

# 基于深度学习的骨髓血细胞检测与 识别技术研究

(申请清华大学电子信息硕士专业学位论文)

培养单位: 电子工程系

专业领域: 电子信息

申请人: 孙天宇

指导教师: 杨健教授

二〇二三年四月

# **Research on bone marrow blood cell detection and recognition based on deep learning**

Thesis submitted to  
**Tsinghua University**  
in partial fulfillment of the requirement  
for the professional degree of  
**Master of Electronic and Information Engineering**

by  
**Sun Tianyu**  
**(Electronic and Information Engineering)**

Thesis Supervisor: Professor Yang Jian

**April, 2023**

# 学位论文指导小组、公开评阅人和答辩委员会名单

## 指导小组名单

杨健 教授 清华大学

## 公开评阅人名单

刘 XX 教授 清华大学  
杨 XX 研究员 中国 XXXX 科学院 XXXXXXX 研究所

## 答辩委员会名单

主席	赵 XX	教授	清华大学
委员	刘 XX	教授	清华大学
	杨 XX	研究员	中国 XXXX 科学院 XXXXXXX 研究所
	黄 XX	教授	XXXX 大学
	周 XX	副教授	XXXX 大学
秘书	吴 XX	博士	清华大学

# 关于学位论文使用授权的说明

本人完全了解清华大学有关保留、使用学位论文的规定，即：

清华大学拥有在著作权法规定范围内学位论文的使用权，其中包括：（1）已获学位的研究生必须按学校规定提交学位论文，学校可以采用影印、缩印或其他复制手段保存研究生上交的学位论文；（2）为教学和科研目的，学校可以将公开的学位论文作为资料在图书馆、资料室等场所供校内师生阅读，或在校园网上供校内师生浏览部分内容；（3）按照上级教育主管部门督导、抽查等要求，报送相应的学位论文。

本人保证遵守上述规定。

作者签名：\_\_\_\_\_

导师签名：\_\_\_\_\_

日 期：\_\_\_\_\_

日 期：\_\_\_\_\_

## 摘要

白血病是一种常见多发且较为凶险的血液疾病，其早期发现与治疗对延长患者生存时间、改善患者生活质量至关重要。目前白血病的临床诊断主要依靠医师对骨髓穿刺涂片进行形态学检查。人工镜检存在着繁琐费时、结果主观性强等缺点，并且培养病理诊断医师要耗费大量时间。近年来，计算机科学技术高速发展，人工智能技术被广泛应用于医学领域，其高效的计算与分析能力为疾病诊疗带来了深刻的变革。目前诸多研究学者开始研究基于深度学习的方法来自动定位与识别血细胞，上述研究主要基于通用的目标检测识别网络，缺少针对骨髓血细胞特性的改进。因此，我们与邃蓝智能科技(上海)有限公司进行合作，针对骨髓血细胞检测与识别关键问题进行研究，提高骨髓血细胞检测识别算法精度，并编写相关软件，为医生的临床诊断提供参考依据。

首先，在数据方面，我们采用主动学习技术实现了骨髓血细胞数据标注迭代框架，并在医生的协作下快速完成了多批次骨髓血细胞边界框与类别信息的标注。在骨髓血细胞检测方面，我们对比了多种单阶段与双阶段目标检测网络，将性能优异 RetinaNet 作为基线模型。我们对比了 RetinaNet 网络在仅检测与检测识别一体化任务上准确率的差异，并确定了先检测再识别的骨髓血细胞处理流程。针对密集、堆叠与黏连骨髓血细胞区域的漏检、错检等问题，我们提出了全局注意力自下而上路径聚合网络结构，IOU 预测分支与最优输运的全局最优标签分配策略，改进方法平均检测精度相比于主流目标检测网络提升了 1% 以上，达到了较为先进的性能。在骨髓血细胞识别方面，我们基于 Vision Transformer 网络实现了骨髓血细胞的细粒度识别。在该方法中，我们采用基于滑动窗的图像块划分方法，保留局部细节信息。此外，我们提出了辨识性区域选择模块，引导网络对细胞核、细胞质等细微差异区域进行聚焦。引入了对比损失，约束骨髓血细胞的类间差异性与类内聚集性。该方法在慕尼黑骨髓血细胞数据集上的平均分类准确率为 91.96%，相比于基线模型提升了 0.74%。最后，我们基于 B/S 架构设计并开发了骨髓血细胞检测与识别软件，将上述骨髓血细胞检测识别算法进行落地部署，软件可以自动完成骨髓血细胞的定位与分类计数，用于辅助医生临床诊断。

**关键词：**医学图像分析；骨髓血细胞；深度学习；检测识别

## Abstract

Leukemia is a common, multiple and dangerous blood disease, whose early diagnosis and treatment are crucial to prolong the survival time and improve the quality of patients' live. At present, the clinical diagnosis of leukemia mainly relies on the morphological examination of bone marrow aspiration smear by physicians. Manual microscopic examination is very cumbersome and time-consuming and the results are highly subjective. In addition, it takes a lot of time to train experienced pathologists. In recent years, with the rapid development of computer science and technology, artificial intelligence technology has been widely used in the medical field. Its efficient computing and analysis capabilities have brought profound changes to the diagnosis and treatment of diseases. At present, many researchers have begun to study methods based on deep learning to automatically locate and identify blood cells. The above researches are mainly based on general-purpose target detection and recognition networks and lack improvements for the characteristics of bone marrow blood cells. Therefore, we cooperate with Deep Voxel Intelligent Technology to research the key issues of bone marrow blood cell detection and recognition. We aim to improve the accuracy of bone marrow blood cell detection and recognition and make software to provide reference for doctors' clinical diagnosis.

First, in terms of dataset, we used active learning technology to realize the iterative framework for bone marrow blood cell data annotations and quickly completed the annotations of bone marrow blood cell dataset with the cooperation of doctors. In terms of bone marrow blood cell detection, we compared a variety of one-stage and two-stage target detection networks. We finally used RetinaNet as the baseline model. We compared the accuracy of the RetinaNet network between the task of only detection and the integration task of detection and recognition and determined the processing flow of detection first and then recognition. Aiming at the missed detection and wrong detection of stacked and cohesive bone marrow blood cell regions, we propose a global attention bottom-up path aggregation structure, IOU prediction branches and optimal transport label assignment strategy. Compared with the mainstream target detection network, our method improved the average detection accuracy by more than one percent and achieved state of the art. In terms of bone marrow blood cell recognition, we realized the fine-grained recognition of bone marrow blood cells based on the Vision Transformer network. In this method,

---

## Abstract

---

we adopt a sliding window-based image patch division method to preserve local detail information. In addition, we propose a discriminative region selection module to guide the network to focus on nuanced regions such as nucleus and cytoplasm. A contrast loss is introduced to constrain the inter-class diversity and intra-class aggregation of bone marrow blood cells. The average classification accuracy of this method on the Munich bone marrow blood cell dataset is 91.96 percent, which is 0.74 percent higher than the baseline model. Finally, we designed and developed the bone marrow blood cell detection and recognition software based on the B/S architecture. We deployed the above bone marrow blood cell detection and recognition algorithm in this software. The software can automatically complete the positioning, classification and counting of bone marrow blood cells to assist doctors in clinical diagnosis.

**Keywords:** Medical Image Analysis; Bone Marrow Blood Cells; Deep Learning; Detection and Recognition

## 目 录

摘 要.....	I
Abstract.....	II
目 录.....	IV
插图清单.....	VII
附表清单.....	IX
符号和缩略语说明.....	X
第 1 章 绪论 .....	1
1.1 研究背景与意义 .....	1
1.2 研究现状与进展 .....	2
1.2.1 骨髓血细胞图像检测现状 .....	2
1.2.2 骨髓血细胞图像识别现状 .....	4
1.3 本文研究内容 .....	6
1.4 论文组织结构 .....	7
第 2 章 基础知识 .....	10
2.1 骨髓血细胞图像及预处理 .....	10
2.1.1 骨髓血细胞形态学介绍 .....	10
2.1.2 骨髓血细胞数据集与预处理 .....	12
2.2 神经网络技术概述 .....	15
2.2.1 神经元与梯度优化 .....	15
2.2.2 卷积神经网络 .....	18
2.3 软件开发相关技术 .....	20
2.3.1 前端技术 .....	20
2.3.2 后端技术 .....	22
第 3 章 基于深度学习的骨髓血细胞检测算法设计与实现 .....	25
3.1 引言 .....	25
3.2 双阶段目标检测网络 .....	25
3.2.1 骨干网络 .....	25

3.2.2 特征金字塔网络.....	27
3.2.3 区域举荐网络.....	27
3.2.4 分类与回归网络.....	29
3.3 单阶段目标检测网络 .....	30
3.3.1 网络结构.....	30
3.3.2 焦点损失函数.....	31
3.3.3 网络预测.....	32
3.4 算法实现与实验结果分析 .....	32
3.4.1 实验环境.....	32
3.4.2 实验结果与分析.....	35
3.5 小结 .....	40
<b>第 4 章 基于改进的 RetinaNet 骨髓血细胞检测算法设计与实现 .....</b>	<b>42</b>
4.1 引言 .....	42
4.2 改进的 RetinaNet 骨髓血细胞检测网络 .....	42
4.2.1 基于全局注意力的路径聚合网络.....	43
4.2.2 IOU 预测分支.....	45
4.2.3 训练标签分配策略.....	46
4.3 算法实现与实验结果分析 .....	51
4.3.1 实验环境.....	51
4.3.2 实验结果与分析.....	52
4.4 小结 .....	56
<b>第 5 章 基于改进 Vision Transformer 骨髓血细胞检测算法设计与实现.....</b>	<b>58</b>
5.1 引言 .....	58
5.2 改进的 Vision Transformer 骨髓血细胞识别网络 .....	59
5.2.1 重叠图像块划分.....	59
5.2.2 编码层.....	61
5.2.3 稀疏注意力模块.....	62
5.2.4 损失函数.....	64
5.3 实验结果分析 .....	65
5.3.1 数据集介绍.....	65
5.3.2 实验环境与评价指标.....	65
5.3.3 实验结果.....	66
5.3.4 消融实验.....	70

## 目 录

---

5.4 小结 .....	72
<b>第 6 章 骨髓血细胞检测与识别软件设计 .....</b>	<b>74</b>
6.1 需求分析 .....	74
6.1.1 功能需求分析.....	74
6.1.2 非功能需求分析.....	75
6.2 软件设计 .....	75
6.2.1 软件架构设计 .....	75
6.2.2 软件数据库设计.....	76
6.3 各个模块设计 .....	77
6.3.1 用户模块.....	78
6.3.2 骨髓血细胞检测模块.....	79
6.3.3 骨髓血细胞识别模块.....	80
6.3.4 患者数据管理模块.....	80
6.4 小结 .....	82
<b>第 7 章 总结与展望 .....</b>	<b>84</b>
7.1 全文总结 .....	84
7.2 工作展望 .....	86
<b>参考文献 .....</b>	<b>87</b>
<b>致 谢 .....</b>	<b>92</b>
<b>声 明 .....</b>	<b>93</b>
<b>个人简历、在学期间完成的相关学术成果 .....</b>	<b>94</b>
<b>指导教师评语 .....</b>	<b>95</b>
<b>答辩委员会决议书 .....</b>	<b>96</b>

