 .....                | 48 |
| 5.4 实验结果及对比分析 .....               | 49 |
| 5.4.1 客观评价指标对比分析 .....            | 49 |
| 5.4.2 模型预测结果对比分析 .....            | 52 |
| 5.5 本章小结 .....                    | 55 |
| 6 战国竹简文字识别系统的设计与应用 .....          | 56 |
| 6.1 系统架构设置 .....                  | 56 |

|                         |    |
|-------------------------|----|
| 6.1.1 Flask 框架 .....    | 56 |
| 6.1.2 MySQL 数据库 .....   | 58 |
| 6.2 系统设计流程 .....        | 58 |
| 6.2.1 系统结构设计 .....      | 58 |
| 6.2.2 系统功能设计 .....      | 59 |
| 6.2.3 数据库设计 .....       | 60 |
| 6.3 系统应用与功能展示 .....     | 61 |
| 6.3.1 用户注册功能 .....      | 61 |
| 6.3.2 用户登录功能 .....      | 62 |
| 6.3.3 图像预处理功能 .....     | 64 |
| 6.3.4 战国竹简文字识别功能 .....  | 65 |
| 6.4 本章小结 .....          | 68 |
| 7 总结与展望 .....           | 69 |
| 7.1 总结 .....            | 69 |
| 7.2 展望 .....            | 70 |
| 参考文献 .....              | 71 |
| 攻读硕士学位期间发表论文及科研成果 ..... | 75 |
| 致 谢 .....               | 76 |

# 1 绪论

## 1.1 研究背景与意义

纵观中国上下五千年历史，文字是中华文明的重要载体。随着中国历史的变迁，汉字也在不断发展。源自商朝的甲骨文字，至周朝的金文，再经历春秋战国时期的竹简文字和魏晋时期的石刻文字，汉字经历了千百年的沧桑变迁，最终演化成为如今通用的简体汉字。而其中，战国时期是中国古代历史上一个重要的时期，涌现出众多思想家、政治家和文化人物，其著作对于理解当时的社会变革、政治体制和文化思潮具有重要意义，如图 1.1 展示了战国竹简文字图像示例。然而，由于战争、时间的推移以及保存条件的限制，许多战国竹简丧失或严重损坏，使得人们无法充分了解这段重要历史时期的真实面貌。

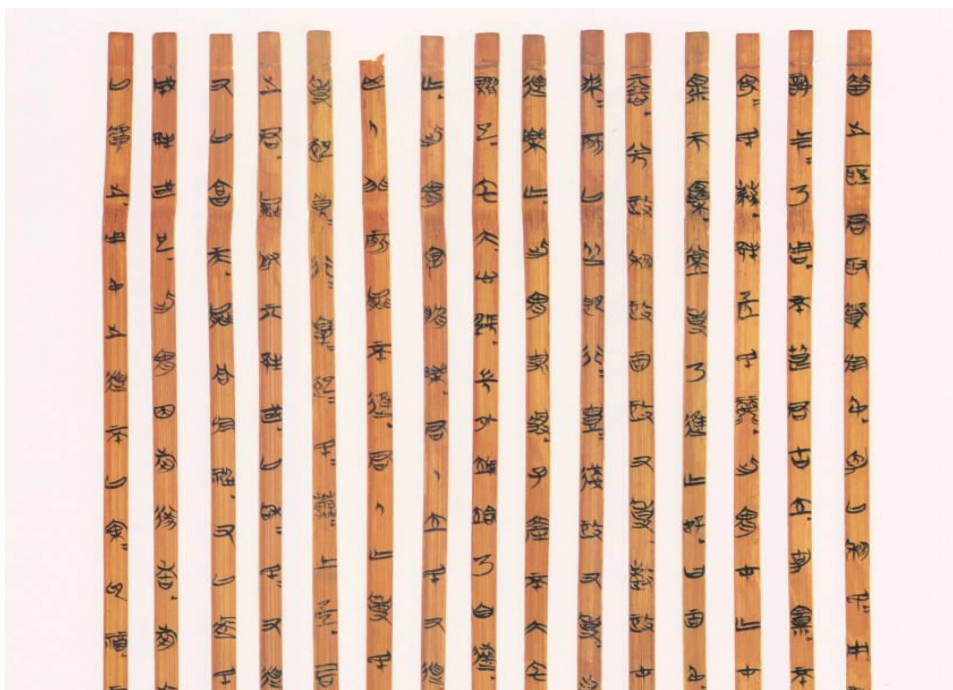


图 1.1 战国竹简文字图像示例

Fig. 1.1 Example of Bamboo Slip characters image in the Warring States Period

战国竹简文字中记录有大量的经书文献，对研究中国历史有重大的意义。但是，由于战国竹简文字与现代汉字的巨大差异，战国文字的整理和破译工作主要依靠专业人员，文字识读繁琐且效率不高。近年来，使用计算机来实现古文字识别的研究工作越来越多，极大地改善了人工处理古文字识别的重复行为，提高了古文字的破译效率。

而深度学习技术的引入为古文字的智能识别、整理和研究提供了全新的途径和方法<sup>[1]</sup>。近年来,已有不少研究将深度学习技术运用于甲骨文智能识别中,提高了甲骨文破译工作的效率。但是,关于战国竹简文字的智能识别工作还较少。

因此,在此背景下,本文对战国竹简文字的智能识别工作进行研究。将深度学习技术应用于战国竹简文字识别中,致力于快速而准确地识别出战国竹简文字,提升战国竹简文字的整理破译效率,有利于中国古代历史的保护与传承。

## 1.2 国内外研究现状

### 1.2.1 古文字识别研究现状

#### (1) 传统方法

在深度学习技术成熟之前,计算机识别古文字领域的研究主要依赖于传统图像处理技术和机器学习方法。2015年,Guo等提出了一种分层表示方法,通过结合Gabor相关的低级表示和稀疏编码器相关的中级表示来识别甲骨文字符和素描字符<sup>[2]</sup>。2016年,顾绍通提出利用甲骨文字形的拓扑结构进行识别。顾绍通对甲骨文字形的拓扑结构进行研究,探索了描述拓扑顶点之间关系的方法以构建特征信息并加以编码,最后计算拓扑距离来实现甲骨文的识别<sup>[3]</sup>。这种方法试图从甲骨文的结构层面捕捉其特征信息,而非仅仅依赖外观。尽管在数据量较少时,图论与拓扑学方法能够取得一定的识别效果,但随着待识别数据量的增加,识别准确率可能会逐渐下降。

2017年,刘永革等学者通过使用机器学习中的支持向量机(Support Vector Machines, SVM)方法来进行甲骨文识别<sup>[4]</sup>。使用SVM方法虽然获得了更好的特征表达能力,但是仍然依赖于研究者进行大量的手动特征工程。2019年,Kaur等提出了一种利用线性支持向量机分类器的婆罗米字符识别系统<sup>[5]</sup>。首先,提取字符图像像素的梯度信息,并将梯度的直方图存储为每个字符图像的特征向量。最后,使用线性SVM对婆罗米字符进行训练。2020年,Narang等提出了一种用于梵文古代手稿识别的光学字符识别系统<sup>[6]</sup>。利用SIFT技术和Gabor滤波特征提取技术的特征来提取手写梵文古字符的特征信息,并使用SVM分类器来实现字符识别。

综上所述,传统的古文字智能识别方法主要通过机器学习的方法。该方法可以在古文字识别工作中起到作用,但是需要手工设计特征工程,实现方式繁琐且在大型数据集上准确率不高。

#### (2) 深度学习方法

随着深度学习技术的兴起,尤其是卷积神经网络在图像识别领域的突破性成功,古文字的智能识别工作也越来越趋向于使用卷积神经网络的方法。2019年,王浩彬构

建了首个大规模甲骨文检测数据集 OBCD 和甲骨文单字符识别数据集 OBC306，并通过全卷积网络和特征金字塔网络搭建了基础的甲骨文字符算法框架<sup>[7]</sup>。针对甲骨文字符中存在样本分布不均衡的问题，通过使用循环生成对抗网络来合成少量样本的甲骨文字符类别，从而减少长尾效应带来的影响。

2020 年，刘梦婷从甲骨文的 10 种著作中搜集甲骨文图像，并制作了甲骨文拓片数据集 OBIS163。针对甲骨文识别算法的识别精度不高的问题，将算法中的方形卷积核更换为长条卷积核，并将长条的特征图进行叠加，使得甲骨文特征更加突出，最终取得了较好的识别效果<sup>[8]</sup>。Zhao 等人在 2020 年将深度学习技术应用于古籍手写水族文字的识别中，通过构建具有拉普拉斯结构的多层对抗神经网络，建立清晰的水字原始图像数据生成模型。并设计了通过聚类标注超参数的模型来提高古籍中手写水族文字的识别率<sup>[9]</sup>。同年，田园对战国竹简文字进行研究，整理了一个小型的战国竹简文字识别数据集 Bambooslips。针对战国竹简文字样本数量少的问题，提出了基于孪生网络的战国竹简文字小样本识别算法来对模型进行训练<sup>[10]</sup>。

2021 年，吴炫奇制作了一个格式统一且完整的金文数据集，并对金文数据集引入了模拟光照的效果，更好的还原了古代金文文字<sup>[11]</sup>。为了提高识别精度，在金文识别算法中添加了基于空间域注意力机制的空间变换网络。同年，Narang 利用深度学习技术来识别古代手稿中的梵文，利用卷积神经网络来训练手写梵文字体模型，实验证明使用 CNN 作为特征提取器的精度明显优于其他方法<sup>[12]</sup>。

2022 年，Chen 等设计了一种具有双重判别器的生成对抗网络 (DDGAN) 来还原古代彝族文献中不完整的字符<sup>[13]</sup>。DDGAN 将深度卷积生成对抗网络与古代彝族文献的对比判别器网络集成在一起。通过两个训练阶段，可以迭代优化古代彝族字符生成网络，得到文本生成器，根据对比判别器的损失，可以优化 DDGAN 生成模式。最终，DDGAN 模型可以生成字符来恢复古代彝族文献中缺失的笔画。

2023 年，Tang 等创建了目前最大的水族手稿字符图像数据集。并提出了一种基于注意力机制的轻量级人工神经网络模型 ShuiNet-A，该模型通过结合通道和空间维度来提取关键特征，并最终识别水族手稿字符<sup>[14]</sup>。通过多组实验验证了 ShuiNet-A 的有效性和稳定性。同年，Chandure 等创建了古印度文字 MODI 手写字符的数据集和开发了基于迁移学习的分类框架，利用深度卷积神经网络训练的模型作为预训练模型来传输权重并重新训练特征提取器，使用 SVM 作为分类器来对字符进行识别<sup>[15]</sup>。

综上所述，基于深度学习的方法可以有效实现古文字的智能识别任务。但是，基于深度学习的战国竹简文字识别研究还较少，主要存在以下问题：

① 数据集缺少。目前并没有公开的战国竹简文字识别数据集，目前对战国竹简文字的识别研究是小型数据集。由于战国竹简文字与现代汉字的巨大差异，战国文字的整理研究主要依靠专业人员，其标签的制作需要查阅专业的书籍进行标注。因此，制作一个战国竹简文字识别数据集是一个繁琐的事情。

② 竹木纹路、墨迹、部分字体缺损等干扰信息。与其他古文字不同，战国文字的载体是由竹子和木头制作的简，而竹木上的纹路会对识别产生影响。并且，竹简上的文字不是通过刀刻方式形成，而是通过人工手持毛笔书写上去的。在书写的过程中，偶尔会出现墨痕或墨迹溅落到竹简的情况。而由于漫长的时间，竹简上的部分文字也会被损坏。这些干扰信息会导致模型识别准确率下降。

③ 长尾化影响。古文字的数据集大多数是样本数量均衡的数据集，而战国竹简文字本身的分布是呈现长尾分布，存在严重的样本数量不均衡现象。样本多的样本有 300 多张图片，少的样本则只有 1-2 张图片，样本分布呈现长尾分布。这种样本数量不均衡现象会使模型的训练变得困难，训练后的模型更容易偏向于具有大量训练数据的头部，导致模型识别精度较低。

针对目前存在的问题，有必要对基于深度学习的战国竹简文字识别进行更加深入、更加具有针对性的研究。

### 1.2.2 战国竹简文字研究现状

在纸张发明之前，简牍是中国古代主要使用的书写材料，承载着丰富的历史文化信息。简牍使用最频繁的时期便是春秋到战国时期，成为当时人们记录思想、交流信息的重要工具。直到东晋末年，简牍才渐渐退出历史舞台，但其留下的文字信息，为后世研究提供了重要的史料和见证。然而，目前对于战国竹简文字的研究主要集中在古文字的形态结构、字体演变规律以及竹简中的文献内容等文学探讨上，很少涉及到战国竹简文字识别与人工智能文字识别的交叉研究。

战国时期出土的竹简主要包括郭店楚墓竹简（简称为“郭店简”）、上海博物馆藏品的楚简（简称为“上博简”）、安徽大学藏战国竹简（简称为“安大简”）、清华大学藏战国竹简（简称为“清华简”）等。

“郭店简”是 1993 年在中国湖北省荆门市纪山镇郭店一号战国楚墓内出土的竹简，共有 804 枚，其中 703 枚具有文字，另外还有 27 枚残缺的竹简，共记录 13000 余字。据考证，这些竹简的抄写时间不晚于公元前三百年，大约处于战国中期。在秦始皇焚书坑儒时期，大量先秦典籍被毁，而郭店楚简则成为了保存下来的极其重要的文献之一。这些竹简包括《老子》、《太一生水》、《缁衣》、《五行》等十六篇道家 and 儒家著作，提供了珍贵的历史资料。

1998 年，文物出版社出版了《郭店楚墓竹简》<sup>[16]</sup>，图 1.2 展示了郭店楚墓竹简示例。这部由荆门市博物馆编写的著作内容为多篇道家的著作，对于揭示中国古代文化和思想具有重要意义。



图 1.2 郭店楚墓竹简

Fig. 1.2 The Guodian Bamboo Slips

“上博简”是上海博物馆购入的一批战国楚竹书竹简，包括残简共计约 1700 余枚，记录约 35000 字。根据上海博物馆文物保护与考古科学实验室的科学分析，得出这批随葬品出土于战国晚期楚国贵族墓中。这批随葬品共计 80 余种，包括超过 20 篇书题。这些古籍内容涉及历史、哲学、宗教、文学、音乐、文字、军事等多个领域，主要是以儒家文献为主，包括同时也包括道家、兵家、阴阳家等经典著作。许多文献著作作为首次发现，提供了丰富的研究价值。

1997 年，上海博物馆完成了竹简的脱水和去除污色工作，并邀请了简牍文字专家对这批竹简进行整理注释工作，图 1.3 展示了上海博物馆藏战国楚竹书示例。2001 年，《上海博物馆藏战国楚竹书（一）》<sup>[17]</sup>出版，内容为儒家经典著作《孔子诗论》、《缙衣》以及《性情论》。到目前，上海博物馆藏战国楚竹书已经发布了九册，并公布了约 60 篇先秦著作。





表 1.1 出土战国竹简对比

Tab. 1.1 Comparison of unearthed Warring States Bamboo Slips

| 竹简名称  | 竹简数量   | 文字数量       | 出版数量 |
|-------|--------|------------|------|
| “郭店简” | 804 枚  | 约 13000 余字 | 1 册  |
| “上博简” | 1200 枚 | 约 35000 余字 | 9 册  |
| “安大简” | 1167 枚 | 约 36200 余字 | 2 册  |
| “清华简” | 2388 枚 | 约 50000 余字 | 13 册 |

由表 1.1 可以得出，“清华简”中竹简数量最多、文字数量最多且相关竹简研究出版书籍数量最多，这样的优势更有利于竹简文字的智能识别研究。因此，本文选择“清华简”作为研究对象和战国竹简文字识别数据集的制作原型。

### 1.3 论文的主要工作与章节安排

#### 1.3.1 主要工作

本文研究以战国竹简文字为研究对象，以竹简文字识别为目标。通过结合深度学习图像识别技术，将战国竹简文字识别智能化，为专业人员破译战国竹简文字工作提供帮助。综合考虑战国竹简文字识别任务中的难点，研究内容如图 1.6 所示，本文主要做了如下工作：

(1) 构建战国竹简文字识别数据集。由于战国竹简文字识别研究缺少公开的大型数据集，本文以“清华简”为研究原型，搜集整理了 1001 种战国竹简文字，共 19042 张图像数据。对图像进行数据标注、数据划分、数据预处理和数据增强操作，完成了战国竹简文字识别数据集 zgzi1001 及其 Many-shot 类、Medium-shot 类和 Few-shot 类数据子集的制作。

(2) 研究了基于空间自注意力机制的战国竹简文字识别算法。由于战国竹简上存在的竹木纹路、墨迹、部分字体断裂等干扰信息会对模型识别产生影响，本文改进了空间自注意力机制模块来使模型更加关注战国竹简文字的主体文字信息，从而提高了模型的识别性能。

(3) 研究了基于自监督双分支网络的战国竹简文字识别算法。由于战国竹简文字识别数据集本身呈现长尾分布，其严重的样本数量不均衡问题会造成模型训练困难和模型识别性能差。因此，本文将双分支网络与自监督对比学习算法结合来重点关注尾部类样本特征信息，从而削弱了数据集长尾分布造成的影响。

(4) 搭建了一个战国竹简文字识别系统，对系统功能进行了完整展示。

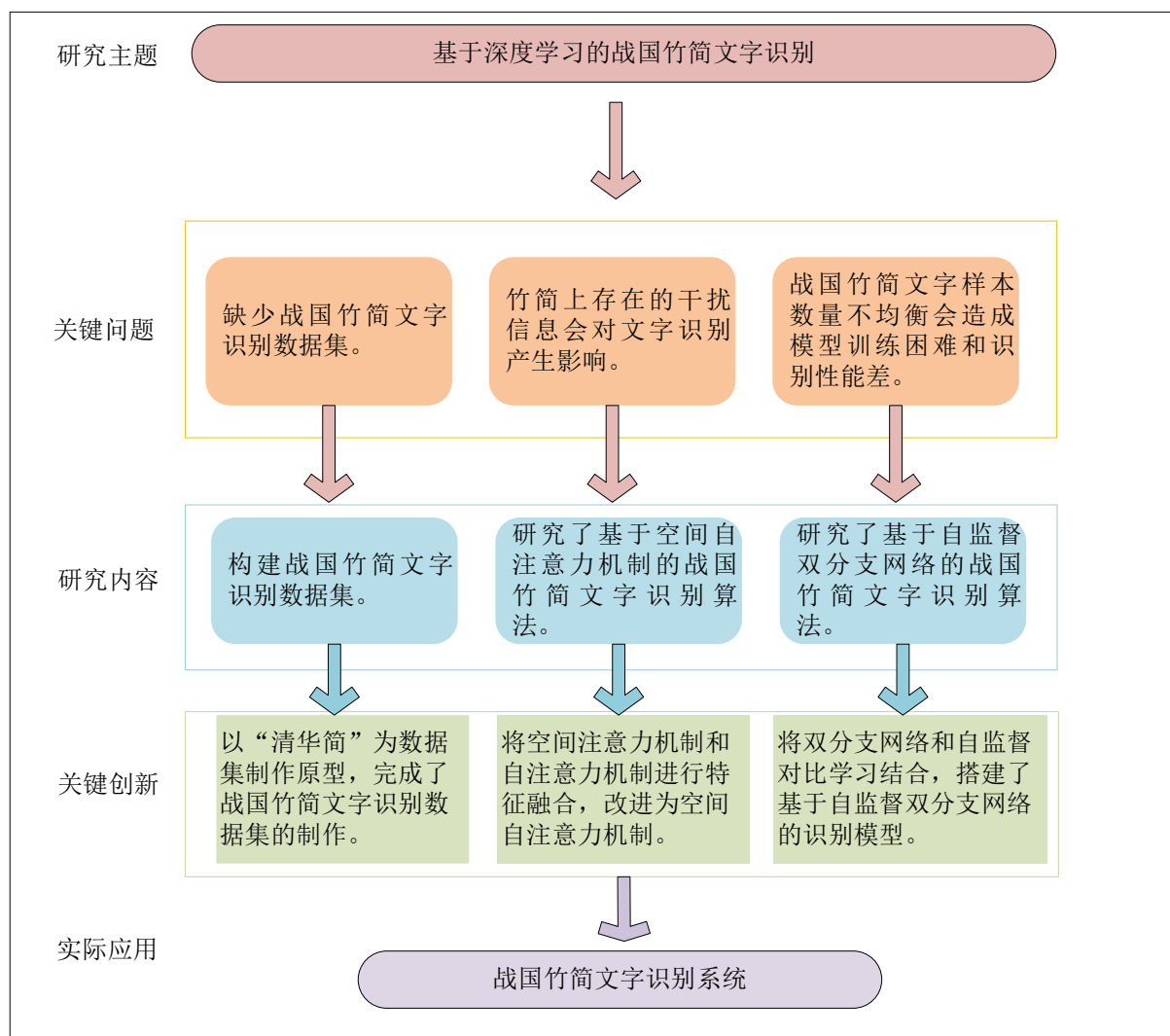


图 1.6 研究内容框架结构图

Fig. 1.6 Study content framework structure diagram

### 1.3.2 章节安排

本文共有七个章节，具体的安排如下：

第一章是绪论。首先，对战国竹简文字识别的研究背景与意义进行介绍；其次，综述了战国竹简文字研究现状并确定“清华简”为本文战国竹简文字研究原型；再次，阐述了古文字智能识别研究现状，总结了现有方法存在的不足；然后，简要说明了本文的主要工作；最后，说明了每个章节的具体安排。

第二章是对深度学习相关技术进行说明。首先，对深度残差网络和注意力机制的基本原理进行了详细的说明；然后，综述了长尾图像识别中的各种方法，分别总结这些方法的优点和局限性；最后，说明了本实验所使用的实验环境。

第三章是构建战国竹简文字识别数据集，为本文的第一个工作重点。首先，通过数据标注、数据划分、数据预处理和数据增强等操作完成了战国竹简文字识别数据集 zgzj1001 的制作；然后，对 zgzj1001 进行属性统计分析，得出 zgzj1001 是一个长尾分布、样本数量严重不均衡的数据集，为后续算法改进作铺垫；最后，对战国竹简文字上的识别难点进行说明，分析其对模型识别产生的影响，为后续算法优化作铺垫。

第四章是基于空间自注意力机制的战国竹简文字识别算法研究，为本文第二个工作重点。首先，对本文所改进的空间自注意力模块进行了详细的说明；其次，对本文改进的网络模型整体框架进行整体概括；再次，说明模型训练时的数据划分、评价指标和训练参数设置，并对深度残差网络系列算法的损失函数进行记录和对比分析；最后，通过测试集的实验结果对比分析，得出本章改进的模型可以削弱竹简上的干扰信息对文字识别带来的影响。

第五章是基于自监督双分支网络的战国竹简文字识别算法研究，为本文第三个工作重点。首先，说明了自监督对比学习的 SimSiam 算法的优点和基本框架结构；其次，对本章改进的网络模型主体框架进行详细的说明，主要包括卷积网络学习分支部分、自监督对比学习分支部分和损失函数计算；最后，进行模型实验结果的对比分析，并得出本章改进的模型削弱了数据集长尾分布造成的影响，提高了模型的识别精度。

第六章设计并应用了一个战国竹简文字识别系统。首先，选择 Flask 框架和 MySQL 数据库进行系统架构设置；然后，进行系统设计，主要包括系统结构设计、系统功能设计和数据库设计；最后，对系统应用和展示，展示了用户注册功能、用户登录功能、图像预处理功能和战国竹简文字识别功能，并对测试结果分析，验证了本系统在各种数据集上都保持着良好的识别效果，具有一定的实用价值。