## 插图清单

图 1.1 论文组织结构 .....	8
图 2.1 骨髓血细胞发育成熟过程示意图 .....	10
图 2.2 骨髓血细胞检验技术流程: (a) 骨髓穿刺抽取骨髓液, (b) 制备骨髓涂片, (c) 瑞特-吉姆萨染色, (d) 清洗晾干, (e) 人工镜检判读, (f) 骨髓形态学图片	13
图 2.3 labelme 标注软件工具 .....	14
图 2.4 主动学习标注框架示意图 .....	14
图 2.5 神经元结构示意图 .....	15
图 2.6 感知机网络模型示意图 .....	16
图 2.7 典型卷积神经网络结构示意图 .....	19
图 2.8 CNN 卷积层示意图 .....	19
图 2.9 CNN 池化层示意图 .....	20
图 2.10 MVVM 框架示意图 .....	22
图 2.11 Django 框架示意图 .....	23
图 3.1 快速区域卷积神经网络结构示意图 .....	26
图 3.2 ResNet50 网络结构示意图 .....	26
图 3.3 BottleNeck 模块结构示意图 .....	27
图 3.4 特征金字塔网络结构示意图 .....	28
图 3.5 特征金字塔网络结构示意图 .....	28
图 3.6 (a) Anchor 在图像中的示意图, (b) Anchor、预测框与真值框之间的关系	29
图 3.7 分类与回归网络结构示意图 .....	30
图 3.8 分类与回归网络结构示意图 .....	31
图 3.9 IOU 计算示意图 .....	33
图 3.10 检测类型判别 .....	34
图 3.11 四种检测网络的 PR 曲线 (IOU=0.75) .....	36
图 3.12 四种检测网络的 PR 曲线 .....	37
图 3.13 四种检测网络的可视化检测结果 .....	38
图 3.14 RetinaNet 检测识别与检测网络的混淆矩阵 .....	40
图 3.15 检测网络与检测识别网络特征图对比 .....	40
图 4.1 RetinaNet 基线模型检测错误示例 .....	43
图 4.2 改进的 RetinaNet 网络结构示意图 .....	43

图 4.3 路径聚合网络结构示意图 .....	44
图 4.4 全局注意力模块 .....	44
图 4.5 置信度高但交并比低的错误检测示例 .....	45
图 4.6 交并比预测分支结构 .....	46
图 4.7 自适应样本选择阈值计算示意图 .....	48
图 4.8 基于最优输运的标签分配策略 .....	50
图 4.9 骨髓血细胞数据集上的检测结果 .....	55
图 5.1 基于改进 Vision Transformer 骨髓血细胞识别网络结构 .....	60
图 5.2 位置编码可视化 .....	61
图 5.3 Vision Transformer 编码模块结构 .....	62
图 5.4 不同类别血细胞的辨识性区域 .....	63
图 5.5 稀疏注意力模块结构示意图 .....	64
图 5.6 改进 Vision Transformer 的识别混淆矩阵 .....	67
图 5.7 可视化自注意力图 .....	69
图 5.8 稀疏注意力模块选择的图像块 .....	69
图 5.9 图像块划分方式对模型性能的影响 .....	71
图 5.10 t-SNE 降维可视化结果 .....	72
图 6.1 骨髓血细胞检测识别软件架构图 .....	76
图 6.2 骨髓血细胞检测识别软件界面 .....	79
图 6.3 用户模块界面示意图 .....	79
图 6.4 骨髓血细胞检测界面 .....	80
图 6.5 骨髓血细胞识别界面 .....	81
图 6.6 医生核验识别结果界面 .....	81
图 6.7 患者数据分析界面 .....	82
图 6.8 患者信息录入页面 .....	83

## 附表清单

表 2.1 骨髓血细胞形态学特征 .....	11
表 3.1 骨髓血细胞检测数据集分布 .....	33
表 3.2 不同网络结构参数量、计算量与速度对比 .....	35
表 3.3 不同目标检测方法在骨髓血细胞测试集上的检测结果 .....	36
表 3.4 RetinaNet 网络各类别的准确率与召回率 .....	39
表 4.1 骨髓血细胞检测数据集分布 .....	53
表 4.2 改进 RetinaNet 骨髓血细胞检测结果 .....	54
表 4.3 不同目标检测方法在骨髓血细胞测试集上的性能对比 .....	54
表 4.4 骨髓血细胞检测网络模块消融实验 .....	55
表 4.5 标签分配策略消融实验 .....	56
表 5.1 TMAMD 血细胞数据集数据分布情况 .....	66
表 5.2 改进 Vision Transformer 方法的识别精确率与召回率 .....	67
表 5.3 不同识别方法性能对比 .....	68
表 5.4 不同 Transformer 结构的速率与准确率 .....	70
表 5.5 不同图像块划分方式的消融研究 .....	71
表 5.6 对比损失的消融研究 .....	72
表 6.1 用户信息表 .....	77
表 6.2 检测文件表 .....	78
表 6.3 识别文件表 .....	78

## 符号和缩略语说明

AP	平均精度 (Average Precision)
BP	反向传播 (Back Propagation)
RPN	区域推荐网络 (Region Proposal Network)
CNN	卷积神经网络 (Convolutional Neural Network)
FPN	特征金字塔 (Feature Pyramid Network)
ROI	感兴趣区域 (Region Of Interest)
IOU	交并比 (Intersection Over Union)
ReLU	修正线性单元函数 (Rectified Linear Unit)
CE	交叉熵损失函数 (Cross Entropy)
ONNX	开放式神经网络交换 (Open Neural Network Exchange)
HTML	超文本标记语言 (Hypertext Markup Language)
CSS	层叠样式表 (Cascading Style Sheets)
DOM	文档对象模型 (Document Object Model)
ATSS	自适应样本选择 (Adaptive Training Sample Selection)
PAA	概率标签分配 (Probabilistic Anchor Assignment)
NMS	非极大值抑制 (Non-Maximum Supresion)
OTA	最优输运标签分配 (Optimal Transport Assignment)
MLP	多层感知机 (Multi-Layer Perceptron)
MSA	多层自注意 (Multi-head Self-Attention)
SA	单头自注意 (Self Attention)
MAP	平均精度均值 (Mean Average Precision)
SGD	随机梯度下降 (Stochastic Gradient Descent)
BMCD	骨髓血细胞数据集 (Bone Marrow Cell Dataset)
ATSS	自适应样本选择 (Adaptive Training Sample Selection)
LN	层标准化 (Layer Normalization)
Faster-RCNN	快速区域卷积神经网络
	矩阵行列式或绝对值

## 第1章 绪论

### 1.1 研究背景与意义

骨髓是人体最主要的造血器官，其存在于人体骨骼内部的空腔中，约占体重的 3.5~5.9%。作为人体的造血组织，骨髓中包含了多种不同发育阶段的血细胞，这些血细胞按照形态与功能可以划分为粒细胞系、红细胞系、淋巴细胞系、单核细胞系与浆细胞系。骨髓血细胞成熟后会通过密质骨中的连通管进入人体外周血，参与人体循环系统的血液循环，保证机体新陈代谢的进行。

血细胞的质与量出现异常通常与某种血液疾病密切相关。白血病<sup>[1]</sup>是一种常见多发的血液疾病，主要表现为细胞异常克隆增生导致的骨髓造血功能异常。白血病属于人体造血系统的恶性肿瘤，在所有恶性肿瘤中占比约 5%，是我国重点防治的十大恶性肿瘤之一。白血病患者临床症状为贫血、出血、发热、乏力等，其致死率较高，早期发现与治疗对延长患者生存时间、改善患者生活质量至关重要<sup>[2]</sup>。

骨髓血细胞形态学检查是精确诊断白血病类型的关键手段之一<sup>[3]</sup>。目前，大型医院的骨髓血检查主要依靠病理学医师对显微设备采集的血细胞图像进行观察，统计不同类型的血细胞数量，然后根据 FAB 分型标准<sup>[4]</sup>对白血病类型进行初步诊断。检测流程如下：首先，需制备骨髓涂片并使用瑞特与吉氏混合液进行染色。接着，使用低倍显微镜判断骨髓增生程度，观察是否存在异常形态的特殊细胞。在低倍镜观察完全片后，再使用油镜从骨髓涂片中部向尾部移动，记录约 500 个有核细胞中各类血细胞的数量。目前人工镜检存在以下不足，人工分类计数过程非常枯燥且繁琐费时，通常需要数个工作日后才能出具诊断报告，不能满足快速临床诊疗的需求。对医师的专业技术要求较高，培养精通细胞病理诊断的医师周期长，年轻医师从事人工镜检的意愿低。诊断结果依赖于医师的专业知识与经验，存在较强的主观性，诊断的规范性与一致性较差。

在过去的 20 年间，计算机科学技术高速发展，医疗硬件设备的不断提升，医学领域积累了大量的医疗诊断数据，人工智能（AI, Artificial Intelligence）技术被广泛应用于医学领域<sup>[5-6]</sup>。目前 AI 已经在医疗机器人、药物研发、智能问诊、智能影像识别等领域落地与应用。AI 高效的计算与分析能力极大提升了医生的工作效率，为疾病检测与诊疗带来了深刻的变革。在血细胞图像智能诊疗方面，诸多研究学者采用深度学习的方法来自动定位与识别血细胞，实现了快速筛选和分类计数。这项技术使得细胞形态学诊断变的自动化、标准化与智能化，将医生从繁重的细胞病理工作中解放出来，具有非常重要的临床辅助诊断的意义。

目前骨髓血细胞自动化检测与识别技术已取得了长足的进步，但仍然面临着诸多挑战。在血细胞检测方面，涂片背景复杂、干扰较多、细胞间相互黏连与重叠会影响检测结果的精确度。在血细胞识别方面，骨髓血细胞种类非常多，存在各类细胞样本数量不均衡、细胞类内差异大、相邻发育阶段细胞类间差异小等问题。因此，基于深度学习的血细胞自动检测与识别方法仍有巨大的提升空间。本文针对骨髓血细胞检测与识别关键问题进行研究，并编写相关软件，为医生的临床诊断提供参考依据，具有非常重要的临床意义与广阔的应用前景。

## 1.2 研究现状与进展

本节介绍骨髓血细胞检测算法与骨髓血细胞识别算法相关研究现状。

### 1.2.1 骨髓血细胞图像检测现状

在血细胞涂片图像处理的过程中，包含血细胞区域的提取至关重要，检测与分割结果的精确性对后续识别任务有很大的影响。如何精准的从血细胞涂片中分割出各类细胞的边界，定位包含血细胞的区域是医学图像处理的重要研究方向之一。近几十年来，国内外学者对此进行了深入的研究，并提出了多种解决方案，主要可以划分为以下四类，基于阈值的检测方法<sup>[7-9]</sup>、基于边缘检测的方法<sup>[10-11]</sup>、基于聚类的检测方法<sup>[12-13]</sup>、基于深度学习的检测方法。

基于阈值分割方法是一种广泛使用的图像分割技术，其基本思想是在选定的颜色空间中根据某种规则选取一组阈值，从而将图像分割为不同的区域。常见的阈值选取方法有大津法(OTSU)、分水岭和区域增长方法等。Cseke<sup>[7]</sup>基于OTSU方法，通过最大化不同色彩区域的类间方差得到分割阈值，实现对细胞核，细胞浆与背景分割，但该方法对细胞浆的分割结果不尽人意。Jiang<sup>[8]</sup>结合尺度空间滤波与分水岭算法实现对细胞核与细胞浆的分割，该分割方法首先利用尺度空间滤波从图像中提取出细胞核，然后对三维HSV直方图进行分水岭聚类分割出细胞浆，该方法在背景复杂或有噪声的情况下存在过度分割的情况。Wu<sup>[9]</sup>等利用HIS颜色空间H分量与S分量开发了一种基于圆形直方图的迭代OTSU白细胞分割方法，该方法在彩色涂片图像上获得了较好的分割结果。

基于边缘检测的方法是找到图像中变化剧烈的像素点集合，这些点通常是目标的轮廓点。边缘检测借助于表征边缘梯度的边缘算子实现。常用的边缘算子有Sobel边缘算子、Canny边缘算子等。在血细胞图像中，基于边缘检测的方法对于染色效果好、细胞间无黏连无重叠的区域的分割效果较好，但在染色欠佳，边界复杂的情况下，很难获得理想的分割结果，该方法通常会结合其他方法来提升图