第七章对本文主要工作进行了总结，并对战国竹简文字识别进一步研究提出展望。

## 2 深度学习相关技术

在古文字识别工作中，传统方法实现方式复杂且在文字样本数量较少的情况下识别性能较差。随着计算机科学技术的发展，深度学习技术能够高效的实现古文字智能识别工作，因此本文使用深度学习技术来实现战国竹简文字识别。首先，对深度残差网络的基本原理进行了详细的说明；其次，对注意力机制的原理进行说明，为后文算法改进作铺垫；再次，综述了长尾图像识别中的各种方法，分别总结这些方法的优点和局限性；最后，说明了本实验所使用的实验环境。

### 2.1 深度残差网络

深度学习被广泛认为是计算机科学与技术研究领域中的一个全新的发展方向。深度学习可以直接从图像数据中学习更抽象的特征信息，而不需要手动设计和提取特征信息<sup>[21]</sup>。卷积神经网络 (Convolutional Neural Networks, CNN) 是一种基于深度学习的模型，通过卷积、池化等操作提取图像特征，并通过多层神经网络进行分类和识别<sup>[22]</sup>。机器学习处理图像时，通常需要对图像进行数字化，再提取其中的特征。而 CNN 通过卷积中共享参数和稀疏连接的方式，可以同时实现对图像进行数字化和提取特征信息，极大地减少了模型的参数量和计算量<sup>[23]</sup>。相较于使用机器学习的图像识别方法，CNN 具有更好的特征提取和表示能力。因此，CNN 被广泛使用在图像识别任务中。

深度残差网络 (Deep Residual Network, ResNet) 是何恺明在 2015 年提出的深度卷积神经网络，通过引入残差连接来解决传统 CNN 在训练过程中出现的梯度消失和表达能力退化问题<sup>[24]</sup>。这一突破性的思想为深度学习领域带来了全新的范式，极大地提升了模型的训练效率和性能表现。因此，ResNet 模型不仅解决了深度学习中的重要问题，还为深度神经网络的设计和 optimization 指明了一条新的道路，对整个深度学习领域产生了深远的影响。ResNet 具有易训练和易优化、模型性能优越、模型参数和计算量小等优点，因此该网络在图像分类、目标检测、图像分割等计算机视觉任务中广泛使用。其残差学习单元如图 2.1 所示。

ResNet 的残差学习单元主要包括两个分支。第一个分支是输入  $x$  经过两个卷积块的操作得到输出  $F(x)$ ，第二个分支则直接将输入作为输出。然后，将这两个输出合并在一起形成最终的输出  $H(x)$ 。因此，网络的训练转换为学习残差部分，即  $H(x)-x$ ，而不是直接学习映射关系  $F(x)$ 。当模型需要学习到一个恒等映射  $H(x)=x$  时，只需要学习的参数部分为 0，这样就不会对输入数据造成额外的变换，从而保留了输入数据的有效信息。相对于其他网络模型，残差结构大大减少了训练时间。

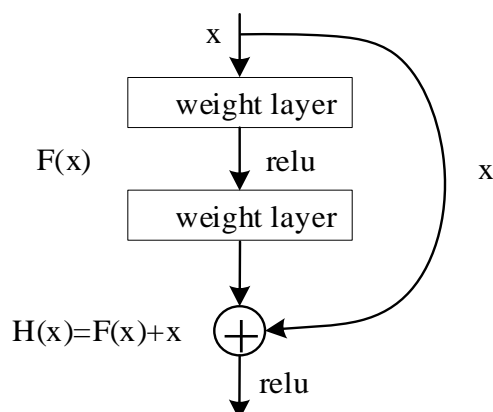


图 2.1 残差学习单元

Fig. 2.1 Residual learning unit

ResNet 共有 5 种不同深度的模型结构，分别为 ResNet18、ResNet34、ResNet50、ResNet101、ResNet152。而由于本文构建的战国竹简文字识别数据集中的图片尺寸固定为  $64 \times 64$ ，对于这种小尺寸图片不需要使用太深的卷积神经网络进行训练。因此，本文主要对其中的 ResNet18、ResNet34、ResNet50 等网络进行了实验，并进行了结果对比分析。表 2.1 为这三种网络的结构。

表 2.1 ResNet18、ResNet34 和 ResNet50 网络结构

Tab. 2.1 ResNet18, ResNet34 and ResNet50 network structures

| layer name | output size    | ResNet18  | ResNet34  | ResNet50  |
|------------|----------------|---|---|---|
| Conv1      | $32 \times 32$ |   | $7 \times 7, 64$ , Stride 2   |   |
| Max Pool   | $16 \times 16$ |   | $3 \times 3$ Max Pool, Stride 2   |   |
| Conv2_x    | $16 \times 16$ | $\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$   | $\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$   | $\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$    |
| Conv3_x    | $8 \times 8$   | $\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$ | $\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$ | $\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$  |
| Conv4_x    | $4 \times 4$   | $\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$ | $\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$ | $\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$ |
| Conv5_x    | $2 \times 2$   | $\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$ | $\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$ |
| APFS       | $1 \times 1$   | Average Pool, 1001-d FC, Softmax  |   |   |

ResNet 中的卷积块采用了残差学习单元，使得网络可以训练得更深。如表 2.1 所示，ResNet50 网络使用的参差结构为瓶颈结构 (Bottleneck)。在 Bottleneck 中使用两个  $1 \times 1$  的卷积核来改变维度，并且可以减少参数的计算量。卷积块采用了  $3 \times 3$  的卷积核，这种大小的卷积核具有较小的感受野和较强的非线性特性，可以更好地捕捉图像中的局部特征，并且具有较少的参数量和计算量。

## 2.2 注意力机制

在深度学习领域内，注意力机制已经成为了一个方便且高效的一个可嵌入式的模块。其主要原理是通过模拟人类的注意力聚焦机制，能够在处理大量数据时自动识别并集中处理最关键的信息<sup>[25]</sup>。这种机制应用在计算机视觉领域，便可以聚焦图像中重点信息，忽略图像中的噪声干扰信息，提高图像识别的准确率。本节主要对空间注意力机制和自注意力机制的原理进行说明，为后续算法的改进作铺垫。

### 2.2.1 空间注意力机制

通过引入空间注意力机制，神经网络可以更有效地捕捉图像中不同位置的重要信息，并且能够根据位置的重要性加权整合特征表示，从而更好地理解图像中的结构和语义信息<sup>[26]</sup>。空间注意力机制有助于提升模型对图像中重要区域的关注程度，并在图像处理任务中取得更好的性能，空间注意力模块如图 2.2 所示。

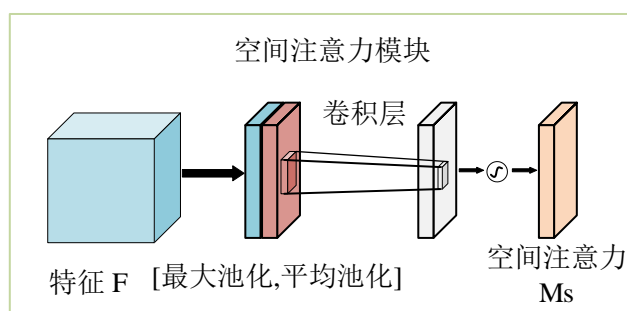


图 2.2 空间注意力模块

Fig. 2.2 Spatial attention module

空间注意力模块的具体实施为：首先，对输入的特征信息  $F$  进行最大池化和平均池化，最大池化通过选取池化窗口中的最大值来保留最显著的特征<sup>[27]</sup>，而平均池化则是选取池化窗口内数值的平均值来实现特征的压缩<sup>[28]</sup>；其次，对最大池化和平均池化的结果沿着通道维度进行拼接；最后，对拼接的结果进行卷积操作并通过 Sigmoid<sup>[29]</sup> 激活函数进行归一化，实现了对图像中重点位置信息的特征提取。

### 2.2.2 自注意力机制

传统的卷积神经网络通常使用不同尺寸的卷积核（如： $3 \times 3$ 、 $5 \times 5$ ）来实现特征提取操作，能够提取出图像中的特征信息<sup>[30]</sup>。但是，这种方式只考虑局部区域，而缺少了图像的全局信息。在2017年，wang等人提出非局部神经网络模型来实现提取图像的全局信息<sup>[31]</sup>。而其中的Non-local模块可以作为一个自注意力机制模块，自注意力机制模块结构如图2.3所示。

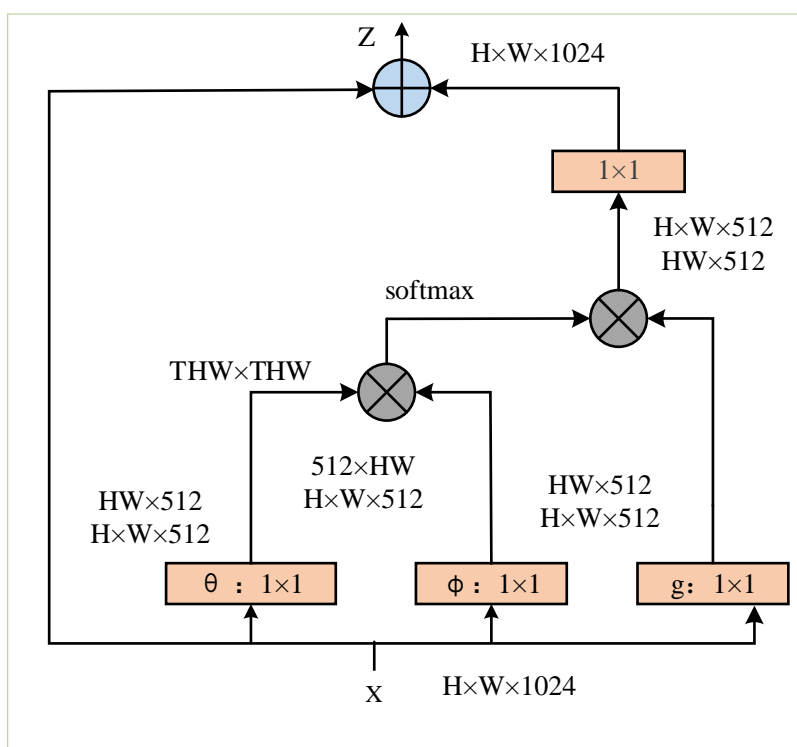


图 2.3 自注意力机制模块  
Fig. 2.3 Self-Attention module

自注意力机制的核心思想是通过对所有位置之间的关系进行建模，以计算每个位置之间的注意力权重，从而更全面地捕捉全局信息。自注意力机制可以使网络能够更好地理解输入数据中不同位置之间的长距离依赖关系，从而增强模型对全局上下文的感知能力。

### 2.3 长尾图像识别方法

长尾图像识别任务是一种特殊的图像分类问题，其数据集中的类别分布呈现长尾分布，即少数类的样本数量占据了总样本数量的大部分为头部类，而大多数类的样本

数量占据了总样本数量的小部分为尾部类，图 2.4 为常见长尾数据分布图。由于头部类别的样本数量较多，其对应的分类器权重更大，尾部类别的分类器权重较小。如果直接将长尾分布数据集用于传统的分类模型训练，可能会导致尾部类别特征学习不足，从而降低模型性能<sup>[32]</sup>。本文的研究对象为战国竹简文字，其文字的分布也符合这个分布规律，因此本文借鉴长尾识别方法来进行战国竹简文字的识别。

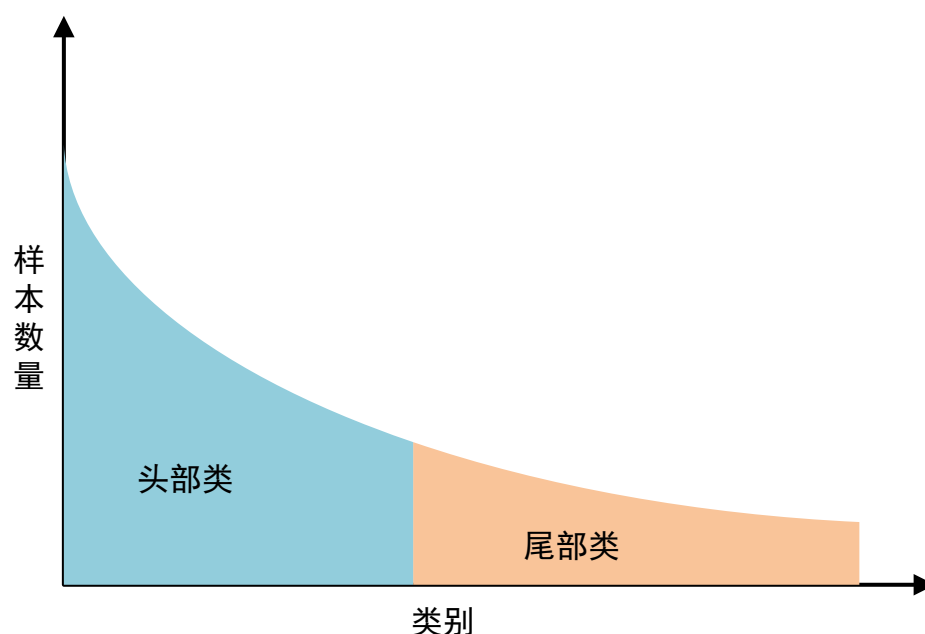


图 2.4 长尾数据分布图

Fig. 2.4 Long-tail data distribution chart

长尾图像识别方法主要包括：重采样、重加权、迁移学习和解耦训练等方法，本节对这些方法进行了对比分析，总结了现有方法的不足，为后文算法改进作铺垫。

### 2.3.1 重采样

在传统的深度学习训练中，数据采样部分采用随机抽样 (Random Sampling, RS) 方法，即在一个小批次内每个样本被采样的概率是相等的<sup>[33]</sup>。然而，在长尾分布数据集中使用 RS 方法，抽样的头部类数据远远大于尾部类数据，这使得训练好的模型在尾部类上效果不好。为了解决这个问题，有学者提出重采样技术来直接改变数据的分布。

重采样是在处理长尾识别问题上最简单也最常用的方法。重采样方法包括随机过采样 (Random Over-sampling, ROS)<sup>[34]</sup>和随机欠采样 (Random Under-sampling, RUS)<sup>[35]</sup>，ROS 在训练时随机重复采样尾部类的样本来平衡数据集，而 RUS 在随机重复采样中丢

弃大部分头部类的样本来平衡数据集。ROS 的缺点是可能造成尾部类样本的过拟合，而 RUS 会损失头部类样本数量，降低头部类样本的模型性能<sup>[36]</sup>。为了解决 ROS 的问题，Chawla 等人<sup>[37]</sup>提出了合成少数类的过采样方法 (Synthetic Minority Over-sampling Technique, SMOTE)。SMOTE 方法通过 K 近邻算法生成新的样本来补充样本的样本数量，这样的采样方式缓解了尾部类样本的过拟合问题。但是，由于 SMOTE 仅仅通过合成样本来增加少数类的数量，而并没有增加真实样本的信息量，这导致模型在处理真实数据时出现困难，模型泛化能力变差。

重采样通过直接改变样本的分布情况，来缓解长尾问题带来的影响。但是，无论是 ROS 还是 RUS 方法都会存在问题。由于尾部类样本数量较少，且尾部类样本间差异不大，使用 ROS 重复对尾部类样本进行重复采样，会导致尾部样本训练过程中出现过拟合现象。而使用 RUS 随机丢弃头部类样本，会导致头部类样本识别效果变差。因此，单独使用重采样方法不是最好的策略。

### 2.3.2 重加权

在传统算法中，训练数据集是均衡的，模型对头部和尾部类别的权重是相等的。而在长尾分布图像数据集训练时，各个类别的样本数量不同，而采样概率一样。当长尾率较大时，头部类别的采样的样本数量会比尾部类别的采样的样本数量多很多。如果模型对每个样本的权重是相等的，那么模型就会对头部类别产生很大的偏向。因此，头部类别会获得更大的权重，而尾部类别获得更小的权重，这样训练的模型对造成尾部类的识别精确率低。

为了有效解决长尾分布问题，重加权方法应运而生。重加权是一种处理长尾分布的方法，长尾分布通常由训练数据集中类别样本量不均衡导致<sup>[38]</sup>。这种方法通过对不同类别赋予不同的权重，有针对性地降低头部类别的权重并提高尾部类别的权重，以达到平衡样本分布的目的，重加权方法主要在算法网络的损失函数上进行优化<sup>[39]</sup>。

重加权的最直接实施方法是使用训练样本的标签频率对损失函数进行调整，即加权 softmax 损失 (Weighted Softmax Loss, WSL) 方法<sup>[40]</sup>。而与 WSL 方法不同，平衡 softmax 损失 (Balanced Softmax Loss, BSL) 方法将标签频率引入到模型预测中，以利用先验知识来减轻类别不平衡的偏差<sup>[41]</sup>。除了重新加权损失函数外，BSL 还提出利用先验知识来调整模型预测，来减少长尾分布造成的影响。与使用标签频率来进行重加权不同，何恺明等人提出焦点损失函数 (Focal Loss, FL) 方法来解决样本不均衡问题。FL 是一种基于预测难度进行重新加权的损失函数，具体来说 FL 为尾部类样本分配较高的权重，而为头部样本分配较低的权重，从而提高模型对尾部类样本的关注度<sup>[42]</sup>。

重加权方法可以在一定程度上缓解数据集长尾分布造成的影响，但是重加权技术本质是加强分类器学习，而会损坏模型学习到的深度特征的代表能力，因此重加权方法也不是最好的方法。

### 2.3.3 迁移学习

迁移学习是一种利用源领域（如数据集）的知识来增强目标领域模型知识的训练方法<sup>[43]</sup>。通过从头部类别中学到的丰富特征迁移到特性信息不足的尾部类别中，能够有效解决长尾分布数据集中数据不平衡的问题，表 2.2 介绍了在长尾图像识别中三种类型的迁移学习方法以及其代表方法。

表 2.2 迁移学习方法

Tab. 2.2 Transfer learning methods

| 方法                                      | 实现过程   | 类型         |
|---|--|------------|
| 元模型网络<br>(MetaModelNet) <sup>[44]</sup> | 首先在头部类别训练集上训练多样本模型，在类别子集用少量数据训练虚假少样本模型来模拟尾部类别。将所学虚假少样本模型映射到多样本模型学习元网络，再用学习的元网络将真实少样本模型在尾部类别上映射以获得更好性能。 | 头尾知识<br>迁移 |
| 领域特定迁移学习(DSTL) <sup>[45]</sup>          | 首先使用所有长尾样本进行表示学习对模型进行预训练，然后在更平衡的训练子集上进行微调。DSTL 逐渐将学习的特征转移到尾部类别，实现所有类别间更平衡的性能。                          | 模型预训练      |
| 多专家学习<br>(LFME) <sup>[46]</sup>         | 首先在几个不太不平衡的样本子集上训练多个专家，然后将这些专家蒸馏成一个统一的学生模型。  | 知识蒸馏       |

如表 2.2 所示为头尾知识迁移、模型预训练和知识蒸馏三种类型的迁移学习方法。其中，头尾知识迁移是指利用从头部类别中学到的知识和特征来辅助尾部类别的训练方法。模型预训练则是在大规模数据上进行预训练，然后将其迁移到目标任务中进行微调。知识蒸馏是一种将一个复杂模型的知识转移到一个简化模型的方法，从而提高尾部类别的识别性能。迁移学习的方法可以一定程度上提高长尾图像识别的性能，但是这种方法需要依靠大规模数据集，对于古文字识别领域，这种方法不太合适，因为在古文字识别中缺少大规模数据集，尤其缺少公开的战国竹简文字识别数据集。

### 2.3.4 解耦训练

传统的卷积神经网络训练是同时训练表示学习和分类器，而解耦训练将学习过程分解为表示学习和分类器训练两个阶段，是一种针对长尾学习的通用范式。

(1) 2020 年, Kang 等人在论文 *Decoupling* 中提出将表示学习中的特征提取部分和分类器解耦的思想, 即解耦训练, 这是长尾识别两阶段训练的开山之作<sup>[47]</sup>。在此之前, 长尾识别模型的训练模型都是单阶段, 即同时训练表示学习阶段和分类器训练阶段。而通过实验发现, 将表示学习和分类器训练分开进行训练, 可以在长尾数据集上取得更好的识别性能。

解耦训练是一种直观而有效的长尾分布模型学习范式, 将网络模型的训练分为两阶段进行。第一阶段进行表示学习的训练, 在初始数据集上通过随机采样的方式进行模型训练, 得到一个预训练模型, 包括经过训练的特征提取部分和训练器部分。第二阶段进行分类器的训练, 加载第一个阶段训练得到的预训练模型, 得到其中的特征提取部分参数, 并固定特征提取部分的可学习参数, 然后对分类器进行微调, 来达到更好的识别性能。

实验证明, 解耦训练可以显著改善长尾识别任务的性能。但是这种两阶段训练方法要分两步对模型进行训练, 模型训练繁琐且模型优化困难。

(2) 2020 年, Zhou 等人提出了双分支网络 (Bilateral Branch Network, BBN), 同时兼顾模型的表示学习和分类器训练<sup>[48]</sup>。BBN 也提出了解耦思想, 但是与解耦训练不同, BBN 是单阶段模型, 表示学习部分和分类器训练是同时进行。因此, BBN 的模型训练更加简单与灵活。同时, BBN 在实验中验证了重采样和重加权等类别重新平衡方法的有效性和不足性。这些方法显著提升了深度网络分类器学习的性能, 然而这些类别重新平衡方法损害了模型的表示学习。

BBN 的框架结构如图 2.5 所示。BBN 主要是由传统学习分支 (Conventional Learning Branch)、重新平衡分支 (Re-Balancing Branch) 和累积学习 (Cumulative Learning) 策略组成。传统学习分支采用随机采样来模拟原始的长尾训练分布, 有利于模型进行特征信息提取。而重平衡分支则应用逆采样器, 在每个小批量中采样更多的尾部类别样本, 以改善尾部类别的性能, 有利于实现分类器的训练。而累积学习策略是控制传统学习分支和重新平衡分支在训练过程中的训练程度, 通过一个学习参数  $\alpha$  来控制两个分支共享权重信息的动态融合, 使得 BBN 的学习重心逐渐从关注图像的特征信息提取逐步到模型的分器训练。

BBN 的双分支结构可以改善尾部类的特征提取能力, 提高了尾部类样本的识别准确率, 从而缓解了数据集的长尾分布问题。经过实验论证, BBN 在多个长尾数据集上可以达到较高的识别性能。但是, 如果尾部类的样本数量太少, 在重平衡分支应用逆采样器对模型性能的提升有限, 且重复对尾部类样本进行抽样可能会导致模型出现过拟合和泛化能力不足。