像分割的精度。马建林<sup>[10]</sup>等提出了一种基于边缘检测的区域增长分割算法，该方法首先利用改进的 Canny 算子进行边缘粗检测，再利用给出的图像灰度值和纹理、颜色等信息进行区域合并，实验结果表明该方法对于医学图像中的复杂区域与畸形区域具有较强的鲁棒性与实用性。Sadeghian<sup>[11]</sup>等提出了一种基于边缘检测的主动轮廓分割方法，首先采用 Canny 算子提取初始的边界，接着利用 GVF snake 算法以初始边界不断迭代提升轮廓分割的精度。

基于聚类的方法是根据某种相似性规则，例如纹理、灰度、颜色等信息将图像中的像素划分为多类，从而实现图像分割。Theera-Umporn<sup>[12]</sup>基于模糊 C 均值 (FCM) 将图像分割为若干小区域，之后计算各个类中心与其他区域中心的关系，之后将小区域进行合并得到最终的分割结果。模糊 C 均值的参数依赖于经验值，对于背景复杂，染色不均的图像难以获得精确的分割结果。Ramoser<sup>[13]</sup>使用 k-means 方法在 HSL 颜色空间将图像分割为细胞核、背景、细胞浆-红细胞这三个部分，之后根据初步分割的结果构造白细胞似然图像，接着应用 MSER 方法计算分割阈值得到白细胞分割图像。

自 2012 年，基于深度神经网络的方法在图像分割、图像识别、目标检测等领域取得了巨大的突破和广泛的应用。现代的医学图像检测、分割任务几乎都是基于深度学习方法。基于深度学习目标检测可以分类两个流派，两阶段检测与单阶段检测方法。前者包含一个从粗糙到细致的筛选过程，而后者只需要一步即可完成目标的定位与分类。在两阶段检测器方面，2014 年 R. Girshick<sup>[14]</sup>提出了 RCNN 模型，首次将 CNN 网络应用到目标检测领域。针对 RCNN 中存在的重叠区域重复计算、图像缩放导致几何形变等问题，何凯明<sup>[15]</sup>、R. Girshick<sup>[16]</sup>分别提出 SPP-Net 与 Fast-RCNN 来提高目标检测的运算速度。2015 年，任少卿<sup>[17]</sup>提出了 Faster-RCNN 网络，其采用 RPN(Region Proposal Network) 网络替代之前的区域推荐方法，从而极大提高了目标检测的速度与精度。单阶段检测器的里程碑是由 R. Joseph<sup>[18]</sup>等在 2015 年提出的 YOLO(you only look once) 网络，其直接使用单个神经网络用于完整的图像检测，摒弃了双阶段检测器推荐区域与进一步坐标回归和分类的范式。Lin 等<sup>[19]</sup>指出密集检测器在训练期间遇到的前景-背景类别极度不均衡是主要原因，并在此基础上引入了 Focal loss，通过重定义标准交叉熵损失函数使得检测器可以将更多的注意力放在困难样本的学习上，进一步提升了单阶段检测器的检测精度。最近基于 Transformer<sup>[20]</sup>的模型也被用于目标检测领域，主要可以归纳为以下三种范式，使用 Transformer 骨干网络替换双阶段目标检测器中的骨干网络来提取图像特征；使用 CNN 作为骨干网络提取特征，并将目标检测视为集合预测问题，通过 Transformer 编码器解码器结构直接输出一组目标的位置与类别信息，代

表网络有 DETR<sup>[21]</sup>; 纯粹基于 Transformer 的端到端目标检测网络, 如 YOLOS<sup>[22]</sup>、PVT<sup>[23]</sup>等。基于深度学习的检测网络在通用目标检测数据集 (COCO、ImageNet 等) 上性能优越。诸多研究学者对上述网络进行改进以适用于血细胞涂片图像的目标检测。Xia<sup>[24]</sup>等使用 Faster-RCNN 网络进行血细胞检测, 检测的准确率达到了 98.4%。Dhieb<sup>[25]</sup>等使用 Mask-RCNN 网络对红细胞与白细胞进行检测与识别, 该方法将 COCO 数据集上预训练的 ResNet-101 网络作为主干网络, 并使用 FPN 网络来提取多尺度的特征用于不同尺寸的细胞检测。由于训练样本较少, 作者采用了多种数据增强方式防止过拟合。该模型对红细胞与白细胞检测准确率分别为 92% 与 96%, 并且可以有效的识别重叠和染色欠佳的细胞。Shakarami<sup>[26]</sup>等基于 YOLOv3 单阶段目标检测网络提出了 FED(Fast and Efficient YOLOv3) 模型在三个尺度上对血细胞进行检测, 其使用 EfficientNet 替换 Dark-Net53 作为主干网络, 并应用空洞卷积增加网络的感受野、深度可分离卷积来减小模型的参数量, 网络在 BCCD 数据集上对血小板、红细胞、白细胞的平均识别准确率分别为 90.25%、80.41%、98.92%。

此外, 一些研究学者使用基于深度学习的语义分割网络将血细胞图像中的细胞分离出来。Ronneberger<sup>[27]</sup>等提出了 U-Net 网络模型, 相比于 FCN 增加了编码器-解码器结构, 编码器负责特征提取, 解码器将提取的特征进行融合并恢复到原图的尺度。U-Net 网络使用了跳跃连接将编码器低层级的特征与解码器高层级特征进行拼接从而保留了目标的细节信息, 该网络在医学图像分割中获得了良好的性能。Lu 等<sup>[28]</sup>基于 U-Net++ 和 Resnet 提出了 WBC-Net 用于血细胞的分割, WBC-Net 设计了一个带有残差模块的特征编码器来提出多尺度特征, 并在密集卷积模块上引入混合跳跃连接来融合不同语义的特征图。WBC-Net 使用基于交叉熵和 Tversky 指数损失函数来训练网络, 并获得了 98.97% 的分割准确率。

### 1.2.2 骨髓血细胞图像识别现状

自动血细胞分类技术按照原理可以分为三类, 分别是物理方法、物理-化学方法与图像分析方法<sup>[29]</sup>。

物理方法包括了电学与光学方法, 其中应用最广泛的是体积-电导-激光散射分析方法 (Volume-Conductivity-Scatter)。该方法中体积通过库尔特理论得到, 即包含细胞的电解液通过细小管道时, 会导致管道两侧的电阻发生改变, 通过这个电信号来确定细胞体积的大小。电导性通过采用高频的探针来探测细胞内部复杂结构, 进而区分杆状核和分叶核。激光散射根据散射信号的差异来判断细胞质内的颗粒类别, 进而区分嗜中性、嗜酸性与嗜碱性细胞。体积-电导-激光检测技术通过对血细胞体积、细胞核与细胞质颗粒进行分析实现了白细胞精准的五分类。但是仅基

于物理的方法无法得到血细胞的形态信息，并且结果容易受到血小板凝集、难溶红细胞等因素的干扰。

物理-化学方法是将细胞化学染色与激光散射相结合的技术。化学染色方法有核酸荧光染色、过氧化酶染色等。在细胞染色后进行激光照射，不同角度的散射光包含了细胞的结构信息，从而实现对血细胞的分类，但该方法同样不能得到直观的细胞形态学信息。

图像分析方法，骨髓血涂片在经过染色后，不同类型血细胞胞体形状各异且细胞核与细胞浆会呈现出不同的颜色与纹理特征。基于数字图像处理的白细胞分类方法包含了图像采集、图像分割与图像识别这三个部分。图像采集通常由自动化采集设备完成，首先采用 $\times 10$ 倍物镜找到只包含单层细胞的区域，接下来转换到 $\times 100$ 倍油镜扫描拍摄该区域得到细胞涂片图像。图像分割如上节所述，目的是从涂片中定位包含血细胞的区域，得到只包含单个血细胞的切片图像。图像识别对得到的单个血细胞切片进行分类，目前主要有两大类方法，基于传统模式识别的分类识别方法和基于深度学习的分类识别方法。

基于传统模式识别<sup>[30]</sup>的血细胞识别方法主要包含特征提取、特征选择与分类器推理预测这三个部分。特征提取是在原始图像中提取出具有区分性的特征，如血细胞的几何形态、色度、纹理、统计等特征。特征选择对提取到的特征进行可分类能力的评估，从中筛选出最典型的显著特征，剔除无关冗余的特征，提升模型的识别速度与精度，常用的特征选择方法有过滤法、包装法、嵌入法、降维法等。分类器对输入样本的特征集合进行类别预测，分类器的种类非常多，例如朴素贝叶斯、支持向量机、决策树、随机森林与K近邻等。Ghosh<sup>[31]</sup>等提取了血细胞的面积、周长、圆度、浆核比等九个几何形态特征，并利用t检验方法对特征进行筛选，最终四个显著特征被输入到朴素贝叶斯分类器中，对五类白细胞的分类准确率为83.2%。Rezatofighi<sup>[32]</sup>等提取了形态特征，并基于灰度共生矩阵与局部二值模式提取了纹理特征，然后采用序列前向选择方法(SFS)对特征进行选择，最后比较了人工神经网络与支持向量机两种分类器的性能。孙凯<sup>[33]</sup>等提取了几何、纹理、小波三部分共63个特征，在PCA降维后得到了八个主成分，接着使用了支持向量机、多层感知机、决策树对其进行分类，最好的分类结果准确率为88.6%。袁满<sup>[34]</sup>对细胞核与细胞浆提取了颜色、纹理形态共100个特征，并对提取的特征使用z-score进行标准化，接着基于Fisher准则选择其中的70个特征，最后使用支持向量机、随机森林和K近邻方法对白细胞进行五分类。

基于传统的机器学习方法依赖专家经验人工设计的特征，无法捕捉高层次的抽象隐含的特征，并且需要进行特征选择来筛选显著性特征，对于大量的数据样

本可能存在模型欠拟合、泛化性能差等缺点，基于深度学习的方法可以实现特征工程的自动化，目前也是血细胞分类识别的主流研究方向。Matek<sup>[35]</sup>等制作了一个包含 18375 张总计 15 类的骨髓血细胞图像数据集，针对类别数量不平衡的问题，采用随机 0°~360° 的旋转变换，与水平翻转垂直扩充训练数据集，接着采用通用的 ResNext 网络进行分类，网络对于常见的骨髓血细胞，例如中性粒细胞、典型淋巴细胞、单核细胞等达到了 94% 的识别准确率，该网络没有针对血细胞图像进行改进，整体的分类准确率不是很理想。Mori<sup>[36]</sup>将骨髓血细胞按发育不良（细胞质颗粒减少）的程度划分为四类，使用 Resnet-152 网络对血细胞进行分类，平均灵敏度与特异性分别为 85.2%、98.9%。2020 年，杭州智微科技<sup>[37]</sup>联合陆军军医大学第二附属医院基于 2018 年到 2019 年间收集的 65986 幅真实病例血细胞图片开发出了一个完整的自动化检测识别系统 morphogo，该系统可以实现血细胞涂片的自动采集，检测识别与结果的可视化。该研究使用了 27 层的卷积神经网络，对于 12 类骨髓血细胞的分类准确率均超过了 85.7%，但是部分类别的召回率只有 40%，网络的特征表达能力有待提升。目前市场上的全自动血细胞形态学分析仪比较少，另外一款产品是瑞典 Cellavision 公司研发 DM 96<sup>[38]</sup>全自动血细胞形态学分析系统，也用于血细胞的预分类辅助医生诊断。Huang<sup>[39]</sup>等首先基于 RetinaNet 检测网络得到只包含单个血细胞的切片图像，接着将自适应注意力模块引入到含残差模块的卷积神经网络中，注意力模块是一个先下采样再上采样的卷积模块，该模块增强了与分类任务相关的区域特征的权重，提升了模型的表达能力，该网络针对六类白细胞实现了 95.3% 的平均分类准确率。

### 1.3 本文研究内容

本文研究所使用的血细胞病理图由邃蓝智能科技（上海）有限公司提供，研究目标是基于深度学习的方法实现骨髓血细胞自动化检测识别任务，并结合图像采集设备硬件，编写骨髓血细胞自动化检测与识别的软件。软件可将骨髓涂片中血细胞的位置与预分类结果呈现给医生，供其参考与核对，从而辅助医生临床诊断。本文涉及的内容主要包含基于深度学习的骨髓血细胞检测、骨髓血细胞识别与系统软件解决方案这三个部分。

1) 骨髓血细胞检测，本文对比了基于深度学习的单双阶段检测器的速度与精度等性能，并选择在血细胞领域较为先进的 RetinaNet 作为基线模型。针对密集血细胞区域，血细胞边界框检测偏差大、黏连细胞边界检测错误等问题，本文在 RetinaNet 的 FPN 层后引入了自底向上的路径聚合模块，该模块将更浅层的特征与 FPN 的深层特征进行融合，缩短了底层与顶层特征之间的信息传递路径，使得浅层

的纹理、边缘等高分辨定位信息可以更容易传递到顶层，提升了网络定位特征的表达能力。此外，我们探究了不同标签分配策略对于检测精度的影响，并采用基于最优输运的标签分配策略，通过全局最优来解决模糊 anchor 的分配问题，进一步提升网络的召回能力。最终结合了上述改进方法的检测模型，相较于原有 RetinaNet 基线模型检测精度有较大提升。

2) 骨髓血细胞识别，目前基于深度学习的血细胞识别研究工作主要使用通用的目标分类网络，并未针对血细胞的特性进行改进。此外，很多研究只关注了血细胞大类，未关注其中的子类别，例如粒细胞的原始、早幼、中幼与晚幼等阶段，血细胞子类之间差异较小，使得其自动化识别更具挑战性。针对上述难点，本文将性能优异的 Vision Transformer 作为基线模型，提出了重叠图像块划分方法、辨识性区域选择模块与对比 loss 对基线模型进行改进。其中重叠图像块划分方法可以更好的保留图像的局部信息、避免破坏图像的局部结构。辨识性区域选择模块采用压缩激发结构学习所有编码层的注意力权重，并将每个注意力头最大权重对应的隐含特征用于分类，该模块让网络关注到不同类别之间的细微差异部分，同时舍弃大量区分度较低的背景、超类共同特征区域，从而提升网络的细粒度特征表达能力。为了进一步增加分类特征的类内一致性与类间差异性。本文引入了对比损失，对特征的间隔进行约束，使得不同标签特征相似度最小，相同标签的特征相似度最大。本文方法相较于其他卷积神经网络与基线 Vision Transformer 模型分类准确率有较大提升。

3) 骨髓血细胞检测识别系统软件研发，本文通过主动学习的技术完成了一万余张骨髓血细胞图像边界框与类别信息的标注，并完成上述相关检测与识别算法训练与优化。本文对骨髓血细胞检测与识别软件进行需求分析，确定软件架构为 B/S 架构，并对数据库各个表单进行详细设计。软件前端使用 VUE+ElementUI 框架进行页面设计与交互。后端使用的框架为 Python+Django。软件包括了用户登录/注册、骨髓血细胞检测、骨髓血细胞识别、患者信息核查与管理、患者数据分析等五大功能模块。

## 1.4 论文组织结构

图为本文的组织结构框架，总共分为七个章节，各个章节的详细内容安排如下：

第一章为绪论。首先介绍了骨髓血细胞形态学检测的背景，并阐述了本文基于深度学习的骨髓血细胞自动化检测与识别的意义。接着，介绍了骨髓血细胞检测与识别的国内外研究现状并分析各个研究的优势与不足。然后简要说明了本文

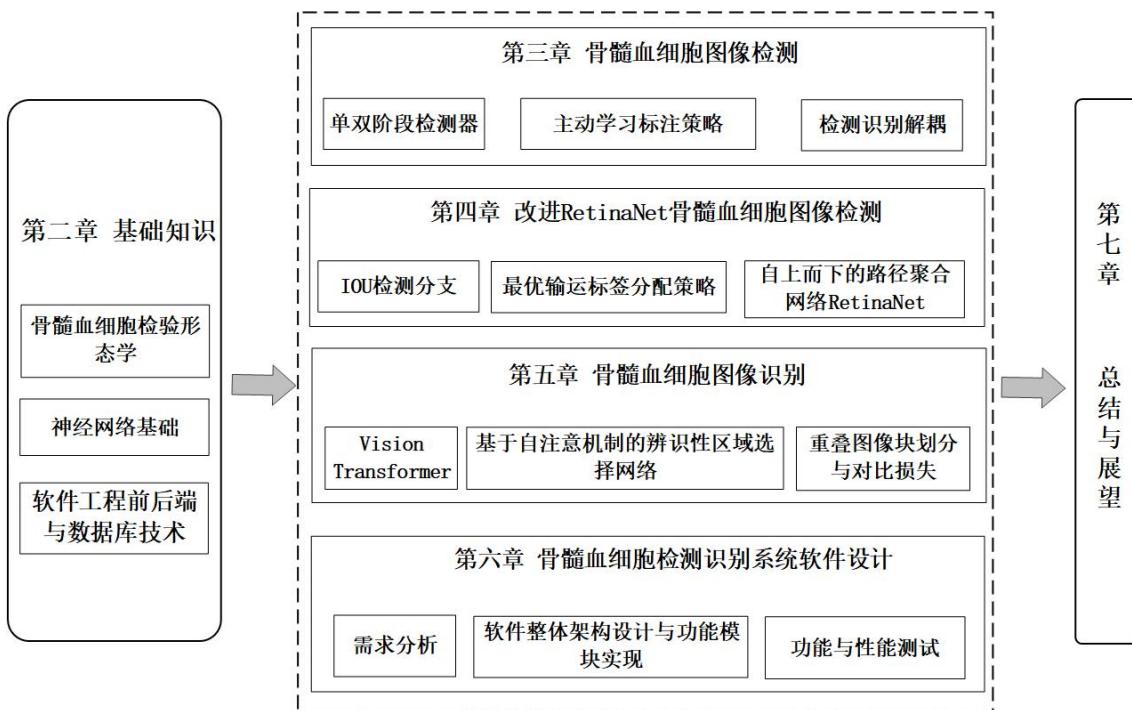


图 1.1 论文组织结构

主要研究内容。最后给出本文的章节组织。

第二章介绍本文的基础知识与技术。本章首先阐述骨髓血细胞相关病理学知识，对比了不同类别骨髓血细胞的形态学差异，介绍骨髓血细胞数据集标注与数据增强方法，并给出了交叉训练集与测试集的划分。接着，概述神经网络的基本理论与相关检测与识别技术。最后，介绍了软件开发使用的前端、后端与数据库技术。

第三章研究骨髓血细胞检测相关问题。本章对比了多种单阶段与双阶段目标检测网络的检测精度、计算量与速度等性能，确定将 RetinaNet 作为骨髓血细胞检测的基线模型。然后，本章探索了 RetinaNet 网络在仅检测任务与检测识别一体化任务上特征提取与识别准确率上的差异，确定了先检测再识别的骨髓血细胞处理流程。

第四章研究基于改进 RetinaNet 的骨髓血细胞检测网络。针对漏检、误检等问题，在网络结构方面引入了路径聚合网络，缩短底层与顶层特征的信息传递路径，提高网络对高分辨定位特征的提取能力，减小定位误差。此外引入了 IOU 预测分支，将检测框的定位质量也纳入到候选框的筛选中。本节对检测网络的标签分配策略进行研究，提出了一种基于最优运输的标签分配策略，提升网络对于血细胞的召回能力与检测精度。

第五章研究骨髓血细胞识别问题。分析了近年来多种基于深度学习的识别网络，提出了一种基于改进 Vision Transformer 的骨髓血细胞识别模型。该网络由多

个堆叠的自注意编码层组成。为了充分利用自注意力机制，本文采用压缩激发模块学习了多个编码层的注意力权重，该模块可以有效捕捉到不同细胞之间细微的差异部分，提高网络的细粒度特征表达能力。在训练过程中，将对比损失与交叉熵损失函数进行有机结合，进一步提升网络提取特征的辨识性，该网络模型在 TMAMD 骨髓血细胞数据集取得了最佳性能。

第六章介绍骨髓血细胞检测与识别系统软件的设计与实现。本节首先对骨髓血细胞检测识别系统进行需求分析，对软件的开发环境与平台进行说明，并设计了软件的整体架构。接着详细介绍了数据库表与各个功能模块的设计与实现。最后对软件进行了功能与性能测试，总结了软件的测试结果。

第七章为总结与展望。总结了本文的工作内容，对骨髓血细胞检测与识别的发展进行展望。

## 第2章 基础知识

本章主要介绍本文所涉及的基础理论知识，首先概述了骨髓血细胞不同类别与发育阶段形态学特征。然后介绍了骨髓血细胞图像数字化与数据集的构建。接着本文介绍了神经网络相关概念，最后对骨髓血细胞检测与识别软件系统涉及到的相关框架进行了介绍。

### 2.1 骨髓血细胞图像及预处理

#### 2.1.1 骨髓血细胞形态学介绍

人体血细胞主要由骨髓内的造血干细胞分化而成。造血干细胞由髓系干细胞与淋系干细胞构成。其中髓系干细胞分化为粒细胞系统、红细胞系统、单核细胞系统与巨核细胞系统，淋系干细胞分化为浆细胞系统与淋巴细胞系统。不同系统的细胞按照发育成熟过程可以分为原始、早幼与成熟这三个阶段。粒细胞与红细胞的幼稚阶段可再具体划分为早幼、中幼与晚幼这三个发育阶段。粒细胞系统根据细胞质内特殊颗粒对酸碱性物质亲和性，可分为嗜酸性粒细胞、嗜碱性粒细胞和中性粒细胞。骨髓血细胞六大系统血细胞的发育过程如图 2.1 所示<sup>[40]</sup>：