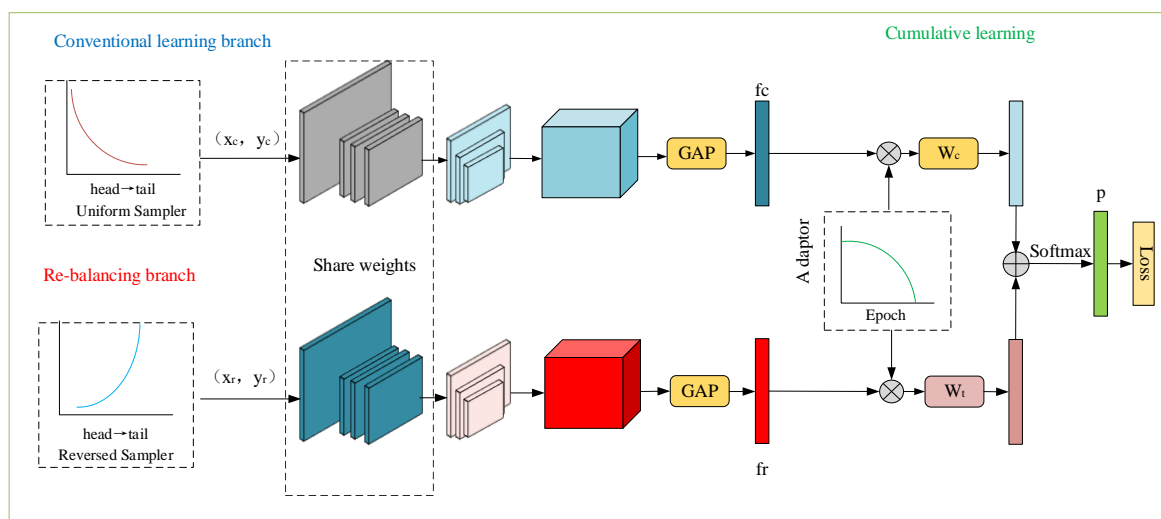


图 2.5 BBN 框架结构

Fig. 2.5 BBN framework structure

综上所述，为了解决长尾识别问题，重采样、重加权、迁移学习和解耦训练都可以起到一定的作用，但是这些方法还是存在一些问题：

(1) 重采样技术直接通过改变数据的分布方式，来缓解长尾数据分布造成的影响，是一种简单易操作的方法。但是，使用重采样技术会导致尾部类样本训练出现过拟合现象和损失一部分头部类样本的识别精度。

(2) 重加权技术主要对损失函数进行改进，为不同类型的样本分配不同的权重，削弱头部类别的权重，提高尾部类的权重。但是，使用重加权技术会导致模型的特征学习能力下降。

(3) 迁移学习通过从头部类别中学到的丰富特征迁移到特性信息不足的尾部类别中，能够解决训练中数据不平衡的问题。但是，本文的研究对象为战国竹简文字，缺少大规模的训练集来进行特征迁移。因此，不能使用迁移学习方法来原因战国竹简文字识别上的长尾分布问题。

(4) 解耦训练是一种直观而有效的长尾分布模型学习范式，通过将模型训练分为表示学习和分类器学习两部分，可以解决长尾识别上的问题。其中，Decoupling 和 BBN 都是解耦训练的形式。Decoupling 方法模型训练困难且模型难以优化，BBN 方法简单易训练，但是在重平衡分支对尾部类样本进行重复采样可能会导致模型过拟合和泛化能力不足。

因此，为了解决战国竹简文字上存在的长尾分布问题，有必要对长尾识别算法进行改进。

## 2.4 实验环境

本文中的所有实验均在表 2.3 的主机环境中进行。

表 2.3 实验环境

Tab. 2.3 Experiment environment

| 项目        | 配置                   |
|-----------|----------------------|
| 操作系统      | Windows 10 64 位      |
| 处理器 (CPU) | Intel i9-12900       |
| 显卡 (GPU)  | GeForce RTX3090 24G  |
| 编译器       | PyCharm              |
| 算法架构      | Pytorch1.12+CuDa11.4 |
| 编程语言      | Python3.9            |

## 2.5 本章小结

本章主要说明了深度学习相关技术的原理，引出本文的研究方案。本章主要做了以下工作：

(1) 对深度残差网络的基本原理进行了详细的说明。本文后续工作将以深度残差网络为基准网络，对其进行改进和搭建战国竹简文字识别模型。

(2) 对空间注意力机制和自注意力机制的原理进行了说明，为后续本文对注意力机制改进作铺垫。

(3) 综述了长尾图像识别的常用方法。对重采样、重加权、迁移学习和解耦训练等长尾识别方法进行了对比分析，总结了现有方法的不足，分析了其不适用于战国竹简文字识别任务的原因，为后续本文对算法的改进作铺垫。

(4) 简要说明了本文实验用到的基础主机环境，本文所有模型都是在此主机环境中进行训练。

### 3 构建战国竹简文字识别数据集

由于目前的战国竹简文字识别工作没有公开的数据集，因此，本章以“清华简”为数据集研究对象来完成数据集的制作工作。首先，搜集整理了 1001 种战国竹简文字，共 19042 张图像数据；其次，对图像数据进行数据标注、数据划分、数据预处理和数据增强操作；最后，完成构建战国竹简识别数据集 zgzt1001。本章主要说明战国竹简文字识别数据集的制作过程、数据集的属性统计以及识别上存在的挑战。

#### 3.1 数据集制作

本节主要进行数据集标注、数据集划分、数据预处理、数据增强等操作。

##### 3.1.1 数据集标注

数据集的图像数据来源于《清华大学藏战国竹简》中的部分内容，包括《尹至》、《尹诰》、《程寤》等多篇记载的文献。将古汉字逐一采集，并对图像数据进行人工标注。搜集到 1001 种战国竹简文字，共 19042 张图像数据。

通过查阅战国文字文献资料，将 19042 张战国文字图片放入到其对应的现代注释的文件夹中，即完成战国竹简文字识别数据集 zgzt1001 的标注工作。战国竹简文字部分类别示例图如图 3.1 所示。



图 3.1 战国竹简文字部分类别示例图

Fig.3.1 Example diagram of some categories of Bamboo Slips in the Warring States Period

### 3.1.2 数据集划分

在数据标注完成后，对战国竹简文字数据集进行划分。为了更好的评估数据集，按照每个类别图片数量大于 100 张图片分为 Many-shot 类，每个类别图片数量为 20-100 张图片分为 Medium-shot 类，每个类别图片数量少于 20 张的分为 Few-shot 类。zgzt1001 数据集呈现出长尾分布，出现严重的样本数量不均衡现象。其中 Many-shot 共 43 类战国文字，Medium-shot 共 297 类战国文字，Few-shot 共 661 类战国文字。

### 3.1.3 数据预处理

为了确保深度学习模型能够接收到高质量、格式规范的输入数据，需要对数据集进行数据预处理操作<sup>[49]</sup>。针对本文收集到的战国竹简文字数据集，主要进行了图片尺寸规范和标准化处理。

#### (1) 图片尺寸规范

原数据集中图片大小都不是统一的，为了不影响深度学习模型的训练，需要保证输入数据的统一性，图片尺寸规范操作如图 3.2 所示。首先，使用图像边缘的像素值将图像填充成正方形；然后，使用将其缩放 (Resize) 到  $64 \times 64$  大小。这样防止图像直接通过 Resize 操作产生的失真问题，保证了模型可以获得准确的数据输入。图 3.2(b)为未进行像素填充直接进行 Resize 操作，图 3.2(c)为像素填充后进行 Resize 操作，可以看出进行像素填充后缩放可以得到更加完整的图像。

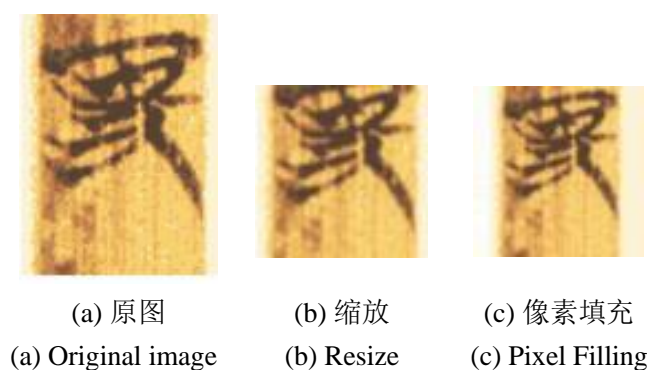


图 3.2 图片尺寸规范

Fig.3.2 Specifications for image size

#### (2) 标准化处理

在深度学习任务中图像标准化是一个非常重要的预处理步骤<sup>[50]</sup>。为了减少战国竹简文字数据中部分图片的噪声问题和加快模型训练，对战国竹简文字数据集进行标准

化处理。首先，计算每个图片像素通道的均值和标准差；然后，使每个图片的像素减去该通道的均值；最后，将得到减去的值除以标准差，处理公式如公式 (3.1) 所示：

$$x_{nor} = \frac{x - \text{mean}(x)}{\text{std}(x)} \quad (3.1)$$

其中， $x$ 表示像素的原始值， $x_{nor}$ 表示经过标准化处理的像素值， $\text{mean}(x)$ 表示计算得到的像素均值， $\text{std}(x)$ 表示计算得到的标准差。

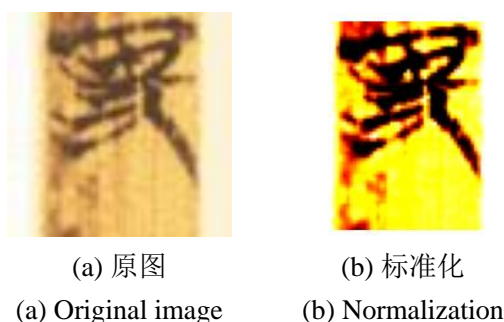


图 3.3 标准化处理

Fig.3.3 Standardized processing

标准化处理如图 3.3 所示，进行图像标准化处理以后，可以有效抑制战国竹简文字上竹木纹理造成的干扰信息。

### 3.1.4 数据增强

通常神经网络模型需要大量的图片数据进行训练，通过大规模数据集的训练可以获得更好的效果。进行数据预处理后，对处理好的图片数据集进行数据增强操作。在 zgj1001 数据集中，Few-shot 类有 661 种战国文字是小于 20 张的类别，因此本文主要对其进行数据增强操作。

数据增强是一种通过人工生成相似数据样本来扩充小样本数据集的技术，能够有效地解决数据不足的问题，并广泛应用于深度学习的各个领域，传统的数据增强技术包括翻转、旋转、缩放、裁剪、亮度变化等方法<sup>[51]</sup>。然而，对于文字数据集，使用翻转和旋转的方法会影响文字的可读性。考虑到古文字的特性，在不破坏战国文字的字体形态和书写风格下，本章主要使用亮度增强技术、腐蚀增强技术和膨胀增强技术。

数据增强操作如图 3.4 所示。通过亮度增强可以使模型识别到不同亮度下的战国文字，提高了模型的鲁棒性。腐蚀增强可以使文字部分更加突出，膨胀增强可以削弱竹简上的干扰信息。并且通过腐蚀增强和膨胀增强可以模拟战国文字笔画断裂和粘连，进而增加了样本空间，增强了模型的泛化能力。

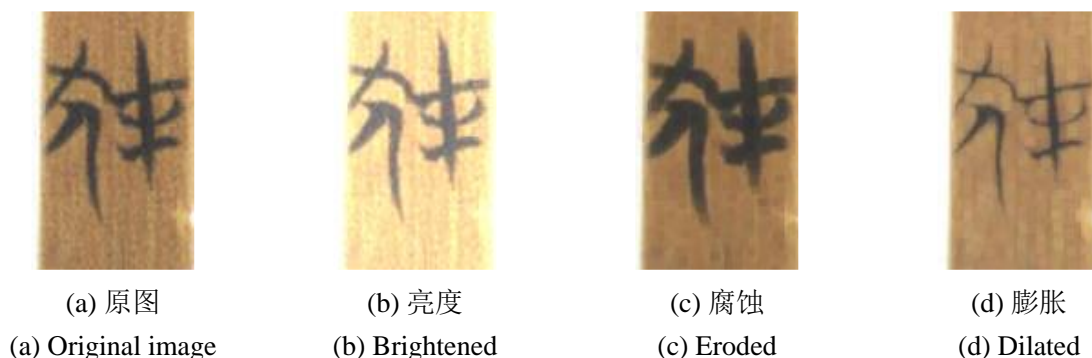
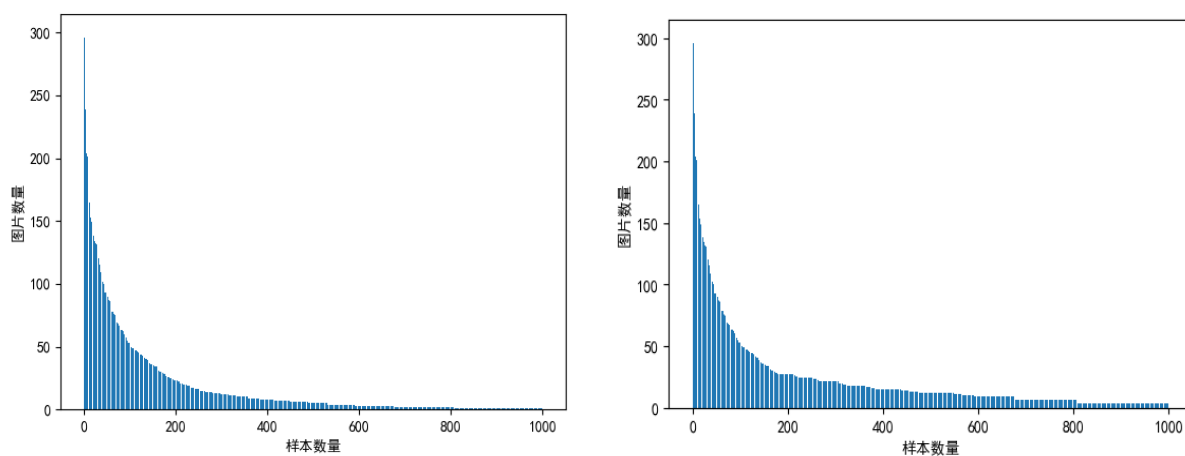


图 3.4 数据增强操作

Fig.3.4 Operations of data augmentation

经过数据增强后，数据集 zgzt1001 的样本总量从 19042 张图像数据增加至 26,564 张图像数据。增广前的样本分布图如图 3.5(a)所示，增广后的样本分布图如图 3.5(b)所示。在 zgzt1001 的样本分布图中可以看出，战国竹简文字样本分布呈现长尾分布，数据集样本分布不均，大部分的样本只有少量的样本数量。



(a) 增强前的样本分布图

(b) 增强后的样本分布图

(a) Sample distribution map before enhancement

(b) Sample distribution map after enhancement

图 3.5 样本分布图

Fig.3.5 Sample distribution map

### 3.2 数据集属性统计

对增强后的样本数量进行统计，四种数据集的详细信息如表 3.1 所示。其中，类别数指的是该数据集中不同类别的数量；样本总数指的是该数据集中所有样本的数量；样本最小类数指的是该数据集中样本数目最少的一类样本的数量；样本最大类数指的

是该数据集中样本数目最多的一类样本的数量；类间平均数指的是该数据集中所有类别样本数量的平均数。

表 3.1 战国竹简文字识别数据集基本信息

Tab. 3.1 Basic information of the Warring States Bamboo Slips character recognition dataset

| 数据集         | 类别数  | 样本总数  | 样本最小类数 | 样本最大类数 | 类间平均数  |
|-------------|------|-------|--------|--------|--------|
| zgzj1001    | 1001 | 26564 | 9      | 300    | 26.54  |
| Many-shot   | 43   | 6825  | 102    | 300    | 158.72 |
| Medium-shot | 297  | 11391 | 20     | 100    | 38.35  |
| Few-shot    | 661  | 8348  | 9      | 19     | 12.63  |

其中 Many-shot 类共有 43 种战国竹简文字类，共 6825 张样本图片，占总数据集的 25.69%，在这 43 种类别中，样本最大数可以达到 300 张，而最小的有 102 张。而在 Few-shot 中，有 8348 张样本图片，占总数据集的 31.42%，样本最大数为 19 张，最小样本数只有 9 张。从这些数据可以看出，战国竹简文字数据集在包含高频出现的战国文字的同时，数据集中还有大量低频出现的战国文字。数据集呈现严重的数据不均衡现象，类别之间样本数量差距很大，这样的样本分布的数据集会对模型的训练产生很大的影响。

### 3.3 识别难点

竹简文字作为古人在竹简上书写的文字，在本质上属于手写体，一些战国竹简文字的实例如图 3.6 所示。图 3.6(a)为竹木纹路干扰，图 3.6(b)为墨迹干扰，图 3.6(c)为部分字体断裂的文字，图 3.6(d)为不同书写风格的字体。

直接使用现代的手写体识别方法不适用于战国竹简文字的识别，其原因主要是：

(1) 竹简文字的载体是竹木，竹木本身具有天然的纹路，这种纹路对文字识别造成干扰，并且竹简的纹理可以使字形变得模糊或者略微变形，因此对字体的识别有一定影响。竹木纹路干扰如图 3.6(a)所示。

(2) 竹简上的文字并非通过刀刻方式形成，而是通过人工使用毛笔书写的。由于书写过程中难免会有墨痕或墨迹溅落到竹简上，这些墨迹给古文字的辨认带来了一定的挑战，对文字识别造成干扰。墨迹干扰如图 3.6(b)所示。

(3) 战国时期距今时代久远，许多竹简最初都是作为陪葬品被埋葬在地下。随着时间的推移，竹简会因长时间的埋藏而腐化和损坏，导致上面的文字残缺不全，这些因年代久远和埋藏造成的损伤给识别来了一定的困难。部分字体断裂如图 3.6(c)所示。

(4) 竹简上记录有不同内容的文献，而每一篇章的内容由不同的人书写。因此，就算是同一个战国文字在竹简上也有不同的书写风格，这要求识别模型具有更好的识别能力和泛化能力。不同书写风格的字体如图 3.6(d)所示。



图 3.6 战国竹简文字样例

Fig.3.6 Example of Warring States Period Bamboo Slip characters

### 3.4 本章小结

目前战国竹简文字的智能识别工作还较少，且未有公开的数据集。因此，本章首先通过数据标注、数据划分、数据预处理和数据增强等操作完成了战国竹简文字识别

数据集 zgzt1001 的制作；其次，对 zgzt1001 进行数据集统计分析；最后，对 zgzt1001 进行识别难点分析。主要工作总结如下：

（1）完成数据集制作。首先，对搜集到的战国竹简文字图片进行数据集标注工作；其次，对战国竹简文字识别数据集按照样本数量划分成 Many-shot 类（类样本数量 > 100），Medium-shot 类（类样本数量为 20~100）和 Few-shot 类（类样本数量 < 20）；然后，对数据集进行了图片尺寸规范和标准化处理两种预处理操作；最后，对数据集进行亮度增强技术、腐蚀增强和膨胀增强等增强操作，即完成战国竹简文字识别数据集 zgzt1001 的制作。

（2）对 zgzt1001 进行数据集属性统计。数据集属性统计发现，zgzt1001 是一个长尾分布的数据集，呈现严重的数据不均衡现象，类别之间样本数量差距很大。这样的样本分布的数据集会对模型的训练产生很大的影响，为后续算法改进作铺垫。

（3）对 zgzt1001 进行识别难点分析。对数据集识别难点分析得出，竹简文字上的竹木纹路、墨迹、部分字体断裂等特征都会对文字识别造成影响，为第四章算法的优化作铺垫。

## 4 基于空间自注意力机制的战国竹简文字识别算法研究

从第三章中战国竹简文字的特征可知，竹简文字上的竹木纹路、墨迹、部分字体断裂等特征都会对文字识别造成影响，导致模型识别准确率下降。因此，针对这一问题。首先，本章将空间注意力机制与自注意力机制结合，改进为空间自注意力机制来关注战国文字的主体特征；其次，对 ResNet50 进行改进，将空间自注意力机制添加到 ResNet50 最后一个卷积层，削弱了竹简文字上的干扰信息带来的影响，进而提高模型的识别性能；最后，搭建了基于空间自注意力机制的战国竹简文字识别模型。

### 4.1 改进空间自注意力模块

对于战国竹简文字识别任务，其竹简文字上的竹木纹路、墨迹、部分字体断裂等干扰信息都会对文字识别造成影响。因此，本节改进空间自注意力机制 (Spatial Self-Attention, SSA) 来解决这个问题，改进后的 SSA 框架结构如图 4.1 所示。

SSA 模块主要由空间注意力部分 (Spatial Attention Part)、自注意力部分 (Self-Attention Part) 和特征融合 (Feature Fusion) 组成。通过融合空间注意力机制 (Spatial Attention) 与自注意力 (Self-Attention) 机制来关注竹简文字上的主体特征，忽略干扰信息对文字识别造成影响，从而提高战国竹简文字的识别性能。

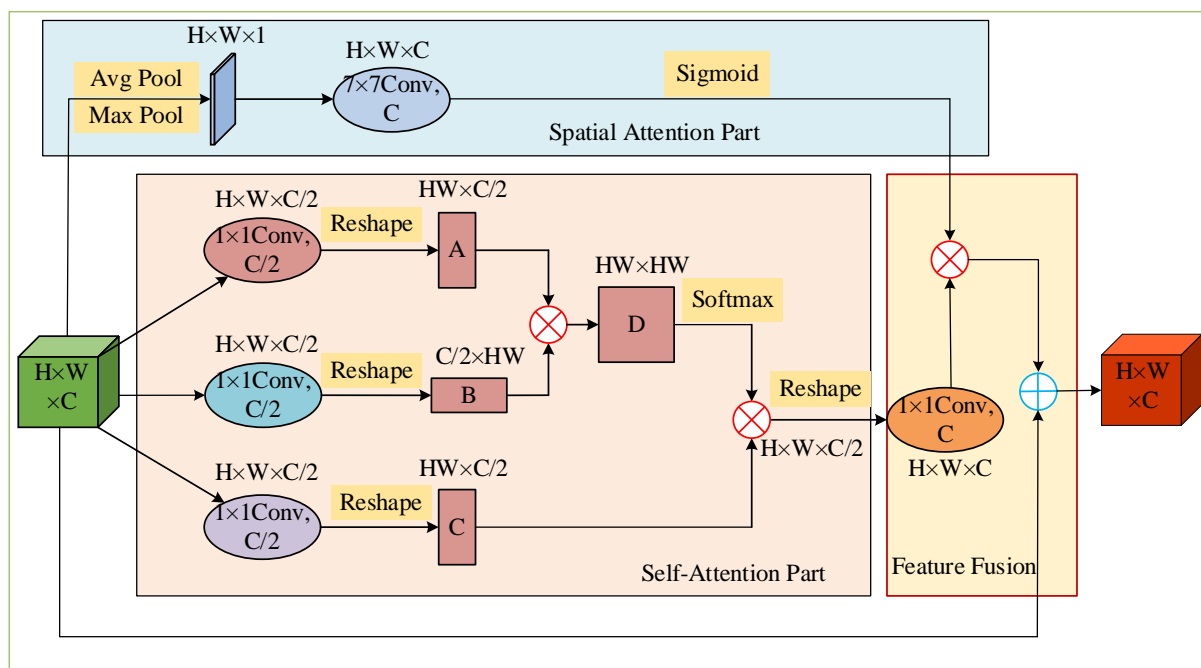


图 4.1 SSA 框架结构

Fig. 4.1 SSA framework structure

#### 4.1.1 空间注意力部分

空间注意力部分通过分析特征图在空间维度上的内在联系，生成具有空间注意力机制的特征图，从而关注了战国竹简上的文字特征信息，其框架结构如图 4.1 中 Spatial Attention Part 部分。

对于一个维度为  $H \times W \times C$  的特征图，首先在通道维度进行平均池化 (Avg Pool) 和最大池化 (Max Pool) 操作，输出维度为  $H \times W \times 1$  的空间特征；然后，将这两组特征图沿通道维度合并，形成一个更丰富的特征表示；其次，使用卷积核大小为  $7 \times 7$ 、padding 为 3、stride 为 1、输出通道为  $C$  的卷积核进行卷积操作，将特征图上采样为  $H \times W \times C$ ；最后，对特征图使用 Sigmoid 函数将空间特征信息映射到 0 到 1 之间，有助于关注更重要的空间位置。