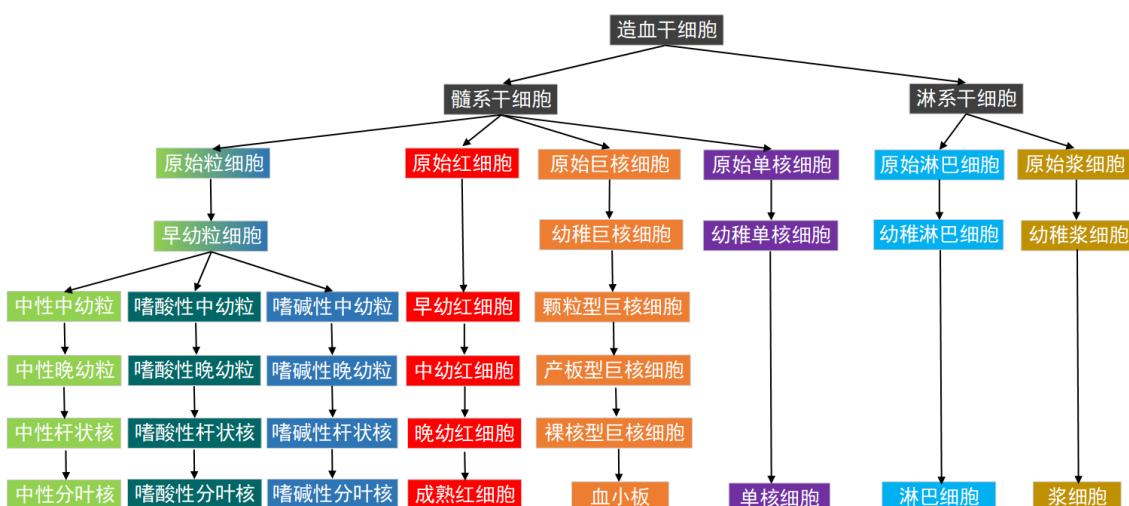


图 2.1 骨髓血细胞发育成熟过程示意图

不同类别的骨髓血细胞体形状各异且细胞核与细胞浆会呈现出不同的颜色与纹理特征。表 2.1 对本文主要关注的骨髓血细胞类别进行细胞核、细胞质等形态学方面的简要介绍。

表 2.1 骨髓血细胞形态学特征

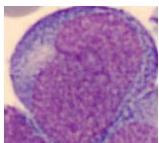
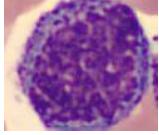
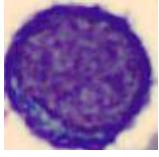
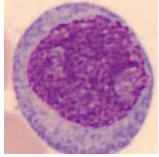
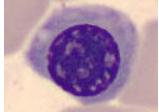
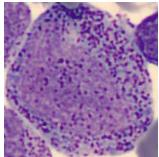
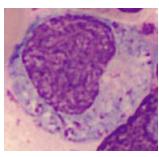
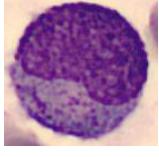
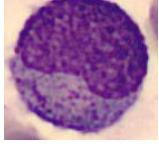
细胞名称	图像示例	胞体特征	细胞核	细胞质
原始细胞		类圆形，直径10~20微米	居中，呈圆形，染色质为颗粒状，具有多个小而清晰的核仁	细胞质较少，无颗粒，呈蓝色或深蓝色
单核细胞		圆形或椭圆形，直径在14~25微米	扭曲折叠，常位于胞体中央或一侧，染色质疏松，核仁消失	通常为浅灰蓝色，可见空泡与紫红色的粉尘样颗粒
淋巴细胞		类圆形或不规则，直径在12~15微米	染色质致密，呈现索块状，形态上存在凹陷或者切迹	细胞质极少，呈现淡蓝色，无颗粒。
浆细胞		椭圆形或不规则，直径在12~16微米	多偏位，染色质聚集	细胞质为不透明的深蓝色，在细胞核周有淡染色带
有核红细胞		规则类圆形，直径在7~10微米	圆形位于细胞中央，内部含多个紫黑色团块	细胞质较多，无颗粒，为淡红色。
早幼粒细胞		较大，圆形或椭圆形，直径在12~25微米	核较大，内部染色质细致，有清晰可见的核仁	细胞质为深蓝色，含有分布不均、形态不一的非特异性颗粒
中性中幼粒细胞		类圆形，直径在10~20微米	半圆形或微凹陷，无核仁，染色质密集索块状	细胞质呈淡蓝色，其中存在大小均匀，密集的淡粉红色中性颗粒
中性晚幼粒细胞		胞体类圆形，直径在10~16微米	呈半月形，存在凹陷，凹陷程度小于直径的1/2，染色质聚集小块状	细胞质多，淡蓝色，存在较多中性颗粒
中性杆状核细胞		胞体类圆形，直径在10~16微米	呈半月形，存在凹陷，凹陷程度小于直径的1/2，染色质聚集小块状	细胞质多，淡蓝色，存在较多中性颗粒

表 2.1 -接上表

细胞名称	图像示例	胞体特征	细胞核	细胞质
中性分叶核细胞		胞体类圆形，直径在 10~16 微米	呈半月形，存在凹陷，凹陷程度小于直径的 1/2，染色质聚集小块状	细胞质多，淡蓝色，存在较多中性颗粒
嗜酸性粒细胞		直径 15~20 微米。	类似中性粒细胞，染色质聚集索块	大小分布均一，橘红色的嗜酸性颗粒
嗜碱性粒细胞		直径 10~15 微米。	染色质细致	颗粒粗大，大小形态不一深紫红色的颗粒，部分颗粒覆盖在细胞核上

## 2.1.2 骨髓血细胞数据集与预处理

### 1) 骨髓血细胞切片与数字化

骨髓血细胞形态学检验技术是临床诊断血液疾病的重要依据。该过程由取材、制片、固定、染色、洗涤、干燥与镜检等流程组成。首先通过骨髓穿刺采集骨髓液样本。接着将骨髓液置于载玻片上制作成薄厚均一的涂片。在涂片制作完成后，使用甲醛或乙醇溶液将其固定。然后采用瑞特-吉姆萨染色剂进行适当时间的着色，染色完成后使用清水冲洗，待自然风干后以备观察。最后使用光学显微镜对骨髓涂片进行观察，统计骨髓中不同类型血细胞的数量、比例、形态以及是否存在异常细胞，进行骨髓评估判断病情，具体如图 2.2 所示。

对骨髓血细胞切片图像进行自动化评估分析首先需要对切片图像进行数字化<sup>[41]</sup>，从而可以在计算机上进行存储、传输与分析。数字化技术可以为医学研究提供大量数据源，促进医学研究与临床实验，方便医生进行更加快速与精准的病情分析。病理图像数字化技术由以下几个步骤组成。1) 数字扫描：对制备好的组织切片使用数字成像与扫描设备生成数字图像，一般通过 CCD 相机采用对焦深度法自动获取对焦位置并扫描成像。2) 数字化处理：对图像进行去噪、分割与边缘检测等进一步提升图像质量。3) 云存储与管理：将数字化图像存储在云端服务器中，保证数据的完整性与安全。4) 数字分析与诊断：利用计算机视觉等深度学习图像分析技术，对数字化后的病理图像进行分析与诊断，提高病理诊断的效率与准确性。

### 2) 数据集标注

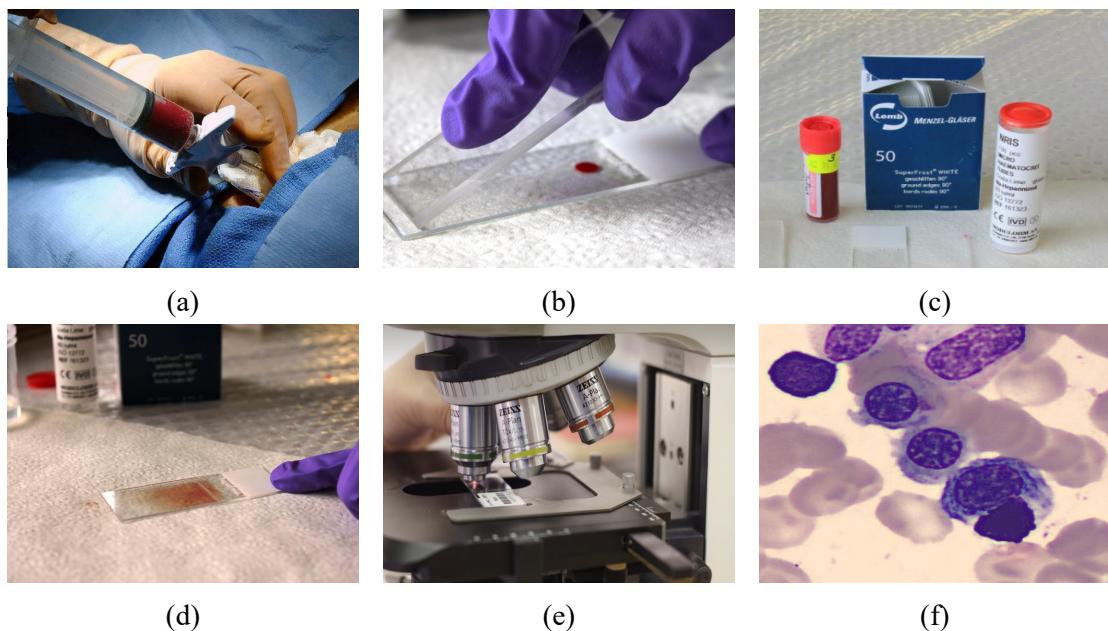


图 2.2 骨髓血细胞检验技术流程: (a) 骨髓穿刺抽取骨髓液, (b) 制备骨髓涂片, (c) 瑞特-吉姆萨染色, (d) 清洗晾干, (e) 人工镜检判读, (f) 骨髓形态学图片

本文使用数据来源于实践基地邃蓝智能科技有限公司合作医院提供的脱敏数据。数据标注是深度学习模型训练的基础，并且直接影响模型的性能和泛化能力。我们在合作医院病理医师的协作下对血细胞的边界框与类别信息进行精准的标注，完成了骨髓血细胞数据集（BMCD, Bone Marrow Cell Dataset）的制作。原始数据集总共包含 9250 张骨髓血细胞图像，我们只关注图像中的有核细胞，而将成熟红细胞作为背景。因为缺少相关专业知识，我们仅标记出血细胞边界框的位置。我们使用 labelme 软件作为标注工具，如图 2.3 所示，针对每一张图像生成 json 标注文件，记录血细胞边界框的左上角与右下角的坐标，最后再将标注转化为 coco 格式，并将数据集划分为训练集、测试集与验证集。经过检测网络，我们得到了单一血细胞图像，即图像中仅有一个完整的骨髓血细胞，这些图像再由经验丰富的病理医生完成类别标签的标注。

大量的数据标注需要消耗很高的人力、物力成本，我们采用主动学习<sup>[42]</sup>技术去发掘数据集中高信息量的样本，提高标注的效率与精度，降低标注成本。主动学习的基本思想是标注少量部分数据，利用已标注的数据训练深度学习模型。然后使用模型对未标注的数据进行预测，根据最大化熵、不确定度采样等进行排序，筛选出不确定性最高的一些数据优先进行标注。不断迭代上述过程，直到标注与模型性能达到预期。具体而言，对于血细胞检测任务，首先标注部分图像，然后训练 Faster-RCNN 网络得到一个初步的检测模型，然后对每张图像进行检测，并生成标注 json 文件，再反馈到 labelme 标注软件中进行微调。针对血细胞识别任务，

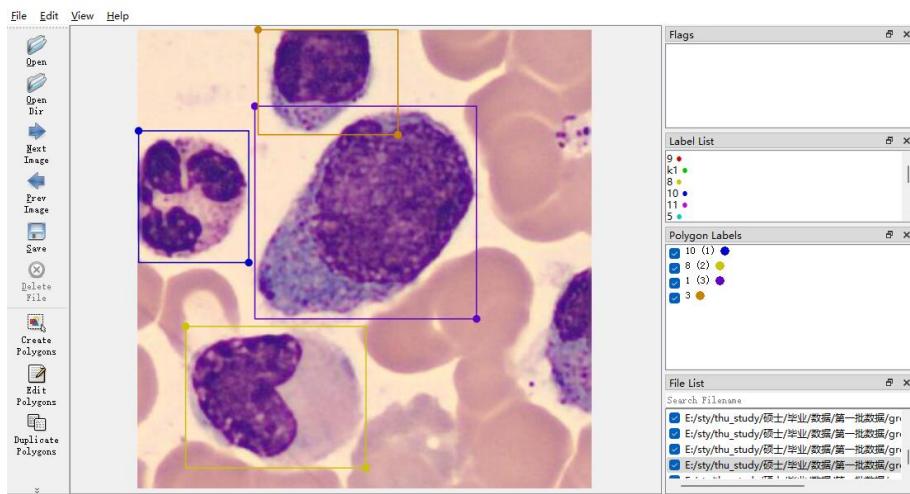


图 2.3 labelme 标注软件工具

经过检测网络，我们得到了多张单个血细胞图像。首先完成部分类别标注，训练初步的识别网络。对未标注的血细胞筛选出熵最大的预测样本反馈给病理医生进行核对。主动学习流程如图 2.4 所示：