空间注意力机制计算如公式 (4.1) 所示：

$$SP(F) = S\left(f^{7 \times 7}([F_{avg}; F_{max}])\right) \quad (4.1)$$

其中， $F$  是原特征图， $SP(F)$  表示空间注意力机制提取到的特征图， $S(\cdot)$  表示 Sigmoid 函数， $f^{7 \times 7}$  表示卷积核大小为  $7 \times 7$  的卷积运算， $F_{avg}$  和  $F_{max}$  表示 Avg Pool 和 Max Pool 操作后的特征图。

#### 4.1.2 自注意力机制部分

自注意力机制部分关注了战国竹简文字的全局特征信息，使网络能够更好地理解输入数据中不同位置之间的长距离依赖关系，从而削弱了战国竹简上的干扰信息对模型识别造成的影响，其框架结构如图 4.1 中 Self-Attention Part 部分。

对于一个输入维度为  $H \times W \times C$  的特征图，首先使用三个  $1 \times 1$  的卷积核对特征图进行降维到  $H \times W \times 1/2C$ ；其次，对三种特征图进行重塑成  $A$ 、 $B$ 、 $E$  特征图；然后，将  $A$  特征图与  $B$  特征图进行矩阵相乘来生成  $D$  特征图，并应用 softmax 函数对  $D$  特征图进行归一化；最后，将  $D$  特征图与  $E$  特征图进行矩阵融合成自注意力机制特征图，从而获取了图像的全局特征。

自注意力机制计算如公式 (4.2) 所示：

$$SA(F) = \frac{\exp(A_i \otimes B_i)}{\sum_{i=1}^N \exp(A_i \otimes B_i)} \otimes E_i \quad (4.2)$$

其中， $\otimes$  表示元素乘法， $SA(F)$  表示经过自注意力机制计算后的特征图的特征信息， $A_i$  表示  $A$  特征图中的特征信息， $B_i$  表示  $B$  特征图中的特征信息， $E_i$  表示  $E$  特征图中的特征信息。

### 4.1.3 特征融合

将空间注意力机制部分提取到的特征与自注意力机制提取到的全局特征进行特征融合，从而生成更加丰富和有用的特征表示。

具体实施为：首先，通过空间注意力部分提取到战国竹简文字上的关键特征；其次，通过自注意力部分关注战国竹简文字的全局信息；最后，将空间注意力部分提取到的空间特征映射到自注意力部分上的特征图上，将其通过跳跃连接加回到原始特征图中。最终的注意力特征图表达式为公式 (4.3) 所示：

$$F_{SSA} = SP(F) \otimes SA(F) + F \quad (4.3)$$

其中， $F$ 是原始特征图， $F_{SSA}$ 表示空间自注意力特征图， $\otimes$ 表示元素乘法， $SP(\cdot)$ 是空间注意力操作， $SA(\cdot)$ 是自注意力操作。通过对空间注意力特征、自注意力特征和原始特征的加权融合，可以获取更加丰富的特征信息。

## 4.2 改进网络模型主体框架

由于战国竹简文字上存在的竹木纹路、墨迹、部分字体断裂等影响因素会导致深度学习算法模型识别性能不佳，因此本文改进基于空间自注意力机制的战国竹简文字识别算法。具体来说，选取 ResNet50 为基准网络，并将空间自注意力机制添加到 ResNet50 最后一个卷积层后，即对最后的特征图添加了空间自注意力机制。所搭建的 SSA-ResNet50 模型框架结构如图 4.2 所示。

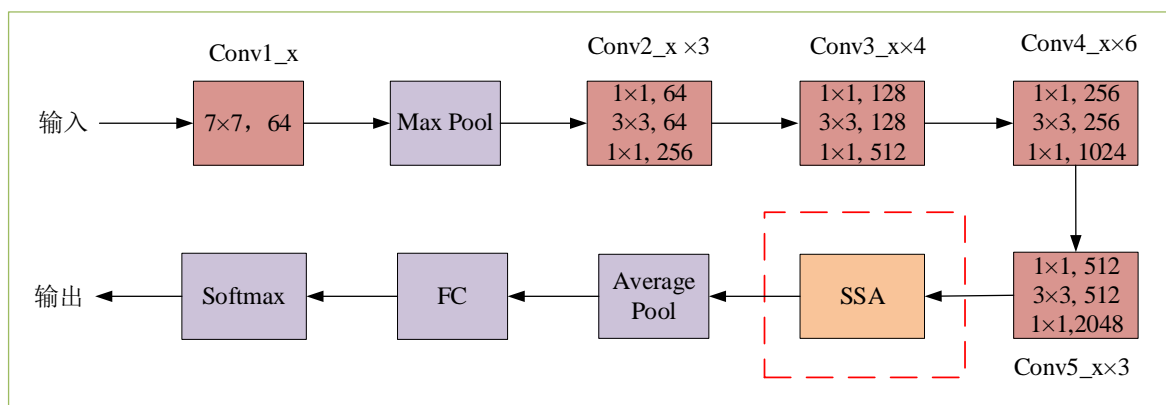


图 4.2 SSA-ResNet50 框架结构

Fig. 4.2 SSA-ResNet50 framework structure

其中的 4 个残差块为 Conv2\_x、Conv3\_x、Conv4\_x、Conv5\_x，参数块数目分别为：3、4、6、3。通过在最后一个参数块后面加入了 SSA 模块，将卷积层中提取到的

特征图转化为空间自注意力机制特征图。空间自注意力机制特征图中重点含有战国竹简文字的主体文字信息，而忽略其中的干扰信息，从而提高了模型算法的识别性能。

### 4.3 模型训练及参数设置

#### 4.3.1 数据划分

在进行模型训练之前，将战国竹简文字识别数据集 zgzt1001 划分成训练集、验证集、测试集。训练集用于训练模型的权重，以便模型能够从中学习到数据的特征和规律；而验证集则用于通过交叉验证的方式来选择最优的模型，测试集则是用于评估模型的性能，通过提供一个独立的数据集来检验模型在真实场景下的表现，确保模型具有较强的泛化能力。

由本文第 3.2 节中数据集属性统计可知，zgzt1001 本身样本数分布不均衡，且样本最小类数为 9 张图片。按照一定比例对数据集进行随机划分会导致某些样本没有验证集和测试集的图片。因此，对战国竹简文字数据集 zgzt1001 按照每个样本类抽取 5 张图片作为验证集，该验证集的分布是均衡的，有助于对模型训练效果进行评估。最后，数据划分后训练集共 20558 张图片，验证集共 5005 张图片，测试集共 1001 张图片。

#### 4.3.2 评价指标

为了评价一个模型的好坏，需要设计指标去评价模型的性能。在选择一个模型时，模型参数量 (Params) 和模型计算量 (Floating-point Operations, FLOPs) 是两个基础的指标。模型参数量是指在模型中所有需要学习的参数数量，计算方式为将每个维度的参数相乘并相加，参数量是衡量一个模型大小的指标，即模型的空间复杂度<sup>[52]</sup>。模型计算量是指浮点运算次数，计算方式为模型中的所有乘法和加法操作，计算量是衡量模型计算资源消耗的重要指标，即模型的时间复杂度<sup>[53]</sup>。在实际的运用中，模型参数量和计算量是设计和选择模型时需要考虑的重要因素。

在深度学习模型中，为了评估分类模型的性能有着多种指标。以二分类模型为例子，将样本根据其真实类别和模型的预测类别得到四个基础指标即：

**True Positive (TP):** 模型正确地将正样本预测为正类；

**False Positive (FP):** 模型错误地将负样本预测为正类；

**False Negative (FN):** 模型错误地将正样本预测为负类；

**True Negative (TN):** 模型正确地将负样本预测为负类。

由以上四个基础指标可以延伸出准确率 (Accuracy)、精确率 (Precision)、召回率 (Recall) 这三个在图像分类领域常用的评价指标。

(1) 准确率：表示模型的预测结果和真实标签匹配的比例中的最高比例。即只有当模型的预测结果中最高得分对应的类别与真实标签一致时，才被认为是预测正确，准确率计算如公式 (4.4) 所示：

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (4.4)$$

(2) 精确率：表示正确分类的正例个数占正例的实例个数的比例。精确率计算如公式 (4.5) 所示：

$$Precision = \frac{TP}{TP + FP} \quad (4.5)$$

(3) 召回率：也叫查全率，表示正确分类的正例个数占实际正例个数的比例，召回率计算如公式 (4.6) 所示：

$$Recall = \frac{TP}{TP + FN} \quad (4.6)$$

通过综合精确度和召回率两个指标，得到了一个新的综合性的评价指标，即 F1 分数 (F1 score)。F1 分数计算如公式 (4.7) 所示：

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (4.7)$$

在评价一个图像分类模型上，准确率是一个常用的评价指标。但由于战国竹简文字数据集 zgj1001 是一个不均衡分布的数据集，使用准确率作为评价指标会出现准确率虚高的情况。因为在不均衡数据集中一些类别的样本数量远远多于其他类别，这可能导致模型倾向于预测样本量较大的类别，从而使得准确率偏高，但对于少数类别的分类效果却很差。

为了避免出现这种情况，使用平衡准确率 (Balanced Accuracy, BA) 来计算每个类别的准确率的均值，BA 计算为公式 (4.8) 所示：

$$BA = \frac{\sum_{i=1}^N ACC_i}{N} \quad (4.8)$$

其中， $N$  表示样本总数， $i$  表示单个样本， $ACC_i$  表示每一个样本类别的准确率。

综上所述，F1 分数通过结合精确度和召回率，提供了一个更加全面的评估指标，平衡准确率通过计算每个类别的准确率，专用于处理不均衡数据集。而本文构建的战国竹简文字识别数据集是一个样本分布呈现长尾分布，样本数量极其不均衡的数据集。因此，为了更加准确地评价模型性能，本章实验中使用参数量、计算量、平衡准确率和 F1 分数等客观评价指标作为算法模型的评价指标。

### 4.3.3 训练参数设置

本章实验在 GeForce RTX 3090 GPU 上使用 PyTorch 工具箱训练 ResNet18、ResNet34、ResNet50 和 SSA-ResNet50 模型。使用随机梯度下降 (Stochastic Gradient Descent, SGD)<sup>[54]</sup> 算法进行优化, 其中 SGD 优化算法中的动量设置为 0.9, 权重衰减率设置为 0.005, 初始学习率设置为 0.001, 并采用余弦退火策略 (Cosine Annealing)<sup>[55]</sup> 逐渐降低学习率 (Learning Rate), 该方法通过在训练过程中控制学习率的大小帮助模型更好地收敛和泛化。损失函数采用 Cross-Entropy Loss<sup>[56]</sup>, 训练的轮次 (Epoch) 设置为 400, 批量大小 (Batch Size) 设置为 32。

模型训练学习率曲线图如图 4.3 所示, 可见在训练过程中 Learning Rate 随着训练 Epoch 的增加逐渐而下降, 并且趋于稳定。由于使用了余弦退火策略, 在模型训练中学习率曲线图呈现余弦分布。这种余弦形状的学习率曲线可以防止模型在训练中由于学习率值过大产生的梯度爆炸现象, 梯度爆炸会导致模型无法优化。并且有助于加快模型的收敛速度, 能够有效减少模型在训练过程中的震荡现象。因此, 通过采用余弦退火策略, 模型在训练过程中能够更加有效地调整学习率, 从而有利于提高模型训练效果并加快收敛速度。

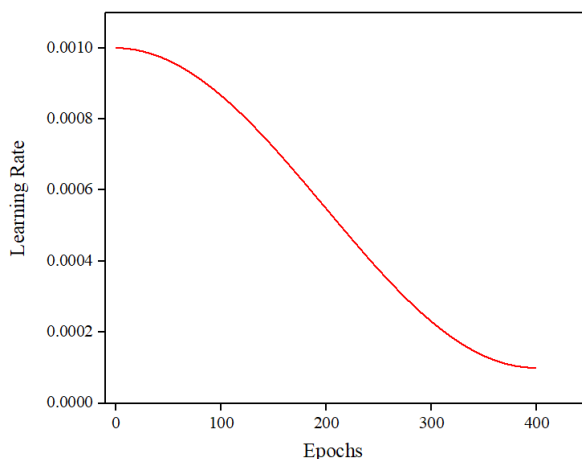
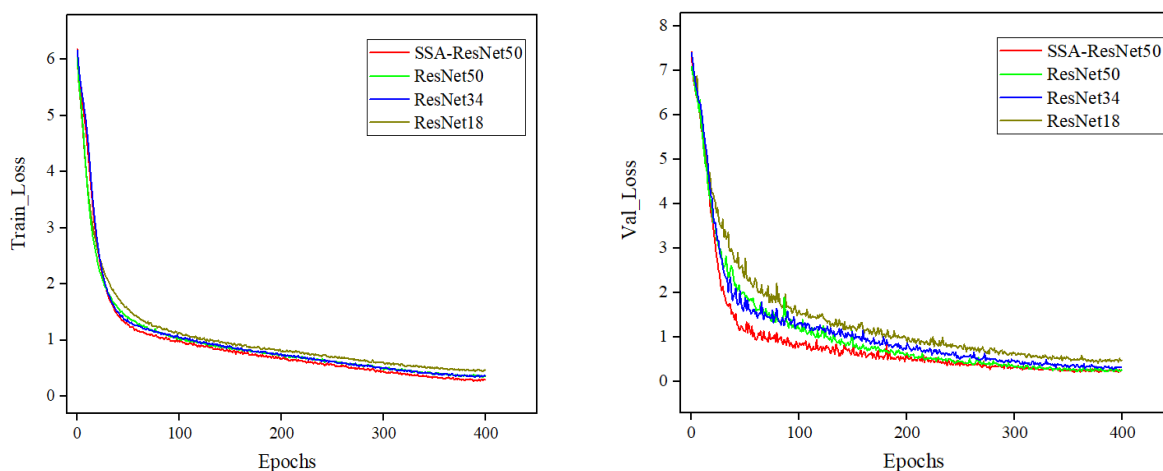


图 4.3 模型训练学习率曲线图

Fig. 4.3 Learning rate curve for model training

### 4.3.4 模型训练

在设置好训练参数设置后, 在战国竹简文字识别数据集 zgj1001 中的训练集上进行 ResNet18、ResNet34、ResNet50 和 SSA-ResNet50 模型的训练, 并且训练一个 Epoch 完成后在 zgj1001 中的验证集上进行模型的评估。



(a) 各种模型训练损失曲线图 (b) 各种模型验证损失曲线图  
 (a) Training loss curves for various models (b) Validation loss curves for various models

图 4.4 模型损失函数曲线图

Fig. 4.4 Graph of the model loss function

模型损失函数曲线图如图 4.4 所示，从图中可以看出：

(1) ResNet18、ResNet34、ResNet50 和 SSA-ResNet50 模型的训练集上的损失函数曲线和验证集上的损失函数曲线随着 Epoch 的提高逐渐下降，并且在 300 个 Epoch 左右时已经达到了收敛状态，此时随着 Epoch 的提高模型损失函数变化不大，说明此时模型已经训练到拟合状态；

(2) 从 ResNet18 模型和 ResNet34 模型的损失函数曲线可以看出随着算法网络卷积层数的加深，模型损失下降加快，说明算法层数的堆叠可以加快损失函数的收敛速度，也就是模型的拟合速度；

(3) 从 ResNet34 模型和 ResNet50 模型的损失函数曲线可以看出算法层数的堆叠已经不能提高模型下降的速度，算法层数的堆叠加快模型的拟合速度是有限的；

(4) 与其他三个模型相比，SSA-ResNet50 模型的损失函数下降速度最快。且从 ResNet50 算法和 SSA-ResNet50 算法的损失函数曲线变化可以说明本章改进的空间自注意力机制可以大幅提高模型的拟合速度。

综上所述，深度学习模型的网络层数可以影响损失函数的收敛速度，随着网络算法层数堆叠，损失函数收敛速度加快，起到加快模型拟合速度的作用。但是算法层数的堆叠的作用是有限的，ResNet34 算法与 ResNet50 算法的模型收敛速度已相差无几。而本章改进的 SSA-ResNet50 算法通过空间自注意力机制更加关注数据集中样本的关键特征信息，从而加快了模型的拟合速度。

## 4.4 实验结果及对比分析

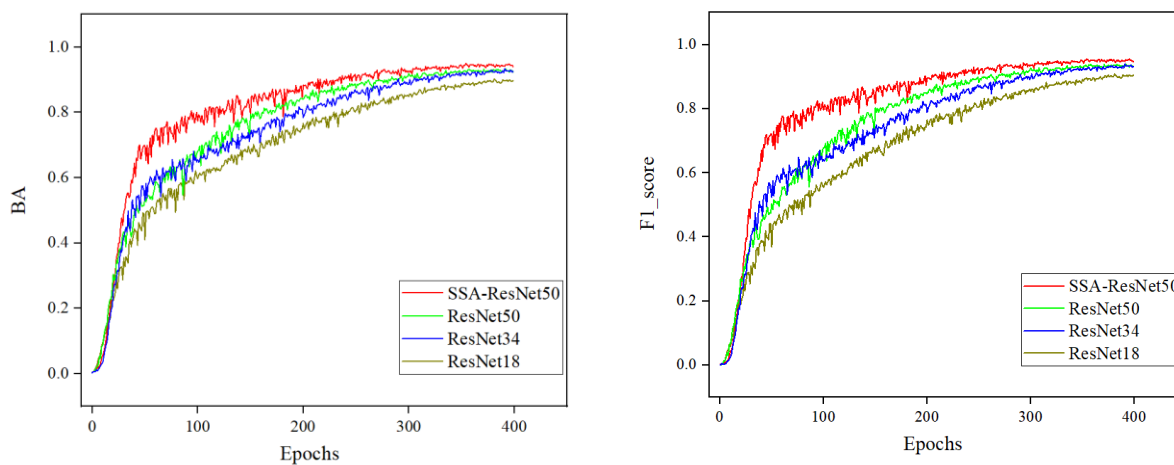
### 4.4.1 ResNet 系列模型的对比分析与预测结果

(1) 在战国竹简文字识别数据集 zgzi1001 验证集上对 ResNet18、ResNet34、ResNet50 和 SSA-ResNet50 模型的平衡准确率、F1 分数、模型参数量和模型计算量等四个客观评价指标进行对比分析, 实验结果对比如表 4.1 所示, 可视化如图 4.5 所示。

表 4.1 ResNet 模型的客观评价指标对比分析

Tab. 4.1 Comparative analysis of objective evaluation indicators of ResNet model

| 模型                  | 平衡准确率<br>(BA, %) | F1 分数<br>(F1 score, %) | 模型参数量<br>(Params, M) | 计算量<br>(FLOPs, G) |
|---------------------|------------------|------------------------|----------------------|-------------------|
| ResNet18            | 89.75            | 90.41                  | 11.69                | 0.15              |
| ResNet34            | 93.29            | 93.19                  | 21.80                | 0.30              |
| ResNet50            | 93.41            | 93.32                  | 25.56                | 0.34              |
| <b>SSA-ResNet50</b> | <b>94.97</b>     | <b>94.57</b>           | <b>34.15</b>         | <b>0.37</b>       |



(a) ResNet 模型 BA 曲线图

(b) ResNet 模型 F1 分数曲线图

(a) BA curve graphs of ResNet models

(b) F1 score curve graphs of ResNet models

图 4.5 模型识别性能曲线图

Fig. 4.5 Curve chart of model recognition performance

从表 4.1 中和图 4.5 中可以看出:

① ResNet34 模型与 ResNet18 模型相比, 模型的参数量和计算量增加了约一倍, 识别性能指标 BA 增加了 3.94%, F1 score 增加了 3.07%。说明算法模型中卷积层个数的

增加会增大参数量和计算量，而通过使用卷积层数较深的深度学习模型可以提高战国竹简文字的识别性能；

② ResNet50 模型与 ResNet34 模型相比，模型层数提高了 16 层，但是参数量与计算量相差不大，因为 ResNet50 算法采用的是 Bottleneck 结构的残差块，可以大幅减少模型的参数量与计算量。而模型识别性能 BA 只增加了 0.13%，F1 score 只增加了 0.14%，说明网络算法中卷积层数提高模型识别性能的作用是有限的，因此本文选择 ResNet50 模型作为基准模型；

③ SSA-ResNet50 模型与 ResNet50 模型相比，参数量增加了 8.59M，计算量增加了 0.03G，BA 提高了 1.67%，F1 score 提高了 1.34%。说明 SSA-ResNet50 中的空间自注意力模块是一种高效的注意力模块，在提高模型的识别性能的同时，占用的参数量和计算量不高；

④ 由图 4.5 可知，SSA-ResNet50 模型的 BA 和 F1 score 最高且拟合速度最快，在第 150Epoch 左右已经达到拟合状态，而 ResNet18、ResNet34 和 ResNet50 在第 250Epoch 左右才达到拟合状态，说明本章改进的空间自注意力机制可以加快模型的拟合速度。

综上所述，ResNet50 算法为这三种 ResNet 系列算法中性能最好的算法，且得出结论：再增加网络算法卷积层个数，模型的性能提升不大。考虑到战国竹简文字上存在的竹木纹路、墨迹、部分字体断裂等干扰信息，而这些干扰信息会对模型识别性能产生影响。因此，本章对 ResNet50 算法进行改进，通过将改进的空间自注意力模块添加到 ResNet50 网络中，提高了模型的识别性能与拟合速度，最终在战国竹简文字识别数据集中 BA 达到了 94.97%，F1 score 达到了 94.57%。

(2) 为了验证本章改进的 SSA-ResNet50 模型对战国竹简文字的关键特征信息提取能力，在战国竹简文字识别数据集 zgzt1001 的测试集上进行模型预测。选取战国竹简文字数据集中墨迹干扰字、断裂字和受到竹木纹路干扰字等类型字进行模型预测结果对比分析。

图 4.6 为 ResNet18 模型预测结果、图 4.7 为 ResNet34 模型预测结果、图 4.8 为 ResNet50 模型预测结果、图 4.9 为 SSA-ResNet50 模型预测结果。

由图 4.6 ResNet18 模型预测结果可知，模型在墨迹干扰字、断裂字和纹路干扰字的识别准确率不高。说明竹简上存在的墨迹干扰、部分字体断裂和竹木纹木干扰会导致模型特征提取能力下降，从而无法提取到足够的战国竹简文字信息，导致影响模型预测的识别准确率。基于此，在后续实验选择卷积网络结构更深的模型来提高模型的特征提取能力，从而提高模型预测的识别准确率。



预测结果: 女 准确率: 95.84%    预测结果: 皮 准确率: 95.19%    预测结果: 异 准确率: 94.83%



(a) 墨迹干扰字 (a) Ink interference character    (b) 断裂字 (b) Broken character    (c) 纹路干扰字 (c) Pattern interference character

图 4.8 ResNet50 模型预测结果

Fig. 4.8 ResNet50 model prediction results

由图 4.9 中 SSA-ResNet50 模型预测结果, 模型在 SSA-ResNet50 模型的识别准确率大于 ResNet50 模型的识别准确率, 说明了改进模型的有效性。

预测结果: 女 准确率: 97.39%    预测结果: 皮 准确率: 96.61%    预测结果: 异 准确率: 96.41%



(a) 墨迹不清字的预测结果 (a) Ink interference character    (b) 断裂字的预测结果 (b) Broken character    (c) 纹路干扰字的预测结果 (c) Pattern interference character