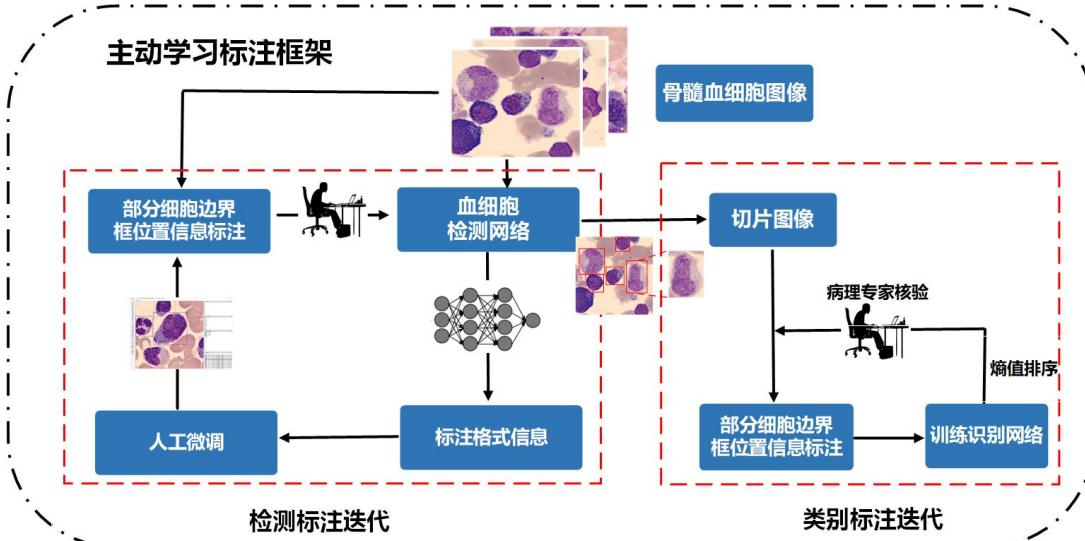


图 2.4 主动学习标注框架示意图

### 3) 数据增强

我们主要关注的五大系统细胞在人体内占比不同，粒细胞系统约占全部细胞的 50%，而浆细胞系统一般占比小于 1.5%，由于细胞天然就存在比例不均衡问题，这也导致了我们的数据集存在严重的类别不平衡问题。例如单核细胞数量仅为有核红细胞数量的 1/9。数据的不平衡会导致网络过多的关注数量较多的类别特征信息，数量较少类别易出现准确率与召回率的不均衡，影响模型的泛化性能。为解决上述问题，本文对于数量较多的类别采用随机欠采样减少样本数量。对于数量较少的类别，采用翻转、旋转、添加噪声与色彩调整增加数据多样性。下面对数

据增强<sup>[43]</sup>方法进行简要介绍：1) 翻转：将图像沿水平或垂直方向进行镜像对称，生成新的图像。当图像  $I(x, y)$  沿水平方向进行翻转后，水平翻转图像为  $I'(x, y) = I(w - x - 1, y)$ ，其中  $w$  为原始图像宽度。2) 旋转：将图像沿着某个点或轴进行旋转，当图像  $I(x, y)$  以  $(c_x, c_y)$  为旋转中心时，旋转角度为  $\theta$ ，旋转后的图像为  $I'(x, y) = I((x - c_x) \cos \theta - (y - c_y) \sin \theta + c_x, (x - c_x) \sin \theta + (y - c_y) \cos \theta + c_y)$ 。3) 添加噪声：常用的图像噪声包括高斯噪声、椒盐噪声与泊松噪声。高斯噪声是一种连续随机噪声，服从零均值、标准差为  $\sigma$  的正态分布，会使得图像变模糊。椒盐噪声是离散随机噪声，图像中会出现亮点与暗点，降低图像的清晰度与对比。4) 色彩调整：常用方法包括了亮度调整、对比度调整、色相调整与饱和度调整。

## 2.2 神经网络技术概述

### 2.2.1 神经元与梯度优化

#### 1) 神经元

神经网络发展历史最早可以追溯到 20 世纪五十年代，当时研究人员受到神经科学启发，模拟生物神经元结构提出了神经网络。神经元是神经网络的基本单元，其结构如图 2.5 所示，由输入、加权、激活函数、输出这四个部分组成。

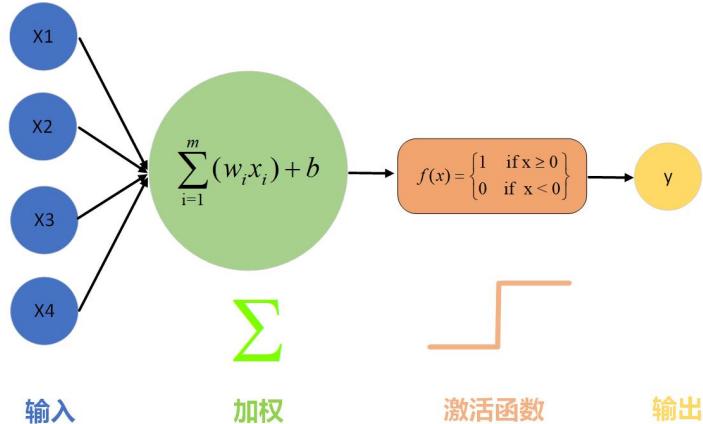


图 2.5 神经元结构示意图

输入部分，神经元接收其他神经元的信息  $x_1, x_2, \dots, x_m$ ，每个输入与权重进行关联。加权部分表示神经元对于输入部分的敏感程度。激活函数将加权和进行变换，保证输出数值范围并引入非线性的特性。输出部分是激活函数处理后的结果，如式 2.1 所示，可作为后续神经元的输入。

$$h(x) = f(\mathbf{W}^T \mathbf{x}) = f\left(\sum_{i=1}^m w_i x_i + b\right) \quad (2.1)$$

式 2.1 中  $f(\cdot)$  为激活函数。常用的激活函数有修正线性单元函数 (Rectified linear unit function, ReLU)、双曲正切函数、sigmoid 函数等。

### 2) 感知机

感知机由美国心理学家弗兰克·罗森布[44]在 1958 年提出，它是一种最简单的神经网络，拓扑结构如图 2.6 所示。网络包含了输入层、隐含层与输出层，前一层的每个神经元都与后一层的神经元相连。前一层神经元输出信息传递到下一层神经元的过程称为前向传播。该过程可以用一系列矩阵乘法与激活函数进行描述，若输入数据为  $x$ ，网络结构总共有  $L$  层，则前向传播过程可以表示为：

$$\begin{aligned} z^1 &= W^1 x + b^1 \\ a^1 &= g^1(z^1) \\ &\vdots \\ z^L &= W^L a^{L-1} + b^L \\ a^L &= g^L(z^L) = \hat{y} \end{aligned} \tag{2.2}$$

其中  $z^L$  表示第  $L$  层的未激活值， $a^L$  表示第  $L$  层的激活值， $W^L$  和  $b^L$  分别表示第  $L$  层的权重矩阵和偏置向量。

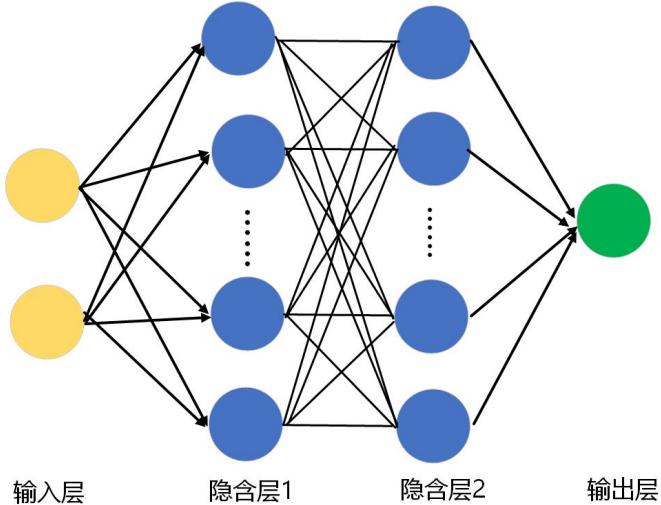


图 2.6 感知机网络模型示意图

### 3) 反向传播与梯度优化

反向传播全称是误差反向传播 (back propagation, BP)，是用于神经网络训练的算法。神经网络经过前向传播得到网络预测值，采用损失函数衡量预测值与真实值之间的误差。通过损失函数最小化计算神经网络参数的导数，然后调整更新神经网络权重参数以减小误差。不断迭代上述过程，直到模型参数收敛。

假设  $L$  为损失函数， $w_{ij}^k$  表示第  $k$  层第  $i$  个神经元到第  $j$  个神经元之间的权重，

$a_j^l$  表示  $l$  层的第  $j$  个神经元的非激活值  $z_j$  表示第  $j$  层神经元输出,  $\sigma_l(\cdot)$  为第  $l$  层激活函数。反向传播过程计算过程如下:

- (a) 网络进行前向传播得到预测值
- (b) 定义输出层的误差  $\delta_k^L$  为损失函数对输出层非激活值的导数, 如式 2.3 所示,

$$\delta_k^L = \frac{\partial L}{\partial a_k^L} = \frac{\partial L}{\partial z_k^L} \frac{\partial z_k^L}{\partial a_k^L} \quad (2.3)$$

- (c) 对于中间隐含层, 可以通过链式法则求出第  $l$  层的传播误差  $\delta_j^l$

$$\begin{aligned} \delta_j^l &= \frac{\partial L}{\partial a_j^l} = \frac{\partial L}{\partial z_j^l} \frac{\partial z_j^l}{\partial a_j^l} \\ \frac{\partial L}{\partial z_j^l} &= \sum_i \frac{\partial L}{\partial a_i^{l+1}} \frac{\partial a_i^{l+1}}{\partial z_j^l} = \sum_i \delta_i^{l+1} w_{ij}^{l+1} \\ \delta_j^l &= \frac{\partial L}{\partial a_j^l} = \sigma'_l(a_j^l) \sum_i w_{ij}^{l+1} \delta_i^{l+1} \end{aligned} \quad (2.4)$$

- (d) 计算损失函数对于某一隐含层  $l$  权重  $w_{ij}^k$  的梯度值

$$\frac{\partial L}{\partial w_{ij}^l} = \frac{\partial L}{\partial a_j^l} \frac{\partial a_j^l}{\partial w_{ij}^l} = \delta_j^l z_i^{l-1} \quad (2.5)$$

- (e) 若训练的学习率为  $\eta$ , 随机抽取小批量样本  $\{(\mathbf{x}_m, \mathbf{y}_m)\}_{m=1}^{N_0}$ , 权重值的更新如下

$$w_{ij}^{l(\tau+1)} = w_{ij}^{l(\tau)} - \eta \frac{1}{N_0} \sum_{m=1}^{N_0} \frac{\partial L_m}{\partial w_{ij}^l} \quad (2.6)$$

- (f) 重复上述步骤, 直到损失函数达到极小值, 得到收敛后最优的网络模型参数

#### 4) 优化算法

在定义好损失函数后, 通过误差反向传播算法调整网络中的权重参数, 使得网络在训练数据集上的误差最小化。但传统梯度优化算法面临着诸多困难, 损失函数存在着大量的鞍点与平坦区域, 可能收敛到局部极小值点。此外训练集数据量巨大, 计算全部数据梯度非常耗时。针对上述问题, 研究学者提出了小批量随机梯度下降算法 (mini-batch stochastic gradient descent), 每次从训练集中随机抽取  $m$  个样本组成小批量样本, 对于这组样本计算梯度均值用于更新权重系数, 如式 2.7 所示

$$\mathbf{g} \leftarrow \nabla_w \left( \frac{1}{m} \sum_{i=1}^m L(f(\mathbf{x}_i; \mathbf{w}), \mathbf{y}_i) \right) \quad (2.7)$$

$$\mathbf{w} \leftarrow \mathbf{w} - \eta \mathbf{g}$$

小批量随机梯度下降算法面临着学习率选择困难问题。学习率过小收敛慢，过大导致震荡。Adagrad<sup>[45]</sup>是最早的自适应学习率优化算法。其设置了一个梯度二阶累积统计量  $r$ ，每次使用小批量梯度平方和来更新  $r$ ，权重更新如下

$$\begin{aligned} r &\leftarrow r + g^2 \\ w &\leftarrow w - \frac{\eta}{\sqrt{r + \delta}} g \end{aligned} \quad (2.8)$$