图 4.9 SSA-ResNet50 模型预测结果

Fig. 4.9 SSA-ResNet50 model prediction results

综上, 从四种模型的预测结果可视化可知:

① 由 ResNet18、ResNet34、ResNet50 和 SSA-ResNet50 模型在战国竹简文字识别数据集 zgzt1001 的测试集上的识别效果, 说明了四种模型没有出现过拟合现象, 在测试集上依然可以识别出战国竹简文字, 具有一定的泛化能力;

② 在 ResNet18、ResNet34 和 ResNet50 模型中的预测结果可以看出，随着算法模型卷积层数的加深，模型的预测结果和准确率提高；

③ 在 ResNet18、ResNet34、ResNet50 和 SSA-ResNet50 四种模型中的预测结果对比分析可知，战国竹简文字上存在的墨迹干扰、部分字体断裂、竹木纹路干扰等问题会影响模型的识别效果；

④ SSA-ResNet50 模型的识别效果为这四种模型中最好，说明本章改进的空间自注意力机制通过关注竹简文字上的关键特征信息，可以很好地解决战国竹简文字上存在的干扰信息导致模型识别准确率下降的问题。

由此得出，战国竹简文字上存在的墨迹干扰、部分字体断裂、竹木纹路干扰等问题会影响模型的识别效果。其中，ResNet50 模型为三种 ResNet 系列模型中预测效果最好的模型。而本章在 ResNet50 模型上进行改进，通过改进空间自注意力机制，使模型更加关注战国竹简文字的主体文字部分，从而提高了模型在这三种干扰下的预测效果。本章改进的 SSA-ResNet50 模型的预测结果最好，且模型的泛化性较好。

#### 4.4.2 改进模型的客观评价指标对比分析

为了验证本章改进的空间自注意力机制模块的有效性，本小节进行了消融实验。消融实验主要分为三个实验：第一个实验在 ResNet50 网络中添加空间注意力机制模块；第二个实验在 ResNet50 网络中添加自注意力机制模块；第三个实验为 ResNet50 网络中添加本章改进的空间自注意力机制模块。

各种注意力模块的实验结果对比如表 4.2 所示。

表 4.2 各种注意力模块实验结果对比

Tab. 4.2 Comparison of experimental results of various attention modules

| 模型                         | 平衡准确率<br>(BA, %) | F1 分数<br>(F1 score, %) |
|----------------------------|------------------|------------------------|
| ResNet50                   | 93.41            | 93.32                  |
| ResNet50+Spatial Attention | 93.81            | 93.61                  |
| ResNet50+Self-Attention    | 94.12            | 94.34                  |
| <b>ResNet50+SSA</b>        | <b>94.97</b>     | <b>94.57</b>           |

由表 4.2 可以看出：

① 与 ResNet50 相比，在 ResNet50 网络中添加空间注意力机制模块后。BA 值提高了 0.43%，F1 值提高了 0.31%，说明添加空间注意力机制可以提高模型识别精度；

② 与 ResNet50 相比，在 ResNet50 网络中添加自注意力机制模块后。BA 值提高了 0.76%，F1 值提高了 1.09%，说明添加自注意力机制也可以提高模型识别精度；

③ 与 ResNet50 相比，在 ResNet50 网络中添加本文改进后的空间自注意力机制模块后，BA 提高了 1.67%，F1 score 提高了 1.34%。由此可知，本章改进的空间自注意力机制模块为这几种注意力机制中提升的最多。

综上所述，通过进行三个对比实验，验证了本章将空间注意力机制和自注意力机制进行特征融合可以提高模型的识别精度。

#### 4.4.3 不同模型的客观评价指标对比分析

在深度学习图像识别领域，AlexNet 模型<sup>[57]</sup>、GoogLeNet 模型<sup>[58]</sup>和 VGG16 模型<sup>[59]</sup>常用于图像分类任务中，而 ConvNeXt 模型<sup>[60]</sup>为先进的卷积神经网络模型。因此，在相同实验条件下，将 SSA-ResNet50 与这四种算法进行对比分析，来验证本算法的性能，不同模型的客观评价指标对比分析如表 4.3 所示。

表 4.3 不同模型的客观评价指标对比分析

Tab. 4.3 Comparative analysis of objective evaluation indicators for different models

| 模型                        | 平衡准确率<br>(BA, %) | F1 分数<br>(F1 score, %) | 模型参数量<br>(Params, M) | 计算量<br>(FLOPs, G) |
|---------------------------|------------------|------------------------|----------------------|-------------------|
| AlexNet <sup>[57]</sup>   | 90.64            | 90.15                  | 7.46                 | 0.02              |
| GoogLeNet <sup>[58]</sup> | 91.06            | 90.97                  | 7.00                 | 0.13              |
| VGG16 <sup>[59]</sup>     | 93.38            | 93.41                  | 138.36               | 1.38              |
| ConvNeXt <sup>[60]</sup>  | 94.49            | 94.81                  | 28.57                | 0.36              |
| <b>SSA-ResNet50</b>       | <b>94.97</b>     | <b>94.57</b>           | <b>34.15</b>         | <b>0.37</b>       |

由表 4.3 可以看出：

(1) 本章改进的 SSA-ResNet50 为这四种模型中 BA 值最高的模型；

(2) 虽然 VGG16 模型与 SSA-ResNet50 模型的 BA 值和 F1 score 值相差不多，但是 VGG16 模型的参数量为 SSA-ResNet50 模型的 4 倍多，计算量高于 SSA-ResNet50 模型约 1G；

(3) 本章改进的 SSA-ResNet50 模型的 BA 值优于先进的 ConvNeXt 模型，且模型计算量和参数量相差不多。

综上所述，本文改进的 SSA-ResNet50 模型能在保证较高的识别性能下，具有较小的参数量和较低的计算量，可以在战国竹简文字识别任务中取得较好的效果。

## 4.5 本章小结

为了解决竹简文字上的竹木纹路、墨迹、部分字体断裂等特征都会对文字识别造成影响的问题，本章改进了基于空间自注意力机制的战国竹简文字识别模型，并将本章模型与其他图像识别模型进行了对比分析，总结如下：

(1) 对 ResNet18、ResNet34 和 ResNet50 模型进行了实验，通过使用卷积层数较深的深度学习模型可以提高战国竹简文字的识别性能。但是 ResNet34 和 ResNet50 模型的识别性能相差不多，说明网络算法中卷积层数提高模型识别性能的作用是有限的，因此本章选择 ResNet50 模型作为基准模型。

(2) 针对战国竹简上存在的墨迹、部分字体断裂、竹木纹路等问题会影响模型的识别效果的问题。本章改进了空间自注意力机制来关注战国竹简文字的主体文字信息，减小了竹简上干扰信息对模型的识别造成的影响，并将其应用于 ResNet50 模型上，搭建了 SSA-ResNet50 模型。

(3) 为了验证改进的空间自注意力机制模块的有效性，本章进行了消融实验。实验结果表明，本章通过将空间注意力机制和自注意力机制进行特征融合可以提高模型的识别精度。

(4) 对模型进行客观评价指标分析，本章改进的 SSA-ResNet50 模型为 ResNet 系列模型中识别性能最好的模型。与 AlexNet、VGG16 和 GoogLeNet 等经典的图像识别模型相比，SSA-ResNet50 模型在各个评价指标上表现较好，且本章改进的模型的 BA 值高于先进的 ConvNeXt 模型。从模型在墨迹干扰字、部分字体断裂字和竹木纹路字的预测结果可以看出，本章模型可以很好的识别战国竹简文字，且模型识别准确率和模型泛化性较好。

## 5 基于自监督双分支网络的战国竹简文字识别算法研究

第四章通过改进的空间自注意力机制模块，使模型更加关注战国竹简文字的主体特征，解决了战国竹简文字上竹木纹路、墨迹、部分字体断裂等干扰信息对模型识别产生影响的问题。然而，由于战国竹简文字识别数据集本身呈现长尾分布，其严重的样本数量不均衡问题会造成模型训练困难和模型识别性能差。因此，本章将 SSA-ResNet50 算法与自监督对比学习中的 SimSiam 算法结合，通过设计双分支网络来解决战国竹简文字识别数据集长尾分布造成的影响，进而提高模型的识别性能。

### 5.1 自监督对比学习

自监督学习是一种创新的学习范式，通过利用数据本身作为监督信息来学习样本数据的特征表达，而不依赖于人工标注的类别标签信息<sup>[61]</sup>。作为自监督学习的一个分支，对比学习通过使表示空间内两个增强批（视为正样本）中的特征向量距离比负样本的特征向量距离更近，从而实现了模型的特征表征能力<sup>[62]</sup>。然而，这种方式需要大量的负样本进行支撑。

2021 年，何恺明团队提出 SimSiam 算法来简化对比学习算法<sup>[63]</sup>。SimSiam 框架结构如图 5.1 所示。SimSiam 网络由两个投影器、两个预测器和两个余弦分类器组成。投影器是一个全连接的层，将向量映射到较低维表示，预测层也是一个全连接的层。

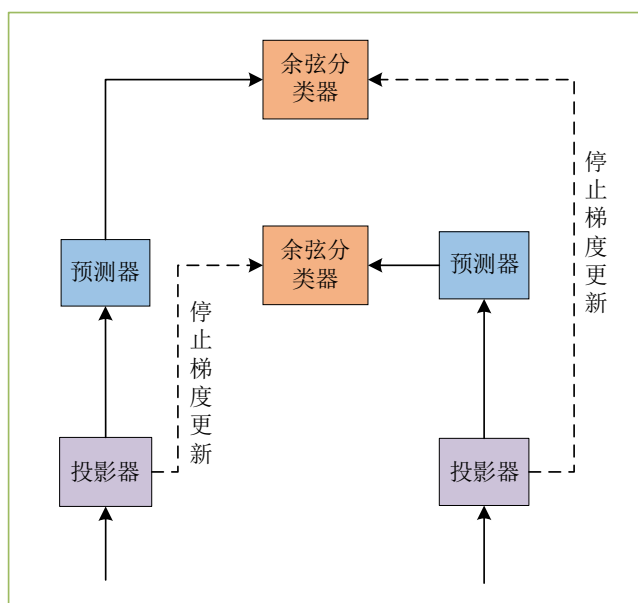


图 5.1 SimSiam 框架结构

Fig. 5.1 SimSiam framework structure

SimSiam 没有使用动量编码器，而是使用一个孪生网络 (Siamese Networks)。通过最大化一个图像的两个增强图之间的余弦相似度来进行特征表示学习，同时采用一种简单的停止梯度技术 (Stop-gradient) 来避免算法训练的崩溃问题。SimSiam 训练不需要大量的负样本、不使用动量编码器、模型参数量小，因此在对比学习中广泛使用。

## 5.2 改进网络模型主体框架

本章将监督对比学习与双分支网络结合在一起，改进了自监督双分支网络算法 (Self-supervised Bilateral Branch Network, SBBN) 来解决战国竹简文字数据集长尾分布造成的影响，SBBN 的模型框架如图 5.2 所示。SBBN 主要由卷积网络学习分支、自监督对比学习分支和损失函数计算三部分组成。

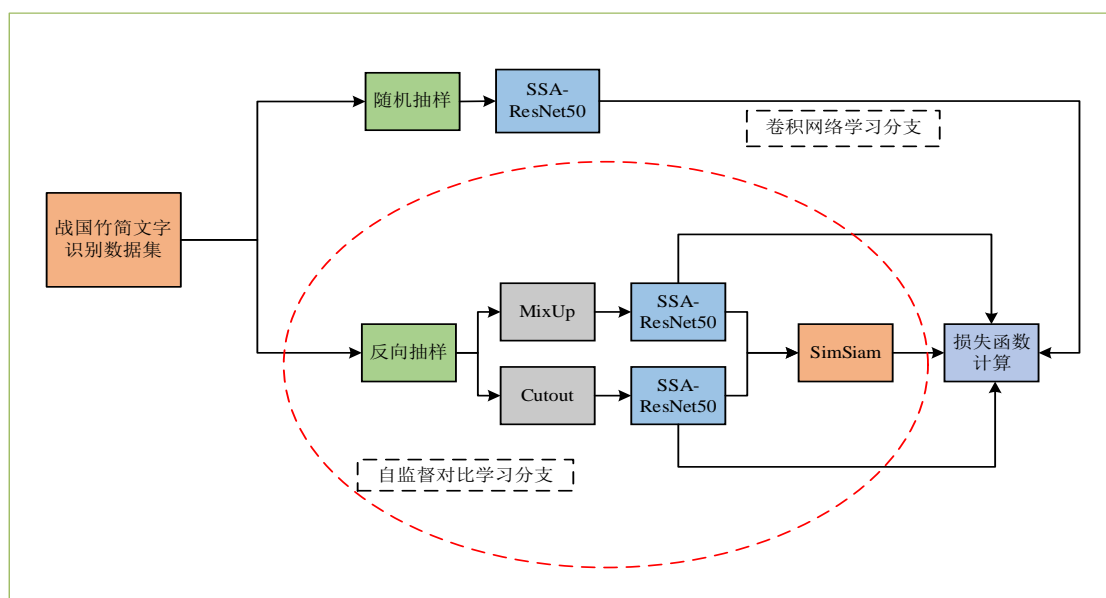


图 5.2 SBBN 框架结构

Fig. 5.2 SBBN framework structure

### 5.2.1 卷积网络学习分支

卷积网络学习分支有利于对战国竹简文字的表达学习。首先，把战国竹简文字识别数据集送入到均衡采样器中；其次，在均衡采样器中对数据进行均衡采样，模拟原始数据集分布；最后，通过 SSA-ResNet50 网络提取特征信息。卷积网络学习分支由均匀采样器和 SSA-ResNet50 网络结构构成。

(1) 均匀采样器通过均匀采样 (Uniform Sample) 策略，确保每张样本在每个 Epoch 中具有相等的概率被采样一次，这种采样方式有助于反映样本的原始数据分布信

息。如图 5.3 所示，均匀采样后数据集呈现出分布情况，其中每张样本被选中的概率是相同的。这意味着样本数量更多的类别将获得更多的被采样样本数据，符合原始数据长尾分布的特征。

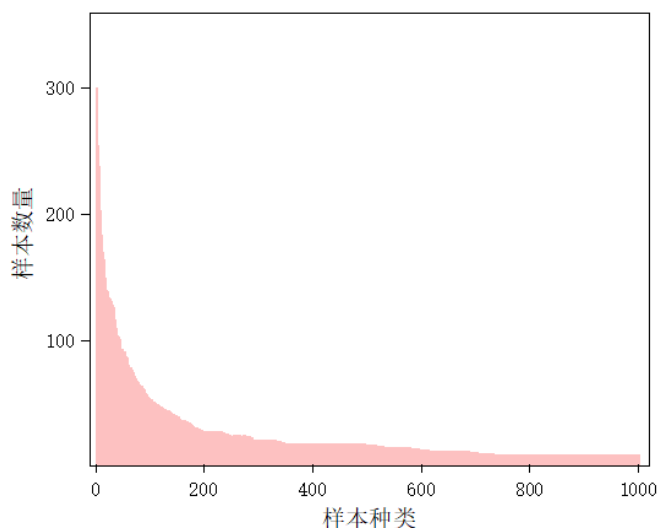


图 5.3 均衡采样

Fig. 5.3 Uniform-Sample

(2) 战国竹简文字识别数据集中的图像通过均匀采样后送入到 SSA-ResNet50 网络，输出为  $1 \times 1 \times 1001$  的张量，通过 softmax 分类器来计算准确率  $\hat{p}_i$ ，准确率计算如公式 (5.1) 所示：

$$\hat{p}_i = \frac{\exp(s_i(x))}{\sum_{j=1}^n \exp(s_j(x))} \quad (5.1)$$

其中， $n$  表示样本类别总数，在这里  $n=10001$ ， $s_i(x)$  表示属于第  $i$  类的得分， $\exp(\cdot)$  表示进行指数运算， $\sum_{j=1}^n \exp(s_j(x))$  表示对所有类别得分进行指数计算的总和。

### 5.2.2 自监督对比学习分支

自监督对比学习分支由逆采样器、Mixup 和 Cutout 图像增强操作、两个 SSA-ResNet50 网络和一个自监督对比学习 SimSiam 网络组成。

(1) 自监督对比学习分支采用逆采样 (Reversed Sample) 策略，在逆采样的过程中，采用频率与样本数量成反比，也就是样本数量较少的类别被采样的更多，从而使得模型训练时更加关注战国竹简文字识别数据集 zgzt1001 的尾部类样本。如图 5.4 所示，逆采样后的样本分布情况和长尾分布的样本分布情况相反。

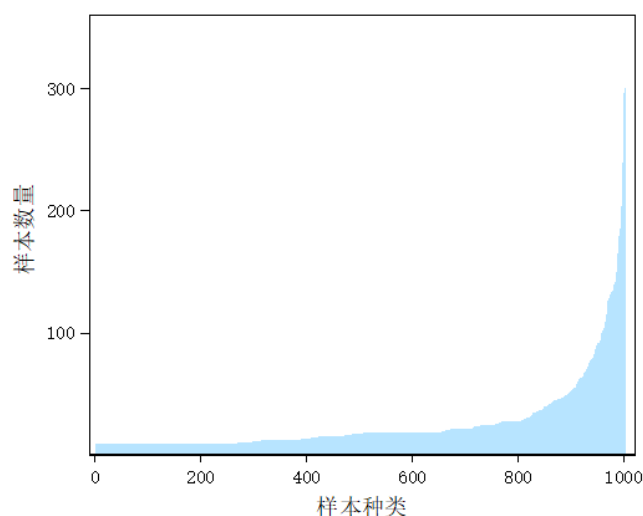


图 5.4 逆采样

Fig. 5.4 Reversed Sample

(2) 经过逆采样策略后，对提取的图像进行分别 Mixup 和 Cutout 操作。

① Mixup 是一种全局混合标签数据增强方法，通过混合两个不同类别的图像来生成混合样本<sup>[64]</sup>。图 5.5 为 Mixup 操作，图 5.5(a)为原图 1，图 5.5(b)为原图 2，图 5.5(c)为将这两种原图进行 Mixup 操作后生成的混合图像。



(a) 原图 1  
(a) Original image 1

(b) 原图 2  
(b) Original image 2

(c) 混合图像  
(c) Mixed image

图 5.5 Mixup 操作

Fig. 5.5 Mixup operation

混合图像可以缓解战国竹简文字识别数据集中样本不足的问题，并且通过引进了一定程度上的噪声，在模拟真实样本图像的同时，能够减少模型对训练数据的过拟合，提高模型的泛化能力。Mixup 计算如公式 (5.2) 和公式 (5.3) 所示：

$$\begin{aligned} \lambda &\sim \text{Beta}(\beta, \beta), \\ \tilde{x}_m &= \lambda x_i + (1 - \lambda)x_j \end{aligned} \quad (5.2)$$

$$\tilde{y}_m = \lambda y_i + (1 - \lambda)y_j \quad (5.3)$$

其中,  $\lambda$  是从一个超参数  $\beta$  的 Beta 分布中随机采样得到的一个参数, 用来控制两个样本进行混合的程度。 $(x_i, y_i)$ 和 $(x_j, y_j)$ 为两幅图像的特征信息和其对应的标签,  $\tilde{x}_m$ 和 $\tilde{y}_m$ 为经过 Mixup 操作后的图像特征信息和对应的标签。

② Cutout 操作通过在图像中随机抹除一部分区域, 生成新的训练样本<sup>[65]</sup>。具体操作是在输入图像上随机选择一个矩形区域, 并将该区域内的像素值设置为 0 或者设置为某个固定的常数。Cutout 操作可以防止模型训练时出现过拟合问题。

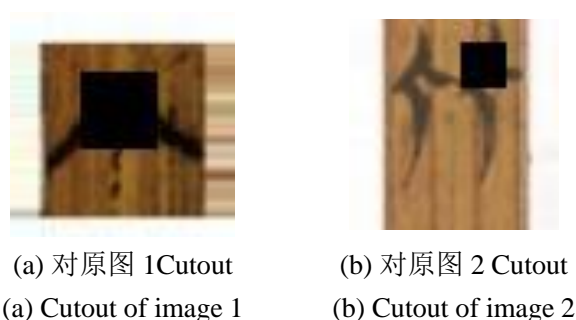


图 5.6 Cutout 操作

Fig. 5.6 Cutout operation

图 5.6 为 Cutout 操作, 其中图 5.6(a)为对原图 1 进行 Cutout 操作, 图 5.6(b)为对原图 2 进行 Cutout 操作。Cutout 操作通过随机遮挡了一部分战国竹简文字, 能够减少模型出现过拟合。并且 Cutout 操作模拟了战国竹简文字出现的部分字迹残缺现象, 从而提高模型的识别性能。

(3) 经过 Mixup 和 Cutout 操作后, 将增强后的两张图像分别送入 ACmix-ResNet50 网络中。两个 ACmix-ResNet50 网络是共享权重的。在 ACmix-ResNet50 中提取到特征 $x_1$ 和 $x_2$ 后, 通过两个 softmax 分类器分别计算 $\hat{p}_{mix}$ 和 $\hat{p}_{cut}$ 。

(4) 通过 SimSiam 网络计算两种增强后的图像的余弦相似性。首先, 将提取到的特征送入到 SimSiam 中的投影器处理, 输出为 $h_g$ 和 $h_l$ 。然后, 分别经过两个预测器将这两种表示转换输出为 $u_g$ 和 $u_l$ 。最后, 在余弦分类器计算 $h_g$ 和 $u_l$ 、 $h_l$ 和 $u_g$ 的余弦相似性, 来获得对比一致性。余弦相似性计算为公式 (5.4) 和公式 (5.5) 所示:

$$\text{sim}(u_g, \text{sg}(h_l)) = -\frac{u_g}{\|u_g\|} \cdot \frac{h_l}{\|h_l\|} \quad (5.4)$$

$$\text{sim}(u_l, \text{sg}(h_g)) = -\frac{u_l}{\|u_l\|} \cdot \frac{h_g}{\|h_g\|} \quad (5.5)$$

其中,  $\|\cdot\|$ 为 $l_2$ 标准化,  $sg(\cdot)$ 表示使用了停止更新梯度, 防止训练崩溃。通过使用负向量和归一化后的向量进行计算可以确保相似度的范围在-1 到 1 之间, 结果为 1 表示完全相似, 而为-1 表示完全不相似。

### 5.2.3 损失函数计算

CNN 通过反向传播算法将损失函数的梯度信息传递到训练网络中, 从而更新网络参数以最小化损失函数<sup>[66]</sup>。本章改进的算法 SBBN 的损失函数主要由卷积网络学习分支中的交叉熵损失函数 $l_{ce}$ 、自监督对比学习分支中经过 Mixup 和 Cutout 增强操作后图像的交叉熵损失函数 $l_{mix}$ 和 $l_{cut}$ 以及 SimSiam 网络中的余弦损失函数 $l_{sim}$ 组成。

(1) 卷积网络学习分支中的交叉熵损失函数 $l_{ce}$ 计算为公式 (5.6) 所示:

$$l_{ce} = - \sum_{i=1}^n y_i \log \hat{p}_i \quad (5.6)$$

其中,  $y_i$ 为第 $i$ 类战国竹简文字的标签信息,  $\hat{p}_i$ 为在卷积网络学习分支中计算出来的单类战国竹简文字的准确率。

(2) 在自监督对比学习分支中, 对于一个战国竹简文字图像输入数据 $(x_i, y_i)$ , 经过 Mixup 和 Cutout 增强操作后得到输出 $(x_g, y_g)$ 和 $(x_l, y_l)$ , 将混合图像分别输入到两个 ACmix-ResNet50 中, 获取两种混合图像的准确率 $\hat{p}_{mix}$ 和 $\hat{p}_{cut}$ , 然后混合图像交叉熵损失函数 $l_{mix}$ 和 $l_{cut}$ , 如公式 (5.7) 和公式 (5.8) 所示。

$$l_{mix} = - \sum_{i=1}^n y_g \log(\hat{p}_{mix}) \quad (5.7)$$

$$l_{cut} = - \sum_{i=1}^n y_g \log(\hat{p}_{cut}) \quad (5.8)$$

(3) 将两种混合图像输入到 SimSiam 网络中, 输出为 $h_g$ 和 $u_l$ 、 $h_l$ 和 $u_g$ , 并计算余弦相似性 $sim(u_g, sg(h_l))$ 和 $sim(u_l, sg(h_g))$ , 将两种余弦相似性的相加作为余弦损失函数 $l_{sim}$ , 如公式 (5.9) 所示。

$$l_{sim} = sim(u_g, sg(h_l)) + sim(u_l, sg(h_g)) \quad (5.9)$$

最后, 使用一个超参数 $\alpha$ 来控制卷积学习分支损失和自监督对比学习分支中的两个增强图像批的损失函数, 并将余弦损失函数作为辅助函数加到总损失里面。计算 SBBN 总损失为公式 (5.10) 所示:

$$l = \alpha l_{ce} + (1 - \alpha)(l_{mix} + l_{cut}) + l_{sim} \quad (5.10)$$

其中,  $\alpha$  的定义为:

$$\alpha = 1 - \left( \frac{T}{T_{max}} \right)^2 \quad (5.11)$$

在公式 (5.11) 中,  $T$  为训练轮次数,  $T_{max}$  为最大的训练轮数。 $\alpha$  的图像如图 5.7 所示, 随着模型训练的过程,  $\alpha$  的值为逐渐减小, 而  $(1 - \alpha)$  的值会逐渐增大。所以模型训练的时候, 一开始是训练卷积网络学习分支, 然后训练的重点逐渐转向对尾部类样本的自监督对比学习分支学习。

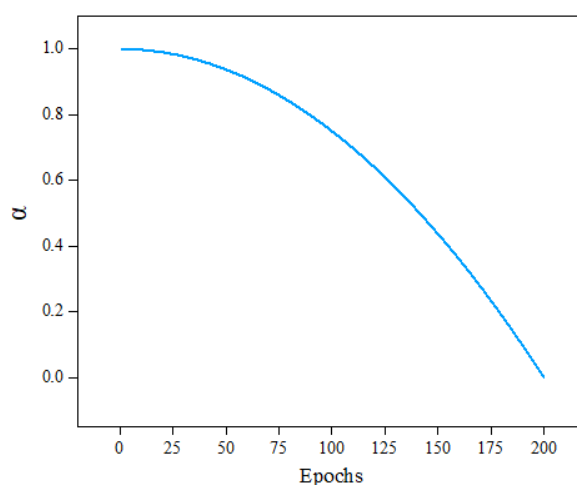


图 5.7  $\alpha$  图像

Fig. 5.7 The image of  $\alpha$

## 5.3 模型训练及参数设置

### 5.3.1 评价指标

在数据集呈现长尾分布时, 其严重的样本数量不均衡会导致图像分类出现准确率虚高的情况, 因此评价指标使用采用第 4.3.2 节中提出的平衡准确率和 F1 分数来综合性评价本章模型的识别性能。并且选用第 3.1.2 节中的 Many-shot 类 (类样本数 > 100), Medium-shot 类 (类样本数 20~100) 和 Few-shot 类 (类样本数 < 20) 作为战国文字识别数据集的子数据集来全面评价本章模型的识别性能。

### 5.3.2 训练参数设置

在 GeForce RTX 3090 GPU 上使用 PyTorch 工具箱训练模型, 使用随机梯度下降 (Stochastic Gradient Descent, SGD) 算法进行优化, 其中 SGD 优化算法中的动量设置为

0.9, 权重衰减率设置为 0.0002, 初始学习率设置为 0.01, 并采用余弦退火策略逐渐降低学习率, 训练的轮次 (Epoch) 设置为 200, 批量大小设置为 32。

## 5.4 实验结果及对比分析

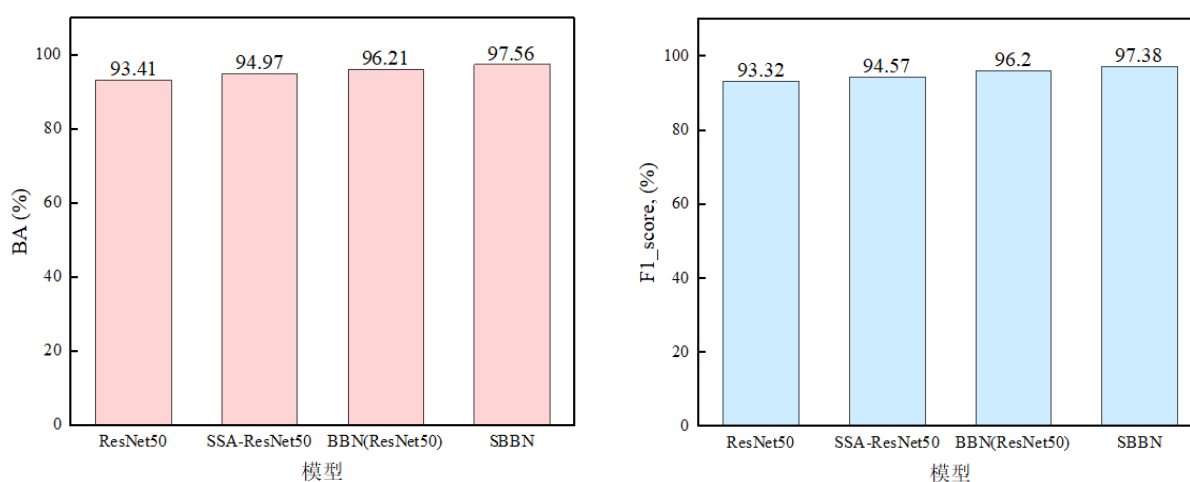
### 5.4.1 客观评价指标对比分析

(1) 本章的客观对比分析在战国竹简文字识别数据集 zgzt1001 对 ResNet50 模型、第四章改进的 SSA-ResNet50 模型、BBN 模型以及本章模型 SBBN 的平衡准确率和 F1 分数进行客观评价指标进行对比分析, 其中 BBN 模型骨干网络 (Backbone) 选择 ResNet50, 对比结果如表 5.1 所示, 可视化如图 5.8 所示。

表 5.1 zgzt1001 数据集上客观评价指标对比分析

Tab. 5.1 Comparative analysis of objective evaluation indicators on the zgzt1001 dataset

| 模型                             | 平衡准确率<br>(BA, %) | F1 分数<br>(F1 score, %) |
|--------------------------------|------------------|------------------------|
| ResNet50 <sup>[24]</sup>       | 93.41            | 93.32                  |
| SSA-ResNet50 (第四章模型)           | 94.97            | 94.57                  |
| BBN (ResNet50) <sup>[48]</sup> | 96.21            | 96.20                  |
| <b>SBBN (本章模型)</b>             | <b>97.56</b>     | <b>97.38</b>           |



(a) 各种模型 BA 值

(a) BA values for various models

(b) 各种模型 F1 分数

(b) F1 scores for various models

图 5.8 各种模型的 BA 值和 F1 分数

Fig. 5.8 BA values and F1 scores for various models

从表 5.1 和图 5.8 中可以看出：

① BBN 模型的 BA 值和 F1 分数都比基础的图像识别模型 ResNet50 和本文第四章所改进的 SSA-ResNet50 模型高，说明 BBN 模型中提出的双分支网络结构能解决战国竹简文字识别数据集中样本种类呈现长尾分布的问题；

② 与 ResNet50 模型相比，本章改进的 SBBN 模型 BA 值提高了 4.44%，F1 分数提高了 4.35%。与本文第四章改进的 SSA-ResNet50 模型相比，本章模型的 BA 提高了 2.72%，F1 分数提高了 2.97%。说明本章改进的基于自监督双分支网络的战国竹简文字识别算法可以有效地解决战国竹简文字识别数据集上存在的长尾识别问题；

③ 与 BBN 模型相比，本章改进的 SBBN 模型 BA 值提高了 1.40%，F1 值提高了 1.23%。说明本章模型在双分支网络上的改进是有效的，在双分支网络的另一条分支上结合自监督对比学习中的 SimSiam 网络可以有效提高战国竹简文字识别模型的识别性能。

(2) 为了更加全面的验证 SBBN 模型在战国竹简文字识别数据集上的识别性能上的提升效果。在战国竹简文字识别数据集的数据子集：Many-shot 类、Medium-shot 类和 Few-shot 类上，对 ResNet50 模型、BBN 模型和本文模型 SBBN 的平衡准确率进行实验结果对比分析，如表 5.2 所示，可视化如图 5.9 所示。

表 5.2 三种数据子集上平衡准确率对比分析

Tab. 5.2 Comparative analysis of balancing accuracy on three subsets of data

| 模型                             | Many-shot BA (%) | Medium-shot BA (%) | Few-shot BA (%) |
|--------------------------------|------------------|--------------------|-----------------|
| ResNet50 <sup>[24]</sup>       | 97.13            | 95.02              | 89.62           |
| SSA-ResNet50 (第四章模型)           | 97.62            | 96.42              | 91.78           |
| BBN (ResNet50) <sup>[48]</sup> | 97.41            | 96.50              | 94.71           |
| <b>SBBN (本章模型)</b>             | <b>98.42</b>     | <b>97.45</b>       | <b>96.90</b>    |

从表 5.2 和图 5.9 中可以看出：

① 在这四种模型中，ResNet50 的 Many-shot 类的 BA 值和其他三种模型的 BA 值相差不大，但是 Medium-shot 类和 Few-shot 类的 BA 值比其他模型差。说明 Medium-shot 类和 Few-shot 类中的样本数量不足会影响模型识别准确率；

② 与 ResNet50 模型相比，本文第四章改进的 SSA-ResNet50 模型通过结合空间自注意力机制关注战国竹简文字的关键信息，从而使 Medium-shot 类和 Few-shot 类分别提升了 1.47% 和 2.41%，但是 Few-shot 类上的 BA 值有待提高；

③ BBA 通过使用双分支网络，更加关注数据集的尾部类样本。在 Medium-shot 类和 Few-shot 类上的 BA 值都有所提高。特别是 Few-shot 类上的 BA 值，比 ResNet50 模型提高了 5.68%，比 SSA-ResNet50 模型提高了 3.19%；

④ 本章改进的 SBBN 模型在 Many-shot 类、Medium-shot 类和 Few-shot 类上的 BA 值是这几个模型中最高的。其中，相比于基础 ResNet50 模型：Many-shot 类 BA 值提高了 1.33%；Medium-shot 类 BA 值提高了 2.56%；Few-shot 类 BA 值提高了 8.12%。相比于 SSA-ResNet50 模型：Medium-shot 类 BA 值提高了 0.82%；Medium-shot 类 BA 值提高了 1.07%；Few-shot 类 BA 值提高了 5.58%。说明 SBBN 模型主要通过提升尾部类样本的平衡准确率来提升整体的识别性能；

⑤ 与双分支网络 BBN 相比，SBBN 模型在 Many-shot 类上 BA 值提高了 1.04%，在 Medium-shot 类上提高了 0.98%，在 Few-shot 类 BA 值提高了 2.31%。说明 SBBN 模型通过自监督对比学习分支提高了尾部类样本的识别性能。

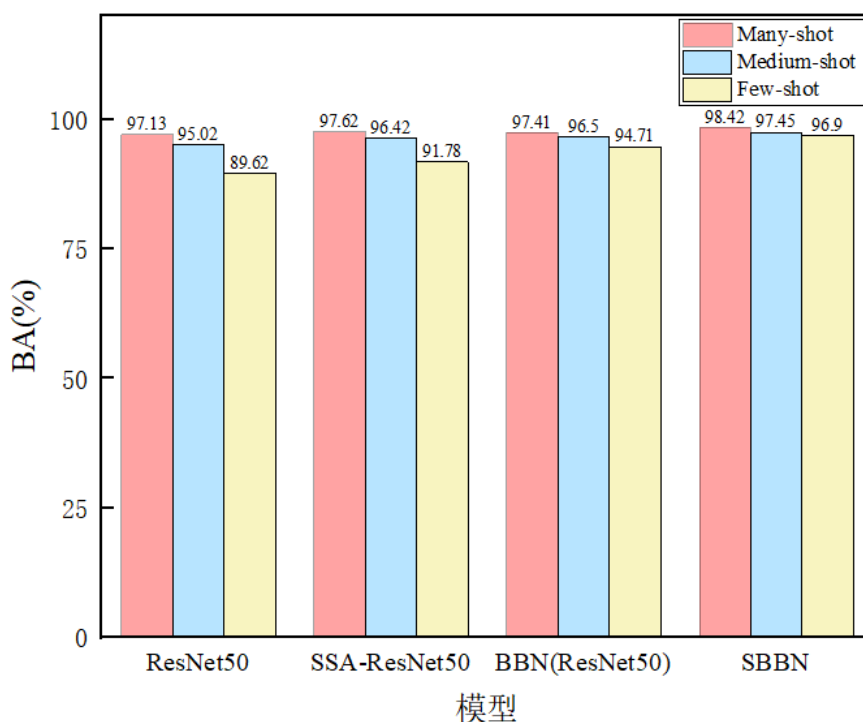


图 5.9 数据子集上平衡准确率

Fig. 5.9 Balance accuracy on a subset of data

综上，本章改进的 SBBN 模型为这四种模型中识别性能最好的模型。在战国竹简文字识别数据集 zgj1001 上达到了 97.56% 的平衡准确率和 97.38% 的 F1 分数；在战国

竹简文字识别数据集中的 Many-shot 类、Medium-shot 类和 Few-shot 类数据子集上分别达到了 98.42%、97.45%和 96.90%的平衡准确率。因此，本章改进的模型能够有效解决数据集中由于长尾分布造成识别性能不高的问题。

#### 5.4.2 模型预测结果对比分析

为了评估 SBBA 模型在战国竹简文字识别数据集上的识别效果，以及模型的泛化性。在 Many-shot 类、Medium-shot 类和 Few-shot 类的测试集上进行模型预测，预测结果如图 5.10、图 5.11 和图 5.12 所示。

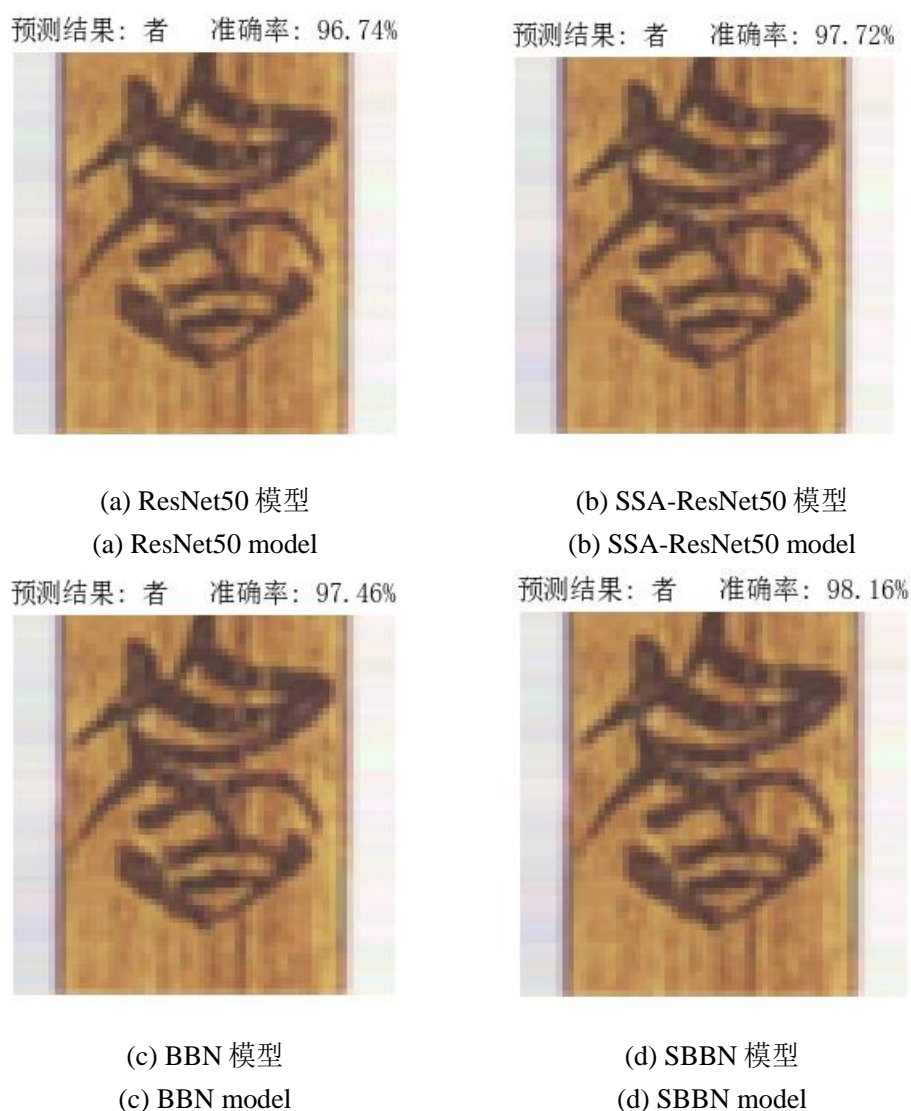


图 5.10 各种模型在 Many-shot 类识别效果

Fig. 5.10 The recognition performance of various models in Many-shot class

由图 5.10 中的各种模型在 Many-shot 类识别效果,在这四种模型在类样本数量大于 100 的时候可以达到很好的识别效果。可以得出这四种模型在数据集中的样本数量充足的情况下可以达到很高的识别准确率。

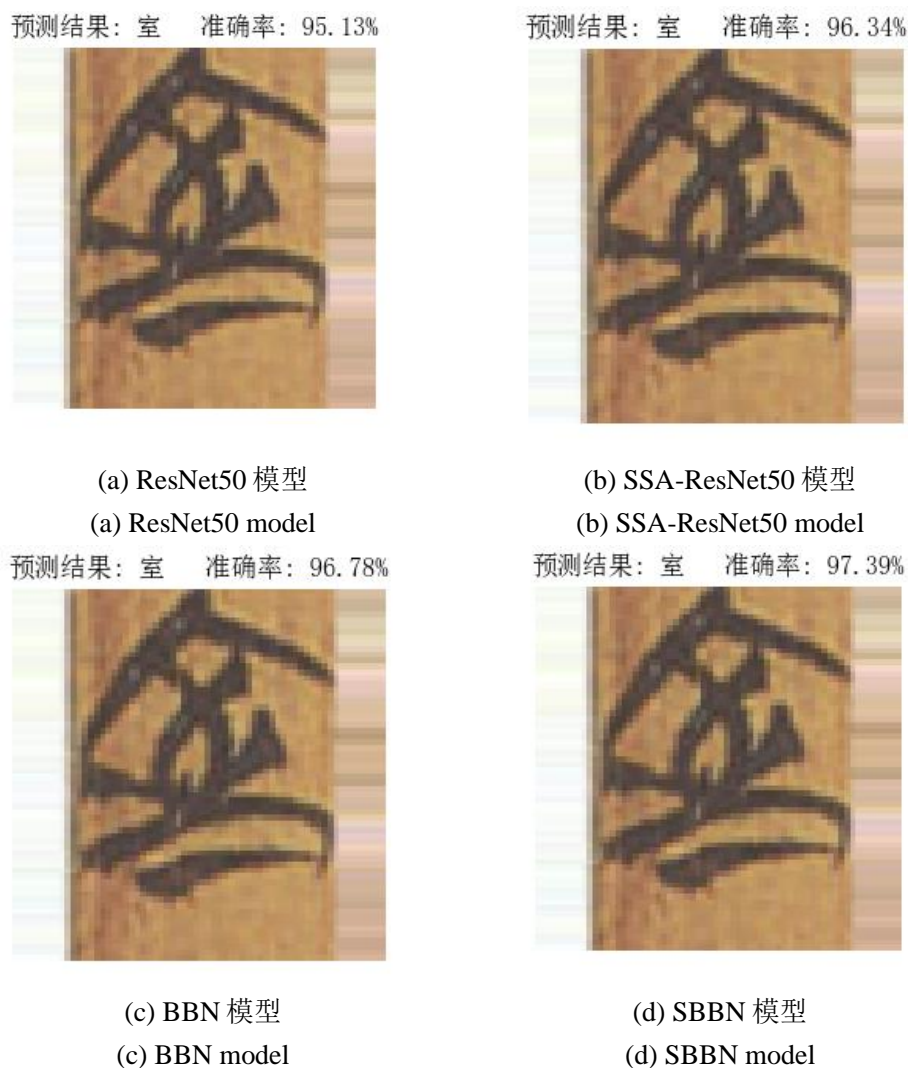


图 5.11 各种模型在 Medium-shot 类识别效果

Fig. 5.11 The recognition performance of various models in Medium-shot class

由图 5.11 中的各种模型在 Medium-shot 类识别效果,本章改进的 SBBN 模型中的识别效果为这四种模型中识别效果最好的模型。而 ResNet50 模型、SSA-ResNet50 模型和 BBN 模型在类样本数量较充足时(类样本数量大于 20 小于 100)的时候,识别准确率较 Many-shot 类上有所降低。可以得出,本章改进的模型在样本数量较充足的情况下依然可以保持很好的识别效果。

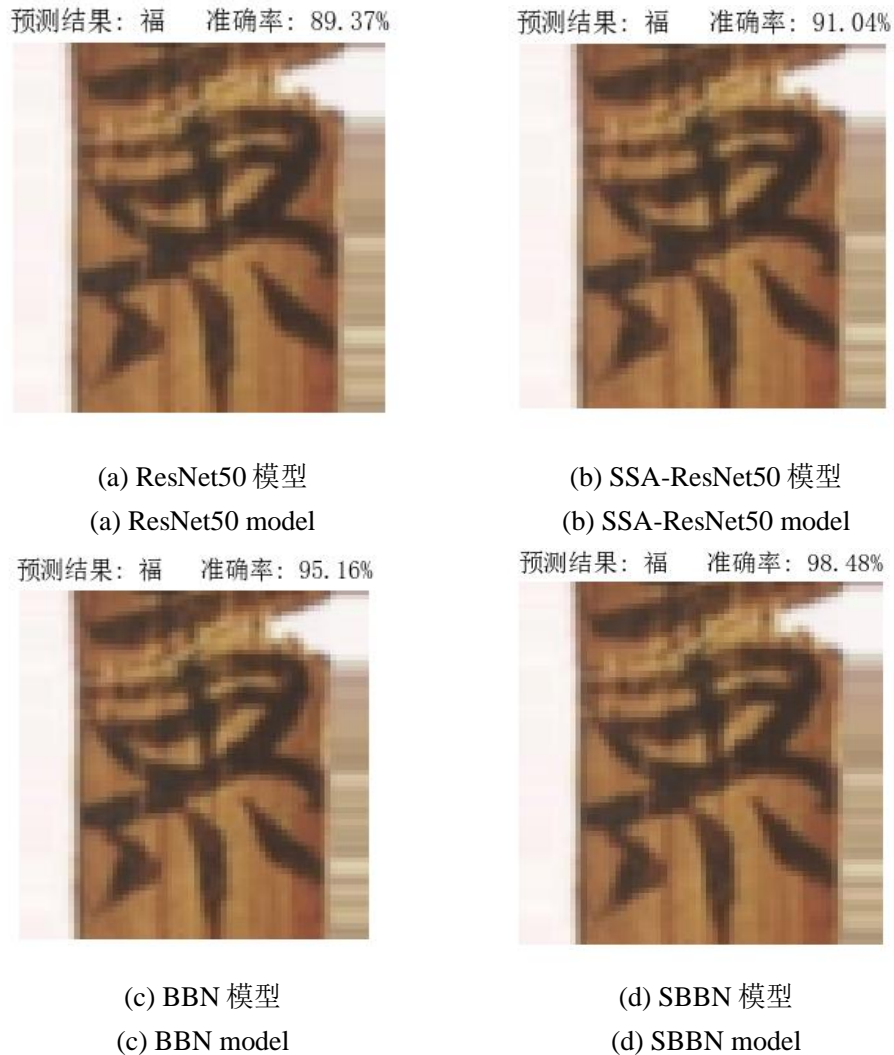


图 5.12 各种模型在 Few-shot 类识别效果

Fig. 5.12 The recognition performance of various models in Few-shot class

由图 5.12 中的各种模型在 Few-shot 类识别效果, 可知:

- (1) ResNet50 模型和 SSA-ResNet50 模型在样本不充足的情况, 识别准确率较低。
- (2) BBN 模型在样本不充足的情况下可以达到良好的识别率, 说明了 BBN 模型提出使用的双分支网络可以提高尾部类样本的准确率。
- (3) 本章改进的 SBBN 模型为 Few-shot 类中识别准确率最高, 说明 SBBN 模型在双分支网络基础上改进的自监督对比学习方法可以更加关注尾部类样本, 从而提高尾部类样本的准确率, 验证了自监督双分支网络对尾部类样本的增益效果。

综上, 本章改进的模型在样本数量不充足的情况下依然可以保持很好的识别效果, 改进的 SBBN 模型中的识别效果是这四种模型中识别效果最好的模型。

## 5.5 本章小结

由于战国竹简文字识别数据集本身呈现长尾分布，其严重的样本数量不均衡问题会造成模型训练困难和模型识别性能差。因此，本章改进基于自监督双分支网络的战国竹简文字识别算法来解决数据集长尾分布造成的影响，进而提高模型的识别性能。主要工作总结为如下：

(1) 本章将 SSA-ResNet50 算法与自监督对比学习中的 SimSiam 算法结合，设计了自监督双分支网络，双分支网络分为卷积网络学习分支和自监督对比学习分支。卷积网络学习分支网络由均匀采样器和本文第四章改进的 SSA-ResNet50 网络构成，均匀采样器采样的数据符合原始数据中长尾分布的特征，有利于模型的表示学习。自监督对比学习分支使用逆采样器对尾部类样本进行重点采样，使模型更加关注尾部类样本。对逆采样的两批图像进行 Mixup 和 Cutout 操作，并在 SimSiam 网络中计算两种增强批图像的余弦相似性，并将余弦损失函数作为辅助函数加到总损失里面。

(2) 在战国竹简文字识别数据集 zgj1001 对 ResNet50 模型、第四章改进的 SSA-ResNet50 模型、BBN 模型以及本章模型 SBBN 的平衡准确率和 F1 分数进行客观评价指标进行对比分析，本章改进的 SBBN 模型平衡准确率为 97.56%、F1 分数为 97.38%，为这四种模型中识别性能最好的模型。

(3) 为了验证 SBBN 模型能够有效解决数据集中由于长尾分布造成识别性能不高的问题，在战国竹简文字识别数据集中的 Many-shot 类、Medium-shot 类和 Few-shot 类数据子集进行对比分析实验。结果显示 SBBN 模型为四种模型平衡准确率最高，在 Many-shot 类、Medium-shot 类和 Few-shot 类分别达到了 98.42%、97.45% 和 96.90% 的平衡准确率。与 SSA-ResNet50 模型相比，Many-shot 类和 Medium-shot 类的平衡准确率分别提高了 0.82%、1.07%，在 Few-shot 类上平衡准确率重点提高了 5.58%，说明了本章改进模型对尾部类样本的提升十分显著。

(4) 为了评估 SBBA 模型在战国竹简文字识别数据集上的识别效果，以及模型的泛化性。在 Many-shot 类、Medium-shot 类和 Few-shot 类的测试集上进行模型预测，结果显示本章改进的模型在样本数量不充足的情况下依然可以保持很好的识别效果。

## 6 战国竹简文字识别系统的设计与应用

由于战国竹简文字与现代汉字存在巨大差异，其种类复杂且不易辨别，目前主要依靠古文字研究的专业人员对战国竹简文字进行破译。本文通过前几章的研究工作完成了战国竹简文字识别模型的构建，并且已经取得了良好的识别效果。为了提高战国竹简文字识别模型的便利性和实用性，促进战国竹简文字的研究和破译工作。本章利用 HTML、CSS 和 JavaScript 实现前端界面，基于 Flask 轻量级 Web 框架实现后端服务，实现了对战国竹简文字识别系统的设计与应用。

### 6.1 系统架构设置

#### 6.1.1 Flask 框架

Flask 是一个基于 Python 语言的轻量级 Web 框架，具有良好的可扩展性和高灵活性，可以帮助开发者快速构建 Web 应用程序，Flask 主要由 Werkzeug 和 Jinja2 两大核心服务构成<sup>[67]</sup>。Werkzeug 负责提供业务处理功能，包括路由请求集成、会话管理和交互式响应等服务。在 Werkzeug 中封装了 WSGI (Web Server Gateway Interface) 协议，定义了 Web 服务器和 Python Web 应用程序之间的接口规范。Jinja2 是一个用 Python 编写的模版引擎，能够根据模版文件生成动态的 HTML 页面，并提供了一些安全功能，可以防止黑客的入侵。

Flask 的工作原理是：首先，客户端发送请求信息给 Web 服务器；其次，Web 服务器接收到请求信息后，调用 WSGI 应用程序接口；最后，Flask 框架选择相应的视图函数对请求的 URL 信息进行处理，返回函数值并响应到客户端。Flask 的原理图如图 6.1 所示。



图 6.1 Flask 的原理图

Fig. 6.1 Schematic diagram of Flask

Flask 框架的实现伪代码如表 6.1 所示。首先导入所需的模块和库，然后根据环境变量中的 DEBUG 值确定是否处于调试模式；其次，根据调试模式选择相应的配置模式，并创建 Flask 应用实例。再次，通过使用 Flask-Migrate 对数据库进行迁移，在调

试模式下，会输出调试信息，包括 **DEBUG** 值、页面压缩状态和数据库连接信息；最后，执行该代码，则可以实现战国竹简文字识别系统的启动。

表 6.1 Flask 框架的实现伪代码

Tab. 6.1 Pseudocode for the implementation of the Flask framework

---

Flask 框架的应用伪代码

---

```
import os
from flask_migrate import Migrate
from flask_minify import Minify
DEBUG = (os.getenv('DEBUG', 'False') == 'True') # 设置调试模式
get_config_mode = 'Debug' if DEBUG else 'Production' # 获取配置模式
try:
    app_config = config_dict[get_config_mode.capitalize()]
    app = create_app(app_config) # 创建应用
    Migrate(app, db) # 初始化数据库迁移
    if not DEBUG:
        Minify(app=app, html=True, js=False, cssless=False)
    # 如果是调试模式，则记录调试信息
    if DEBUG:
        app.logger.info('DEBUG      = ' + str(DEBUG))
        app.logger.info('Page Compression = ' + 'FALSE' if DEBUG else 'TRUE')
        app.logger.info('DBMS      = ' + app_config.SQLALCHEMY_DATABASE_URI)
        app.logger.info('ASSETS_ROOT = ' + app_config.ASSETS_ROOT)
    if __name__ == "__main__":
        app.run(debug=True) # 运行应用
```

---

本章选用 Flask 框架来进行系统搭建，原因如下：

(1) 本文战国竹简文字识别模型是基于 python 语言代码构建的，而 Flask 框架底层代码也是 Python，相同编程代码对于整个模型的搭建和维护更加便捷；

(2) 与 Python 的其他 Web 框架如 Django 相比，Flask 框架的更加灵活，并且含有丰富的插件库利于开发人员进行个性化的网站搭建，可扩展性更强；

(3) Flask 框架是一个轻量级的 Web 框架，其核心代码相比其他框架很少。使用 Flask 框架开发本文战国竹简文字识别系统，可以提高战国竹简文字识别系统的打开和运行速度。

综上所述，本文战国竹简文字识别系统的搭建基于 Flask 框架进行设计，通过 Flask 框架可以实现后端服务与前端界面的交互。

### 6.1.2 MySQL 数据库

MySQL 作为一款开源的数据库管理系统，被广泛的应用于各种 Web 应用程序中，MySQL 数据库主要由数据库、表、行、列、索引和视图等组成<sup>[68]</sup>。其中，数据库是数据的容器，用于存储一组相关的表，是整个数据库系统的最顶层结构。表是数据库中的基本单位，用于存储具有相同结构的数据记录。行是表中的记录，每一行代表一个数据条目，包含了各列对应的具体数值或内容。列定义了表中每条记录的属性，包括字段名称、数据类型、约束条件等信息。索引是用于加快数据检索速度的数据结构，可以提高查询效率，常用于加速数据查询操作。视图是一个虚拟表，提供更直观的数据展示方式。

MySQL 支持使用结构化查询语言 (Structured Query Language, SQL) 进行数据库操作。SQL 是一种标准化的数据库查询语言，可用于执行诸如创建数据库、建立表格、插入数据、更新数据、删除数据以及查询数据等多种操作。通过使用 SQL 语句，用户可以方便地管理和操作数据库中的数据，实现数据的增、删、改、查等功能。MySQL 提供了强大而灵活的 SQL 语法支持，使得用户能够轻松地进行各种数据库操作，满足不同应用场景的需求。MySQL 与其他数据库管理系统相比有很多优势：

(1) 性能强大。MySQL 通过采用多种数据优化技术，如索引技术、分区技术和缓存技术，提高了数据处理效率；

(2) 易于管理。MySQL 通过浏览器可以直接进入数据库管理界面，其图形化的管理方式极大地提高了开发人员对数据库的设计和管理工作；

(3) 高扩展性。MySQL 支持多种数据库存储引擎和多种编程语言的接口，能够在根据业务需求选择合适的存储引擎，并支持多种操作系统中使用。

因此，本章选择 MySQL 作为战国竹简文字识别系统的数据库。

## 6.2 系统设计流程

本章的主要研究目的是搭建一个战国竹简文字识别系统，能够为用户提供便捷和准确的战国竹简文字识别服务，该系统主要包括用户的登录和注册、战国竹简文字图片上传、图像预处理操作、战国竹简文字识别和结果展示等功能。战国竹简文字识别系统的设计主要包括系统结构设计、系统功能设计和数据库设计。

### 6.2.1 系统结构设计

战国竹简文字识别系统的结构设计主要包括前端页面设计、后端服务实现。前端页面设计采用 HTML<sup>[69]</sup>、CSS<sup>[70]</sup>和 JavaScript<sup>[71]</sup>等 Web 开发语言设计一个前端页面，其中使用 HTML 创建了页面的结构和内容，使用 CSS 实现了页面的各种个性化样式，

使用 JavaScript 实现了页面的动态交互功能。后端服务基于 Flask 框架实现，用户在客户端通过 API 接口调用 Web 应用程序，包括图像预处理和加载识别模型。战国竹简文字识别系统的结构设计分为前端界面设计和后端服务调用，如图 6.2 所示。

输入一张战国竹简文字进入系统，首先前端服务会将图像数据发送到后端服务；其次，通过后端服务调用图像预处理、加载识别模型操作；最后，将识别结果返回到前端界面显示出来。

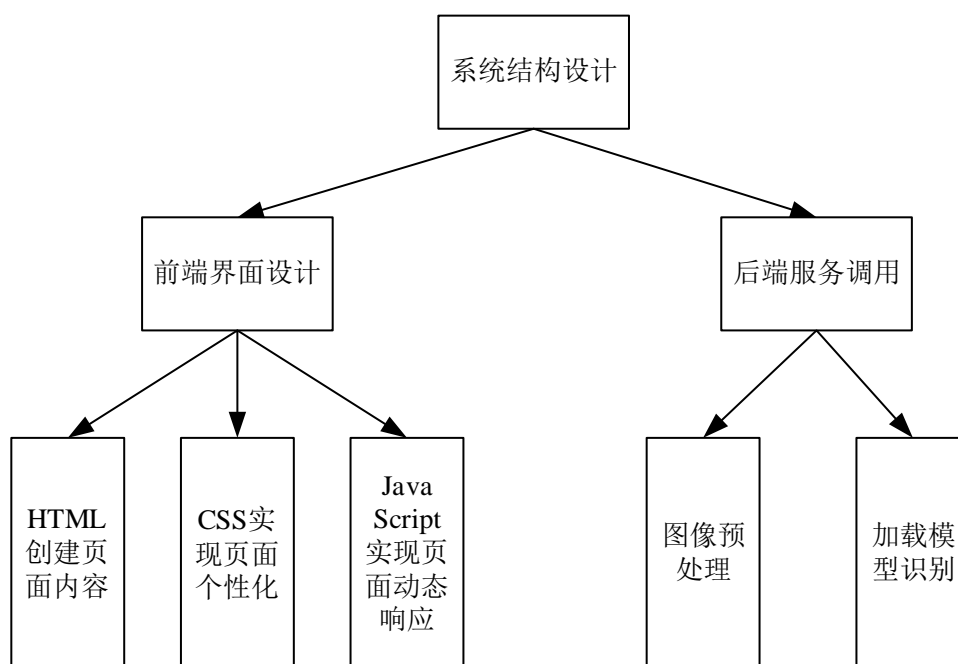


图 6.2 战国竹简文字识别系统结构设计

Fig. 6.2 Structural design of the Warring States Bamboo Slips character recognition system

### 6.2.2 系统功能设计

在对战国竹简文字识别系统的结构设计完成后，对其中的功能模块进行设计。如图 6.3 所示，战国竹简文字识别系统共包含登录注册模块、图像上传模块、图像预处理模块和图像识别模块四个部分。各模块功能如下所示：

- (1) 登录注册模块实现了用户的注册、登录和身份验证等功能；
- (2) 图像上传模块提供给用户图像选择和上传的功能；
- (3) 图像预处理模块负责对上传的图像进行预处理操作，包括像素填充和图片尺寸规范到 64×64，图像预处理操作后保存处理后的图像；

(4) 识别模块负责调用本文构建的战国竹简文字识别模型 (SSBN)，对预处理后的战国竹简文字图像进行识别并展示识别效果。

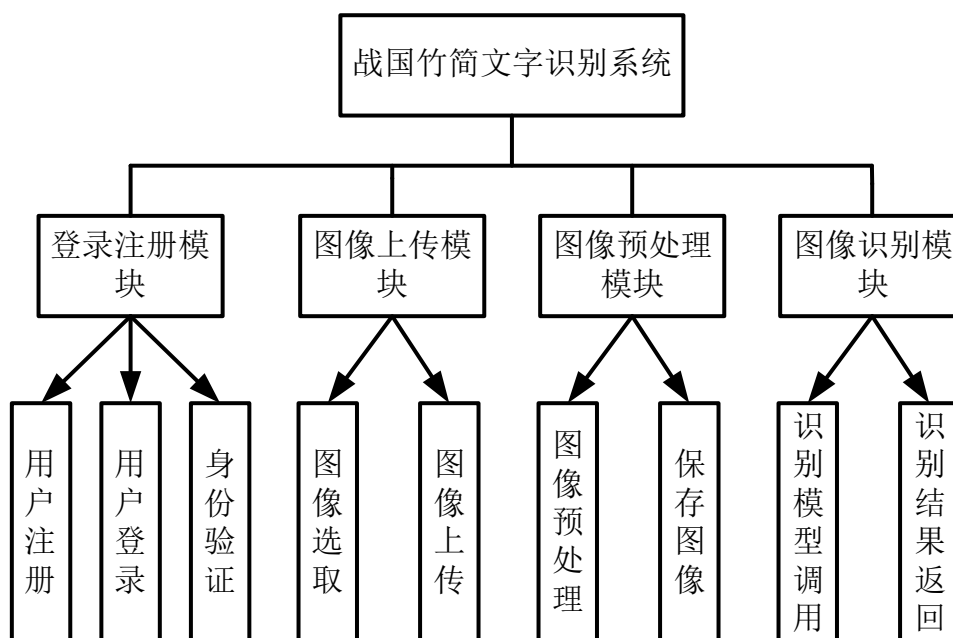


图 6.3 战国竹简文字识别系统功能设计

Fig. 6.3 Functional design of the Warring States Bamboo Slips character recognition system

### 6.2.3 数据库设计

在战国竹简文字识别系统中，数据库对数据信息的管理起着非常重要的作用。本系统设计了用户信息表来为了存储用户的注册信息。

本文基于 MySQL 设计了一个用户信息表，如表 6.2 所示。该表主要包括以下字段：用户 ID (User\_id)、用户名称 (User\_name)、密码 (Password) 等信息。其中，将 User\_id 和 Phone\_number 设置为主键，确保用户注册信息的唯一性。为了保护用户的安全，Password 字段采用加密方式存储。通过设计用户信息表，能够有效实现存储用户信息。

表 6.2 用户信息表

Tab. 6.2 User information table

| 字段        | 类型      | 长度 | 是否主键 | 是否允许为空 |
|-----------|---------|----|------|--------|
| User_id   | Int     | 20 | 是    | 否      |
| User_name | Varchar | 10 | 否    | 否      |
| Password  | Varchar | 20 | 否    | 否      |

## 6.3 系统应用与功能展示

### 6.3.1 用户注册功能

在用户首次使用战国竹简文字识别系统时，需要进入注册界面进行新用户的注册。具体步骤如下：

- (1) 输入注册信息。填写相应的注册信息，包括用户名和密码；
- (2) 密码重复性检查。系统对用户输入的密码进行检查，以确保密码校验无误。系统将比较用户两次输入的密码是否一致，不一致就重新输入注册信息；
- (3) 用户重复性检查。系统在数据库中对用户信息进行检查，如果不存在用户信息，就重新输入用户注册信息；
- (4) 保存用户信息。最后密码重复性检查和用户重复性检查校验无误后，将用户信息保存到数据库，返回注册成功提示。

战国竹简文字识别系统的用户注册流程图如图 6.4 所示。

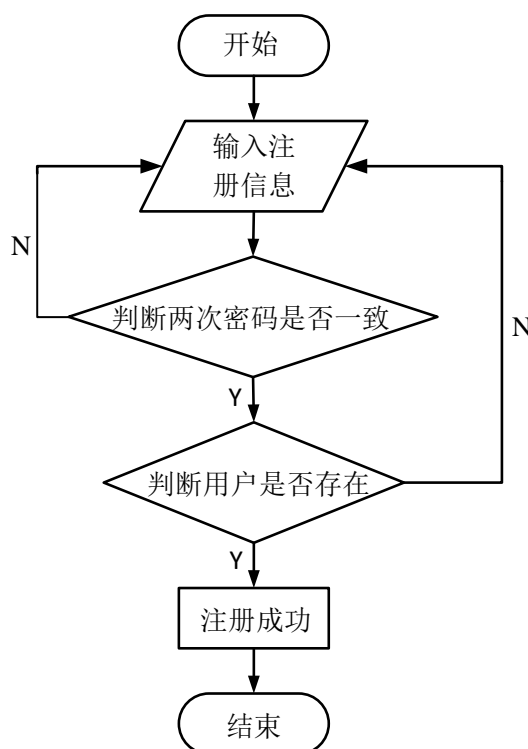


图 6.4 用户注册流程图

Fig. 6.4 User registration process diagram

战国竹简文字识别系统的用户注册界面如图 6.5 所示。



图 6.5 用户注册界面

Fig. 6.5 User registration interface

### 6.3.2 用户登录功能

用户注册完成后，通过用户登录功能来实现用户身份验证功能，提高战国竹简文字识别系统的安全性。战国竹简文字识别系统的用户登录流程图如图 6.6 所示，战国竹简文字识别系统的用户登录界面如图 6.7 所示。

当用户进入到战国竹简文字识别系统登录界面时，系统主要进行如下步骤：

- (1) 输入用户信息。输入用户名和密码并点击登录；
- (2) 用户信息判定。系统在数据库中查询用户信息并进行判定：若数据库中不存在用户信息，则提示用户进行注册；
- (3) 登录密码验证。若数据库中不存在用户信息，则对用户登录密码进行验证，验证不通过返回错误信息；
- (4) 登录系统。在用户信息存在且密码验证通过后进入系统。

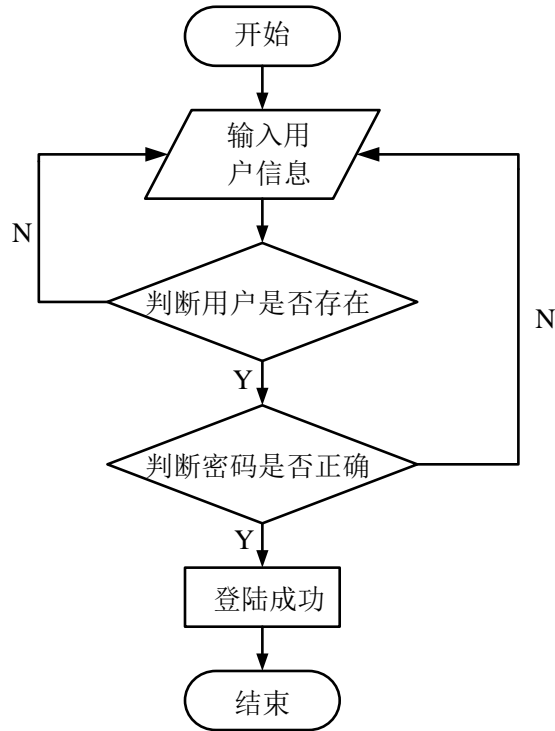


图 6.6 用户登录流程图

Fig. 6.6 User login flowchart



图 6.7 用户登录界面

Fig. 6.7 User login interface

### 6.3.3 图像预处理功能

用户登录系统后进入系统的主界面，如图 6.8 所示。主界面左边是功能区，功能区包含用户管理、图像预处理、识别、历史记录、设置和注销等功能。主界面右边为操作区，用户在操作区中进行上传图片文件和显示预处理后的图像以及识别结果。