Adagrad 算法在训练后期，由于累计平方很大，网络更新停止。针对上述问题，RMSProp 优化算法设置了一个衰减系数  $\rho$ ， $r$  的更新公式如下：

$$r \leftarrow \rho r + (1 - \rho)g^2 \quad (2.9)$$

Adam 优化算法 (Adaptive Moment estimation)<sup>[46]</sup>集成了动量与多种优化算法的优势，使梯度更新更平滑，适用于大多数的神经网络优化问题，是目前应用最广泛的梯度优化算法。该算法计算了梯度的一阶矩  $m$  与二阶矩  $v$ ，一阶矩类似于动量，降低梯度随机变化大的影响，二阶矩用于控制自适应学习率。参数更新公式如下：

$$\begin{aligned} m_t &\leftarrow \beta_1 m_{t-1} + (1 - \beta_1) g_t \\ v_t &\leftarrow \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \\ \hat{m}_t &\leftarrow \frac{m_t}{1 - \beta_1} \\ \hat{v}_t &\leftarrow \frac{v_t}{1 - \beta_2} \\ w &\leftarrow w - \frac{\eta}{\sqrt{\hat{v}_t + \delta}} \hat{m}_t \end{aligned} \quad (2.10)$$

## 2.2.2 卷积神经网络

### 1) 基本概念与结构

卷积神经网络 (Convolutional Neural Network, CNN)<sup>[47]</sup>是计算机视觉领域应用最广泛的神经网络。相比于传统的全连接神经网络，CNN 引入了卷积层与池化层，通过稀疏连接与权值共享，可以更加有效的提取空间数据的特征模式，降低了模型参数量与计算复杂度。卷积神经网络由多个卷积层、池化层、全连接层等组成，典型结构如图 2.7 所示。

### 2) 卷积层

卷积层是卷积神经网络的基础结构。卷积层由多个卷积核构成，每个卷积核用于抽取不同的图像特征，输出一个卷积通道特征。如图 2.8 所示，输入特征图有三个通道，该卷积层有四个卷积核，最终输出包含四个通道的特征图。多个卷积

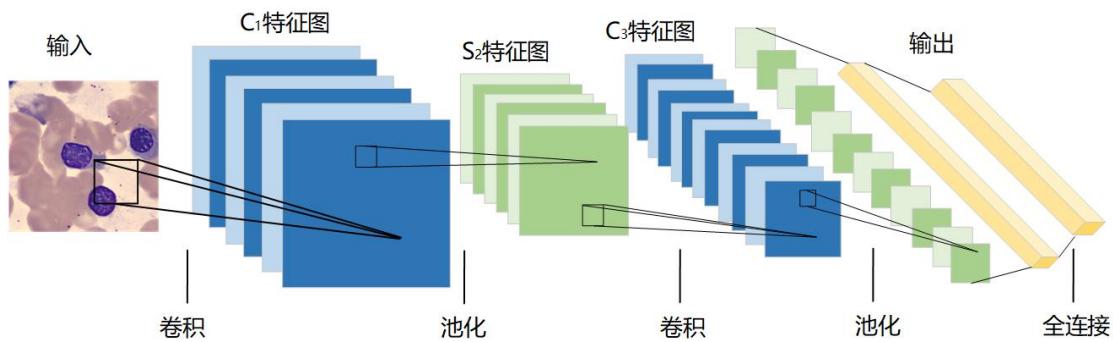


图 2.7 典型卷积神经网络结构示意图

通道拼接后组成卷积层的输出。卷积核将输入特征图局部区域像素与卷积核对应元素相乘后求和，再将结果写入到输出特征图对应位置，如式 2.11 所示。卷积核有四个描述参数，卷积核大小（kernel size）、步幅（stride）、边界扩充（padding）、输入与输出通道数量（channel）。通过卷积操作抽取图像特征，再不断将特征进行组合，形成最终图像的特征描述。

$$\begin{aligned} y(m, n) &= \sum_i \sum_j I(i, j) h(m - i, n - j) \\ &= \sum I(m - i, n - j) h(i, j) \end{aligned} \quad (2.11)$$

卷积核大小为卷积核的宽度与高度，通常小卷积核来提取边缘、线条等纹理特征，大卷积核提取高级抽象特征。步幅控制卷积核在图像上滑动的距离，决定了输出特征图的大小。步幅增大使得特征图大小减小，从而降低网络的计算复杂度。边界扩充是图像边界上进行零填充，可以使得图像边缘像素作为卷积核的中心进行卷积，避免边缘信息丢失。卷积核的输入通道数量由输入特征图的通道数决定。输出通道数据量为卷积核的个数，提高卷积核数量可以增加提取特征的丰富性，从而提高网络的表达能力。

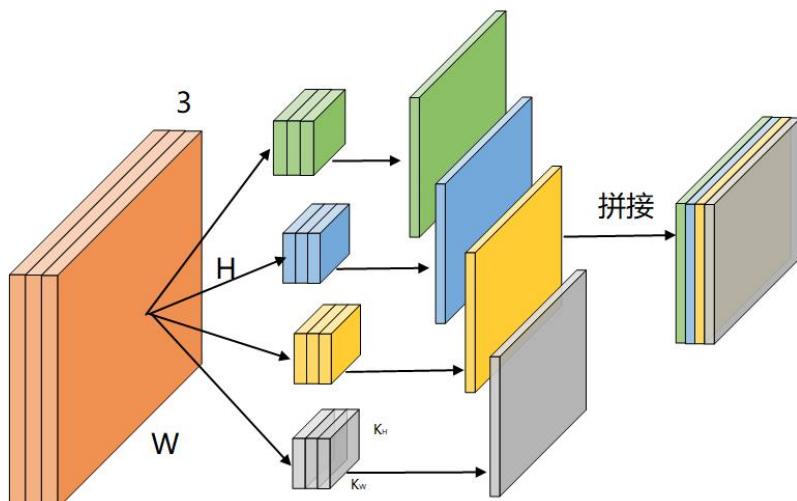


图 2.8 CNN 卷积层示意图

### 3) 池化层

池化层位于卷积层后，池化也称为降采样，主要用于特征选择与降维，降低特征图大小，从而降低后续卷积等操作的计算量，同时能够让网络学习有效特征，提高网络的鲁棒性与泛化性，避免过拟合问题。池化操作通过池化核来完成，池化核在图像局部区域内进行下采样，用一个值代表当前区域特性。然后根据步长在图像上从左向右，从上到下滑动。常用的池化方式有最大池化与平均池化，最大池化在特征图局部区域中选择最大值来表征此区域，最大池化可以使得特征图对比更加显著。平均池化在局部区域特征图中使用均值来表征此区域，整体特征信息更加平滑。通常池化核大小选择为 $2 \times 2$ ，然后将输入特征图划分为多个相同大小的区域，以最大池化的方式进行。

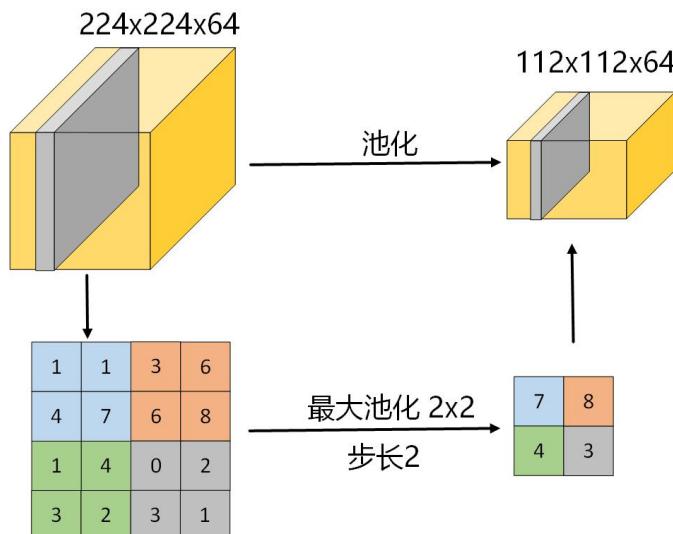


图 2.9 CNN 池化层示意图

## 2.3 软件开发相关技术

本节主要介绍骨髓血细胞检测与识别软件开发使用的技术框架。首先介绍前端核心技术，Vue 框架与 Element UI 组件库。然后介绍后端使用的框架 Django 与深度学习模型部署工具 ONNX。

### 2.3.1 前端技术

前端技术主要用来构建优美的用户图像界面、并提供良好的用户交互。随着互联网技术的飞速发展，前端 web 技术已经步入到 2.0 时代。无论是在移动端还是 PC 端，我们都能体验到设计精妙绝伦的界面，并进行丰富多元的交互，极大提升了用户的浏览效率与用户体验。前端开发的三个核心基础要素是超文本标记语言 HTML、层叠样式表与 Javascript 语言。目前，随着前端技术的不断发展，已存在多

种前端框架与组件库工具，可以帮助开发人员可以快速搭建出风格优美统一的界面，提升开发效率。本小结对软件开发使用到的前端框架与组件库进行简要介绍：

### 1) HTML、CSS 与 Javascript

HTML 的全称是超文本标记语言（Hypertext Markup Language），是用来构建和布局界面内容的描述语言。该语言使用标签结构与属性标记对文字、图像、视频、超链接、表格等内容进行不同形式的呈现。目前 HTML 主要遵循 HTML5 规范，该版本引入了更多种类的标签，使得界面结构更加清晰。此外引入了更多的新表单控件。支持多种音频、视频等多媒体形式，无需三方插件。在性能方面也使用了多线程通信与缓存技术，可以更加快速的访问资源与处理请求。

CSS 的全称是层叠样式表（Cascading Style Sheets），它是一种样式表语言，主要用来定义 HTML 元素的表现形式。通过 CSS 可以对于页面上的元素进行精确的布局描述，并为元素添加圆角、阴影、动画或者复杂的过度效果。此外 CSS 可以根据设备屏幕大小与分辨率对页面的布局和样式进行响应以适应不同的设备。

Javascript 是一种解释型的轻量级 Web 开发编程语言，作为编程代码嵌入到 HTML 页面后，可以由浏览器进行解释执行。通过 javascript 语言可以实现动态效果与用户的交互，例如用户提交表单验证、响应用户的请求、动态界面等。通过上述三种技术来构建内容，设计样式与行为控制，开发者可以快捷、高效、灵活的创建出更好图形界面应用。

### 2) VUE 前端框架

随着前端功能越来越复杂，出现了多种前端框架来简化前端编程开发。JQuery 是最早的前端框架，对 javascript 进行了多种封装，可以更加便捷的操作文档对象模型（Document Object Model）。在 JQuery 的基础上演化出 MVC 架构，即模型（Model）-视图（View）-控制器（Controller）的架构。该架构将界面、数据业务逻辑、信息输入更新进行分离，使得代码易于维护与升级。目前主流的前端架构是 MVVM 架构，如图 2.10 所示，MVVM 是模型（Model）-视图（View）-视图模型（ViewModel）的缩写。模型代表软件应用的数据模型与业务逻辑。视图是用户界面，将数据可视化呈现。ViewModel 监听模型数据改变并控制视图行为，它通过数据的双向绑定将模型与视图连接起来，模型中的变化会自动同步到视图中，视图中的变化也同步到模型中。Vue.js<sup>[48]</sup>是一款轻量级、渐进式的 MVVM 前端框架。相比于其他框架拥有如下的优势：双向数据绑定，避免操作 DOM，易于维护；组件化，将界面拆分为多种组件，每个组件拥有自己的 view 与 model，代码复用性与模型化程度更高；响应式设计，采用虚拟 DOM 与 Diff 算法优化渲染效率，极大减少了修改 DOM 元素的次数；生态环境丰富，Vue 有大量的插件，方便扩展，社