图 6.8 系统主界面

Fig. 6.8 System main interface

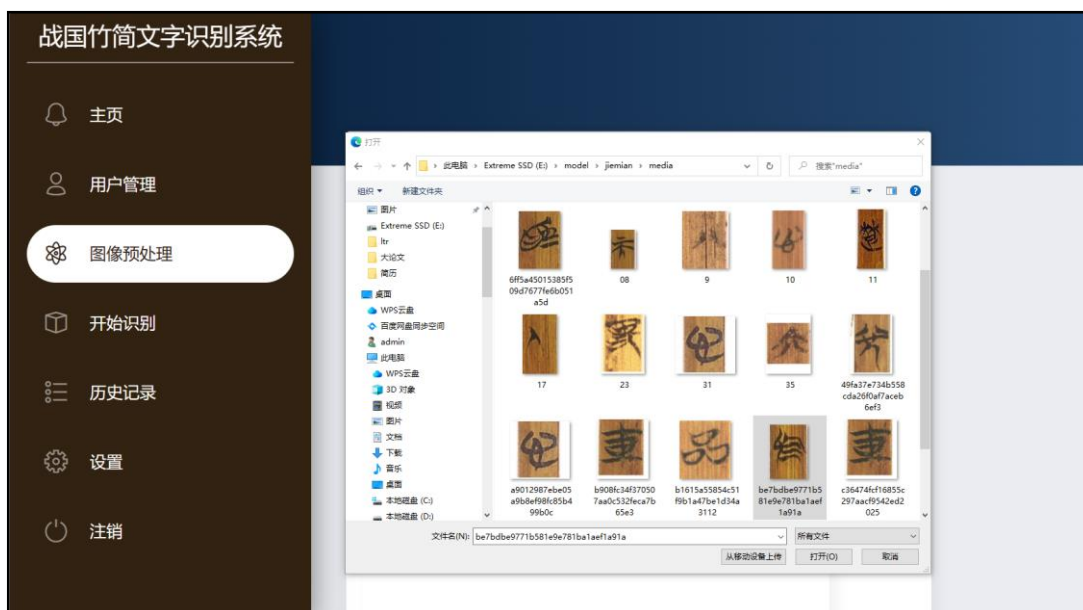


图 6.9 系统图像上传界面

Fig. 6.9 System image upload interface

图 6.9 为系统图像上传界面，在此处选择需要处理的图片。为了提高系统识别效果，首先对图像进行预处理操作，包括像素填充和规范图像尺寸，图像预处理操作后保存图像。图 6.10 为系统图像预处理结果，左边为原图片，右边为预处理后的图片。



图 6.10 系统图像预处理结果

Fig. 6.10 System image preprocessing results

#### 6.3.4 战国竹简文字识别功能

战国竹简文字识别功能为本系统最为核心的功能，对于用户上传的战国竹简文字图像，经过图像预处理后，用户可以使用识别功能对战国竹简文字进行识别。图 6.11 为系统识别的流程图，其步骤如下：

- (1) 读取预处理图像。系统开始读取预处理后的图像，并将图像送入到模型中；
- (2) 加载战国竹简文字识别模型。本文改进的 SSBN 模型识别性能较好，适合于战国竹简文字识别任务中，因此选择加载 SSBN 模型；
- (3) 进行识别。系统选择调用 SSBN 模型对图像进行识别，并返回识别结果和识别准确率在页面上。

为了验证本系统的性能，战国竹简文字识别数据集的 Many-shot 类、Medium-shot 类和 Few-shot 类的测试集上随机挑选图像进行结果测试。图 6.12 为 Many-shot 类测试结果，图 6.13 为 Medium-shot 类测试结果，图 6.14 为 Few-shot 类测试结果。

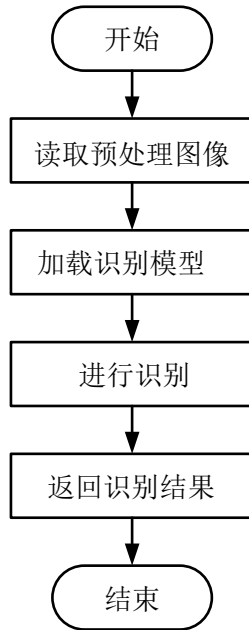


图 6.11 系统识别流程图

Fig. 6.11 System identification flowchart



图 6.12 系统在 Many-shot 类测试结果

Fig. 6.12 Test results of the system in the Many-shot class



图 6.13 系统在 Medium-shot 类测试结果  
Fig. 6.13 System test results in Medium-shot class

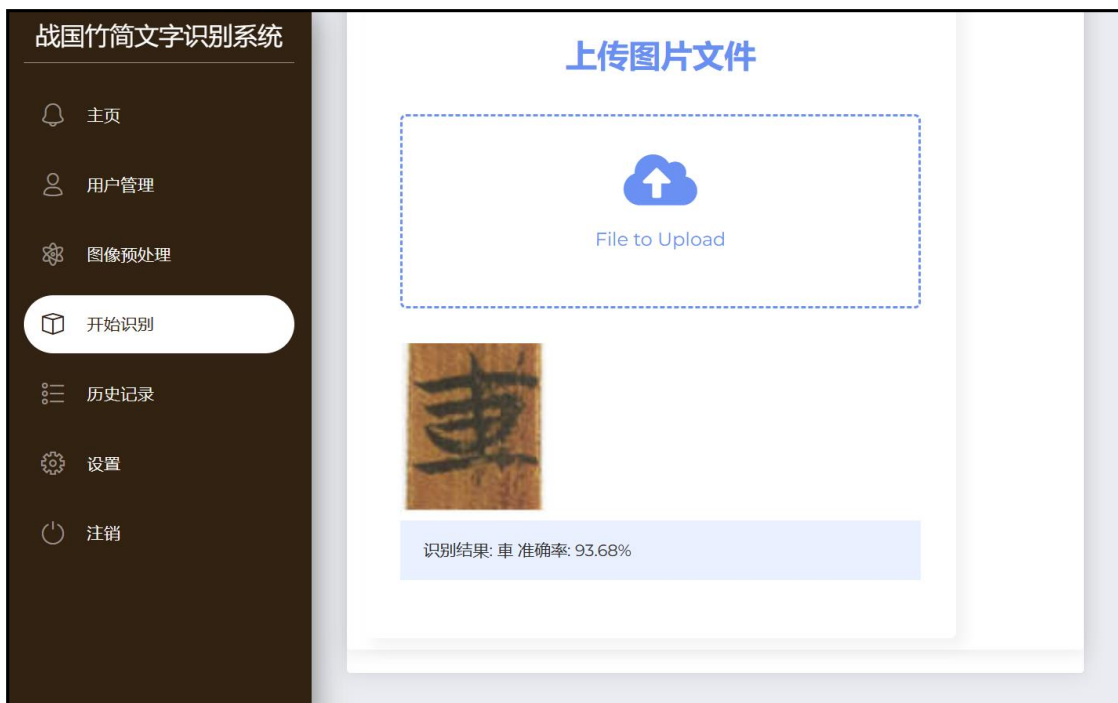


图 6.14 系统在 Few-shot 类测试结果  
Fig. 6.14 Test results of the system in the Few-shot class

由测试结果表明，本系统在各类子数据集上都保持着良好的识别效果。该系统能够使用在战国竹简文字识别任务中，为战国竹简文字整理研究工作提供准确和方便的帮助，具有一定的实用价值。

## 6.4 本章小结

本章主要进行战国竹简文字识别系统的设计与应用工作，从战国竹简文字识别模型的实际运用角度出发，通过利用 HTML、CSS 和 JavaScript 实现前端界面，结合基于 Flask 的轻量级 Web 框架实现后端服务，实现了对战国竹简文字识别系统的设计与应用。系统的设计主要分为系统结构设计、系统功能设计和数据库设计。其中：

### （1）系统结构设计

结构设计主要包括前端页面设计、后端服务实现。前端页面设计使用 HTML 创建了页面的结构和内容，使用 CSS 实现了页面的各种个性化样式，使用 JavaScript 实现了页面的动态交互功能。后端服务基于 Flask 框架实现，负责用户的请求和响应功能。

### （2）系统功能设计

系统功能设计共包含登录注册模块、图像上传模块、图像预处理模块和识别模块四个部分。登录注册模块实现了用户的注册、登录和身份验证功能。图像上传模块提供给用户图像选择和上传的功能。预处理模块负责对上传的图像进行预处理操作。识别模块负责调用本文构建的战国竹简文字识别模型，对预处理后的战国竹简文字图像进行识别并展示识别效果。

### （3）数据库设计

在战国竹简文字识别系统中，数据库对数据信息的管理起着非常重要的作用。本系统设计了用户信息表来存储用户的注册信息，为系统的登录和注册功能提供了基础。通过设计用户信息表，能够有效存储用户信息。

最后，在战国竹简文字识别数据集的 Many-shot 类、Medium-shot 类和 Few-shot 类的测试集上随机挑选图像进行结果测试。结果表明，该系统实现了对战国竹简文字的准确识别，能够满足用户对战国竹简文字识别的需求，具有一定的实用价值。

## 7 总结与展望

### 7.1 总结

针对战国竹简文字识别上存在缺少公开的战国竹简文字识别数据集以及战国竹简文字本身存在的竹木纹路干扰、墨迹干扰、部分字体断裂和严重的样本数量不均衡等问题,本文主要完成了以下几方面的内容:

(1) 构建了战国竹简文字识别数据集。本文以“清华简”为研究原型,首先从竹简上搜集文字图像数据,并对图像数据进行标注工作;然后进行数据划分,按照样本数量划分成 Many-shot 类, Medium-shot 类和 Few-shot 类;最后进行数据预处理和数据增强,增强后共 1001 类战国竹简文字, 26564 张图像数据,即完成战国竹简文字识别数据集 zgzt1001 的制作。

(2) 针对战国竹简上存在的墨迹、部分字体断裂、竹木纹路等干扰信息会影响模型的识别效果的问题。本文改进了空间自注意力机制来关注战国竹简文字的主体文字信息,减小了竹简上干扰信息对模型的识别造成的影响,并将其应用于 ResNet50 网络上,搭建了基于空间自注意力机制的战国竹简文字识别模型 (SSA-ResNet50)。实验结果表明,与改进前 ResNet50 模型相比, SSA-ResNet50 模型在平衡准确率上提高了 1.67%, F1 分数提高了 1.34%。

(3) 针对战国竹简文字识别数据集本身呈现长尾分布,其严重的样本数量不均衡会造成模型训练困难和模型识别性能差的问题。本文以 SSA-ResNet50 为基础算法,将双分支网络 (BBN) 和自监督对比学习中的 SimSiam 算法结合在一起,搭建了基于自监督双分支网络的战国竹简文字识别模型 (SBBN)。实验结果表明, SBBN 模型在 zgzt1001 上平衡准确率为 97.56%、F1 分数为 97.38%。与 SSA-ResNet50 模型相比, SBBN 模型在 zgzt1001 上平衡准确率提高了 2.72%, F1 分数提高了 2.97%。在 Few-shot 类上平衡准确率提高了 5.58%,提高了尾部类别样本的识别精度。最终,在 Many-shot 类、Medium-shot 类和 Few-shot 类分别达到了 98.42%、97.45%和 96.90%的平衡准确率,证明了 SBBN 模型可以在样本数量不均衡的情况下可以保持较好的识别效果。

(4) 从战国竹简文字识别模型的实际运用角度出发,通过利用 HTML、CSS 和 JavaScript 实现前端界面,结合基于 Flask 的轻量级 Web 框架实现后端服务,实现了战国竹简文字识别系统的设计。对系统进行了用户注册登录功能、图像预处理功能和战国竹简文字识别功能展示。结果表明该系统实现了对战国竹简文字的准确识别,能够满足用户对战国竹简文字识别任务的需求,具有一定的实用价值。

## 7.2 展望

虽然本文在战国竹简文字识别中取得了一些进展，但是依然有许多问题需要解决和改进，因此，本文对后续工作给出以下展望：

(1) 扩大战国竹简文字识别数据集。本文以“清华简”为研究对象，构建了一个包含 1001 种类别的战国竹简文字识别数据集。但是，部分文字的现代注释没有在计算机里进行编码，无法对其进行数据标注工作。后续工作可以考虑对不存在的字体进行自建字符来实现数据标注，从而实现扩大战国竹简文字识别数据集。

(2) 减小模型复杂度。本文改进的模型可以初步解决战国竹简文字识别上的问题，实现对战国文字的智能识别。但是，在训练过程中，训练难度增大，对训练设备的要求也提高。后续工作可以考虑减少模型的参数量，如改进更加轻量级的模型进行训练。

(3) 实现模型检测功能。本文战国竹简文字识别系统主要实现了战国竹简文字的单类字符识别，在实际运用中，需要人工截取需要识别的文字。后续工作后可以通过借鉴目标检测算法训练一个检测模型，实现对战国竹简上的所有文字自动化截取工作，提高系统的实用性。

## 参考文献

- [1] 何家源. “长尾分布”的古文字分类算法研究[D].上海:华东师范大学,2022.
- [2] Guo J and Wang C, Roman-Rangel E, et al. Building hierarchical representations for oracle character and sketch recognition[J]. *IEEE Transactions on Image Processing*, 2015, 25(1): 104-118.
- [3] 顾绍通.基于拓扑配准的甲骨文字形识别方法[J].*计算机与数字工程*,2016,44(10):2001-2006.
- [4] 刘永革,刘国英.基于 SVM 的甲骨文字识别[J].*安阳师范学院学报*,2017,(02):54-56.
- [5] Kaur S and Sagar BB. Brahmi character recognition based on SVM (support vector machine) classifier using image gradient features[J]. *Journal of Discrete Mathematical Sciences and Cryptography*, 2019, 22(8):1365-1381.
- [6] Narang S R, Jindal M K, Ahuja S, et al. On the recognition of Devanagari ancient handwritten characters using SIFT and Gabor features[J]. *Soft Computing*, 2020, 24(22): 17279-17289.
- [7] 王浩彬. 基于深度学习的甲骨文检测与识别研究[D].广州:华南理工大学,2019.
- [8] 刘梦婷. 基于深度卷积神经网络的甲骨文字识别研究[D].郑州:郑州大学,2020.
- [9] Zhao H, Chu H, Zhang Y, et al. Improvement of Ancient Shui Character Recognition Model Based on Convolutional Neural Network[J]. *IEEE Access*, 2020, 8: 33080-33087.
- [10] 田园. 基于深度度量学习的战国简文字识别技术[D].郑州:河南大学,2020.
- [11] 吴炫奇. 基于深度学习的商周金文文字识别研究[D].上海:华东师范大学,2021.
- [12] Narang SR, Kumar M and Jindal MK. DeepNetDevanagari: a deep learning model for Devanagari ancient character recognition[J]. *Multimedia Tools and Applications*, 2021, 80(13):20671-20686.
- [13] Chen S, Yang Y, Liu X, et al. Dual discriminator GAN: Restoring ancient Yi characters[J]. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 2022, 21(4):1-23.
- [14] Tang M, Xie S and Liu X. Ancient Character Recognition: A Novel Image Dataset of Shui Manuscript Characters and Classification Model[J]. *Chinese Journal of Electronics*, 2023, 32(1):64-75.
- [15] Chandure S and Inamdar V. Handwritten Modi character recognition using transfer learning with discriminant feature analysis[J]. *IETE Journal of Research*, 2023, 69(5):2584-2594.
- [16] 荆门市博物馆.郭店楚墓竹简[M].北京:文物出版社,1998.
- [17] 马承源.上海博物馆藏战国楚竹书（一）[M].上海:上海古籍出版社,2001.
- [18] 李学勤.清华大学藏战国竹简（壹）[M].上海:中西书局出版社,2011.
- [19] 黄德宽,徐在国. 安徽大学藏战国竹简（一）[M]. 上海:中西书局出版社, 2019.
- [20] 安徽大学藏战国竹简（二）[M].上海:中西书局出版社,2022.
- [21] Wang P, Fan E and Wang P. Comparative analysis of image classification algorithms based on traditional machine learning and deep learning[J]. *Pattern Recognition Letters*. 2021, 141:61-67.
- [22] Luo W, Lu J, Li X, et al. Rethinking motivation of deep neural architectures[J]. *IEEE Circuits and Systems Magazine*, 2020, 20(4):65-76.
- [23] Wang S, Huang L, Gao A, et al. Machine/deep learning for software engineering: A systematic literature review[J]. *IEEE Transactions on Software Engineering*, 2022, 49(3):1188-1231.

- [24] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, June 27-July 1, 2016, PP. 770-778.
- [25] Brauwiers G and Frasincar F. A general survey on attention mechanisms in deep learning[J]. IEEE Transactions on Knowledge and Data Engineering, 2023,35(4):3279-3298.
- [26] Woo S, Park J, Lee J Y, et al. Cbam: Convolutional block attention module[C]. Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, September 8-14, 2018, pp. 3-19.
- [27] J Masci, U Meier, D Ciresan, et al. Steel defect classification with max-pooling convolutional neural networks[C]. The 2012 International Joint Conference on Neural Networks (IJCNN), Brisbane, QLD, Australia, June 10-15, 2012, pp. 1-6.
- [28] Zhao S, You F, Chang W, et al. Augment BERT with average pooling layer for Chinese summary generation[J]. Journal of Intelligent & Fuzzy Systems, 2022, 42(3):1859-1868.
- [29] Elfving S, Uchibe E and Doya K. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning[J]. Neural Networks, 2018, 107:3-11.
- [30] Krizhevsky A, Sutskever I and Hinton GE. ImageNet classification with deep convolutional neural networks[J]. Communications of the ACM, 2017,60(6):84-90.
- [31] Wang X, Girshick R, Gupta A, et al. Non-local neural networks[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, June 18-22, 2018, pp.7794-7803.
- [32] Yang L, Jiang H, Song Q, et al. A survey on long-tailed visual recognition[J]. International Journal of Computer Vision, 2022,130(7):1837-1872.
- [33] Noor S, Tajik O and Golzar J. Simple random sampling[J]. International Journal of Education & Language Studies, 2022, 1(2):78-82.
- [34] Abdi L and Hashemi S. To combat multi-class imbalanced problems by means of over-sampling techniques[J]. IEEE Transactions on Knowledge and Data Engineering, 2015, 28(1):238-251.
- [35] Liu T, Zhu X, Pedrycz W, et al. A design of information granule-based under-sampling method in imbalanced data classification[J]. Soft Computing, 2020, 24(22):17333-17347.
- [36] Lin C, Tsai CF and Lin WC. Towards hybrid over-and under-sampling combination methods for class imbalanced datasets: an experimental study[J]. Artificial Intelligence Review, 2023, 56(2):845-863.
- [37] Chawla NV, Bowyer KW, Hall LO, et al. SMOTE: synthetic minority over-sampling technique[J]. Journal of Artificial Intelligence Research, 2002, 16:321-357.
- [38] 范峻植. 面向长尾分布的图像识别算法研究[D].成都:电子科技大学,2022.
- [39] Qin J, Liu G and Duan K. A New Imbalanced Encrypted Traffic Classification Model Based on CBAM and Re-Weighted Loss Function[J]. Applied Sciences, 2022, 12(19):9631.
- [40] Cui Y, Jia M, Lin TY, et al. Class-balanced loss based on effective number of samples[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, June 16-20, 2019, pp. 9268-9277.
- [41] Ren J, Yu C, Ma X, et al. Balanced meta-softmax for long-tailed visual recognition[C]. Advances in Neural Information processing Systems (NeurIPS), Virtual, December 6-12, 2020, pp. 4175-4186.

- [42] Lin TY, Goyal P, Girshick R, et al. Focal loss for dense object detection[C]. Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, October 22-29, 2017, pp. 2980-2988.
- [43] Zhuang F, Qi Z, Duan K, et al. A comprehensive survey on transfer learning[J]. Proceedings of the IEEE, 2020, 109(1):43-76.
- [44] Wang YX, Ramanan D and Hebert M. Learning to model the tail[C]. Advances in Neural Information Processing Systems (NeurIPS), Long Beach, CA, USA, December 4-9, 2017, pp. 30.
- [45] Cui Y, Song Y, Sun C, et al. Large scale fine-grained categorization and domain-specific transfer learning[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, June 18-22, 2018, pp. 4109-4118.
- [46] Xiang L, Ding G and Han J. Learning from multiple experts: Self-paced knowledge distillation for long-tailed classification[C]. Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23-28, 2020, pp. 247-263.
- [47] B Kang, S Xie, M Rohrbach, Z Yan, et al. Decoupling representation and classifier for long-tailed recognition[C]. Published in International Conference on Learning Representations (ICLR), Virtual, April 26-May 1, 2020.
- [48] Zhou B, Cui Q, Wei XS, et al. Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, June 13-19, 2020, pp. 9719-9728.
- [49] Peng S, Sun S and Yao YD. A survey of modulation classification using deep learning: Signal representation and data preprocessing[J]. IEEE Transactions on Neural Networks and Learning Systems, 2021, 33(12):7020-7038.
- [50] Yu L and Gao XS. Improve robustness and accuracy of deep neural network with L<sub>2</sub>,  $\infty$  normalization[J]. Journal of Systems Science and Complexity, 2023, 36(1):3-28.
- [51] Khalifa NE, Loey M and Mirjalili S. A comprehensive survey of recent trends in deep learning for digital images augmentation[J]. Artificial Intelligence Review, 2022, 55(3):2351-2377.
- [52] Menghani G. Efficient deep learning: A survey on making deep learning models smaller, faster, and better[J]. ACM Computing Surveys, 2023, 55(12):1-37.
- [53] Zhang Q, Zhang M, Wang M, et al. Efficient deep learning inference based on model compression[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, June 18-22, 2018, pp. 1695-1702.
- [54] Zhou BC, Han CY and Guo TD. Convergence of stochastic gradient descent in deep neural network[J]. Acta Mathematicae Applicatae Sinica-English Series, 2021, 37(1):126-136.
- [55] Loshchilov I and Hutter F. SGDR: Stochastic Gradient Descent with Warm Restarts[C]. In Proceedings of the 5th International Conference on Learning Representations (ICLR), Toulon, France, April 24-26, 2017, pp.1-16.
- [56] Ho Y and Wookey S. The real-world-weight cross-entropy loss function: Modeling the costs of mislabeling[J]. IEEE Access, 2019, 8:4806-4813.
- [57] Krizhevsky A, Sutskever I and Hinton GE. ImageNet classification with deep convolutional neural networks[J]. Communications of the ACM, 2017,60(6):84-90.

- [58] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, June 7-12, 2015, pp. 1-9.
- [59] Simonyan K and Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition[C]. International Conference on Learning Representations (ICLR), San Diego, CA, USA, May 7-9, 2015, pp. 1-14.
- [60] Liu Z, Mao H, Wu CY, et al. A convnet for the 2020s[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, June 19-24, 2022, pp. 11976-11986.
- [61] Liu X, Zhang F, Hou Z, et al. Self-supervised learning: Generative or contrastive[J]. IEEE Transactions on Knowledge and Data Engineering, 2021, 35(1):857-876.
- [62] Li Z, Yu D, Wu M, et al. Adversarial supervised contrastive learning[J]. Machine Learning, 2023, 112(6): 2105-2130.
- [63] Chen X and He K. Exploring simple siamese representation learning[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, June 10-25, 2021, pp.15750-15758.
- [64] Xie TY and Ge Y. Enhance Knowledge Graph Embedding by Mixup[J]. IEEE Transactions on Knowledge and Data Engineering, 2024, 36(2):569-580.
- [65] Takahashi R, Matsubara T and Uehara K. Data augmentation using random image cropping and patching for deep CNNs[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2019, 30(9):2917-2931.
- [66] Hajiabadi H, Molla-Aliod D, Monsefi R, et al. Combination of loss functions for deep text classification[J]. International Journal of Machine Learning and Cybernetics, 2020, 11(4): 751-761.
- [67] Lee MC, Chiu SY and Chang JW. A deep convolutional neural network based Chinese menu recognition app[J]. Information Processing Letters, 2017, 128:14-20.
- [68] Durga BK and Rajesh V. A ResNet deep learning based facial recognition design for future multimedia applications[J]. Computers and Electrical Engineering, 2022, 104:108384.
- [69] 吕云翔,刘猛猛,欧阳植昊等.HTML5 基础与实践教程[M].北京:机械工业出版社,2020.
- [70] 张鑫旭.CSS 世界[M].北京:人民邮电出版社,2017.
- [71] 吕云翔,欧阳植昊,张远.JavaScript 前端开发基础教程[M]. 北京:机械工业出版社,2021.