区活跃，大量优秀的开发者在持续的对框架进行维护、更新与完善。

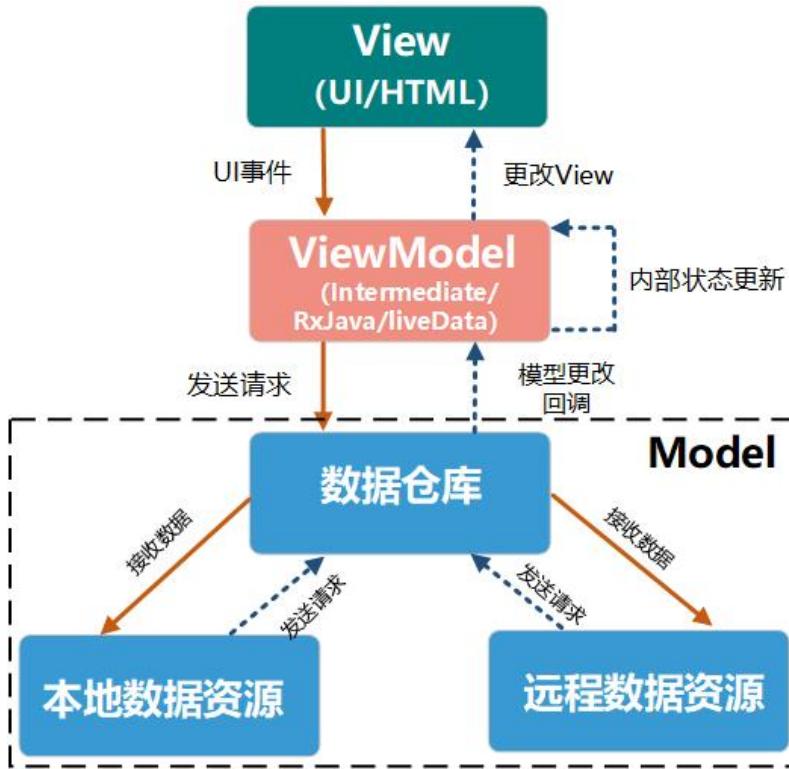


图 2.10 MVVM 框架示意图

### 3) Element-UI 组件库

Element-UI 是一套基于 Vue 2.0 的桌面端组件库，提供了一系列开箱即用的精美 UI 组件，如表单、布局、弹窗、导航、数据展示等多种组件。这些组件的设计简洁、优美、风格统一，可以帮助开发人员高效快速的构建美观、易用的界面程序。Element-UI 提供的 API 接口简单易学，可以轻松的自定义组件扩展。该组件库使用了虚拟滚动技术，极大的提高了页面的渲染性能。

## 2.3.2 后端技术

### 1) 后端框架

在计算机领域，框架是指一套规范、标准、结构完整的程序模版。框架通常包含了一系列的库、工具与接口，开发者可以按照框架的规范快速进行应用程序开发，减少重复工作，提高开发效率同时保障软件的可靠性。目前各个编程语言均有自己的 Web 后端框架，这些框架各具特色，一些框架亮点在分布式、高性能，一些框架优势在于高成熟与高扩展度。

在诸多的编程语言中，Python 是一种解释型、面向对象的高级程序设计语言。其优势在于拥有丰富的标准库、框架与详细的官方文档，并且可以方便的跨平台运行。Python 语言被广泛应用在多种领域如数据分析、人工智能、后端开发。在深

深度学习领域，Pytorch 是由 Facebook 开发的基于 Python 的开源深度学习框架，在学术界与工业界广泛使用。本文使用 Pytorch 框架进行模型训练。

Django<sup>[49]</sup>是一个基于 Python 的开源 Web 后端框架。它最早由 Adrian Holovaty 和 Simon Willison 在 2003 年开发，用于维护劳伦斯集团旗下的几个新闻网页。该框架于 2005 年 7 月在 BSD 许可证下发布，截至目前，Django 已经经历过三个大版本的迭代。最新的 Django 3.0 版本于 2019 年 12 月发布，主要引入了对异步通信编程的原生支持，可以支持更高并发、高流量的应用程序。Django 框架基于 MTV 架构，MTV 的全称是模型（Model）-模板（Template）-视图（View）。模型是数据业务逻辑，主要进行数据库的增删改查。模板可以动态生成 HTML，决定如何对内容进行展示。视图封装负责处理用户请求与返回响应的逻辑。框架结构如 2.11 所示：Django 框架具有如下的优势：（1）高度可扩展，框架的每个组件均可以替换与修改，便于添加与修改功能。（2）强大数据库访问组件，可以根据 Model 快速更改数据库模式，便捷的进行数据迁移，数据库操作代码简洁。（3）安全性，Django 内置了多种安全机制和功能，包括了跨站请求伪造保护、跨站脚本攻击保护、SQL 注入保护与认证与授权等。

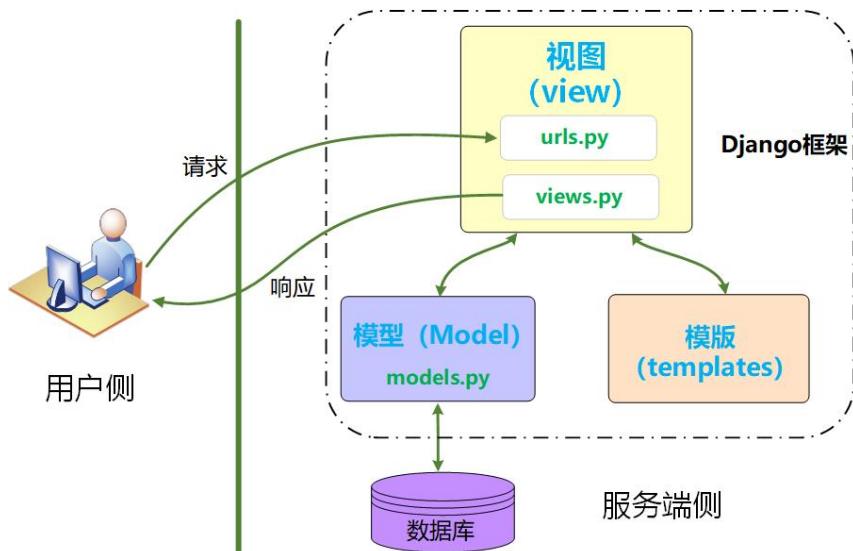


图 2.11 Django 框架示意图

## 2) ONNX 模型部署工具

在深度学习模型训练结束后，我们要让模型能够在生产环境中运行。部署时首先要对运行环境进行配置，此外还要对网络结构进行优化精简提高模型的推理性能。目前针对神经网络的部署主要采用深度学习框架-中间表示-推理引擎这种流水线方式。首先是定义模型结构并选择一种深度学习框架进行训练。之后，将模型结构与参数转化成一种通用的中间表示，在此基础上进行效率优化。最后将中间表示文件在含有推理引擎的硬件平台上高效运行。

开放式神经网络交换 (Open Neural Network Exchange, ONNX) 是微软与 Facebook 在 2017 发布的用于描述计算图的一种格式。它是一种开放的规范，定义了模型的标准数据类型，内置运算算子与可扩展的计算图模型。ONNX 目前支持多种深度学习框架如 Pytorch、Tensorflow、Caffe2、Mxnet 等。在 Pytorch 中可以使用 `torch.onnx.export()` 函数将 Pytorch 模型转为 ONNX 格式的静态计算图模型。ONNX Runtime 是由微软开发的一个跨平台、高性能的机器学习推理引擎，它是 ONNX 文件的运行环境，在该环境下可以读取并运行.onnx 文件。目前 ONNX Runtime 支持多种硬件环境，包括了 CPU、GPU、地平线的 BPU、华为海思等国产推理芯片。

## 第3章 基于深度学习的骨髓血细胞检测算法设计与实现

### 3.1 引言

骨髓血细胞形态学检查是血液疾病诊断的重要依据，主要通过人工镜检来完成，上述过程繁琐枯燥，可靠性差。在骨髓血细胞自动化识别算法中，骨髓血细胞检测是将血细胞从涂片图像中定位并裁剪得到单一的血细胞图像，该过程是后续分类识别基础，直接影响血液疾病诊断的结果，因此一直都是医学图像处理的热点研究方向之一。目前基于深度学习的目标检测算法在很多领域都有广泛应用，如自动驾驶、安防监控、人脸识别、医学影像诊断等，其目的是定位或者跟踪相关目标。

根据文献调研，目前血细胞检测主要是对血细胞图像中的红细胞、白细胞与血小板进行检测。检测算法主要基于通用的深度学习目标检测算法，包括了单阶段检测算法与两阶段检测算法。在两阶段方法中，通过区域举荐网络生成少量感兴趣区域，并将这些区域提取到的特征输入到后续的分类与回归分支中。在单阶段算法中，如 SSD<sup>[50]</sup>、YOLO<sup>[18]</sup>等网络直接将全部图像作为输入，学习类别概率与边界框位置。两类方法各有优劣，本章对这些方法进行阐述，并进行性能分析，最终选取性能较好的 RetinaNet 作为骨髓血细胞检测的基线模型。

### 3.2 双阶段目标检测网络

快速区域卷积网络（Faster-RCNN）是目标检测领域最为经典的双阶段检测器，其网络架构如图3.1所示，主要由骨干网络特征提取器（BackBone）、区域举荐网络（Region proposal Network, RPN）与分类回归网络（R-CNN）这三部分组成。骨干网络为深度卷积神经网络，用于图像特征提取。RPN 网络在提取的特征图上快速生成区域坐标与相应前景分数，这些区域用于后续的分类与坐标回归。R-CNN 网络对于这些区域首先进行 ROI 池化（Region of Interest Pooling）将区域转换为特定大小的特征图，然后这些特征图会分别进入到分类分支与回归分支，得到待检测目标的类别与位置信息。

#### 3.2.1 骨干网络

骨干网络是深度卷积神经网络的主体部分，用于提取图像抽象的语义特征，为后续任务提供图像的嵌入特征向量表示。因此，骨干网络的性能对于整体网络

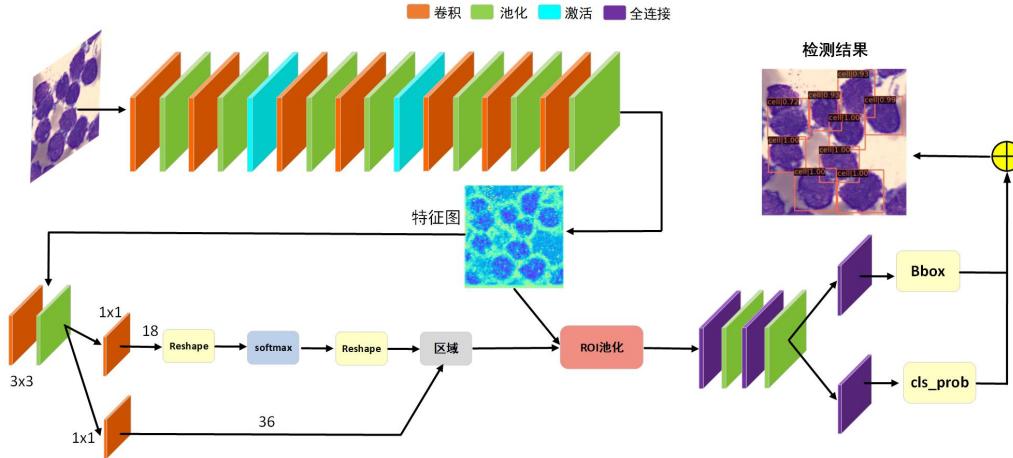


图 3.1 快速区域卷积神经网络结构示意图

性能有很大的影响。常用的骨干网络有 VGG<sup>[51]</sup>、ResNet<sup>[52]</sup>、DenseNet<sup>[53]</sup> 与 Inception<sup>[54]</sup>等，其中 ResNet 是应用最为广泛的骨干网络，其特点是引入了残差模块，可以实现很深层的卷积网络。

综合考虑模型的精度、速度、参数量等因素，本节模型选择的骨干网络均为 ResNet50，其结构如图 3.2 所示。ResNet50 总共由五个阶段 (stage) 构成，第一个阶段由一个  $7 \times 7$  的卷积组成，可视作对图像的预处理。其余阶段均是由沙漏模块 (BottleNeck) 堆叠而成，第二到第五阶段分别有 3、4、6、3 个沙漏模块。若输入图像的大小为  $224 \times 224 \times 3$  每经过一个 stage，特征图的大小减小为原来的一半，通道数变为原来的两倍。五个阶段输出的特征图分别为  $C_1, C_2, \dots, C_5$ ，其中  $C_5$  特征图的大小为  $7 \times 7 \times 2048$ 。最后特征图经过均值池化变为 2048 的向量，经过全连接层用于分类识别。

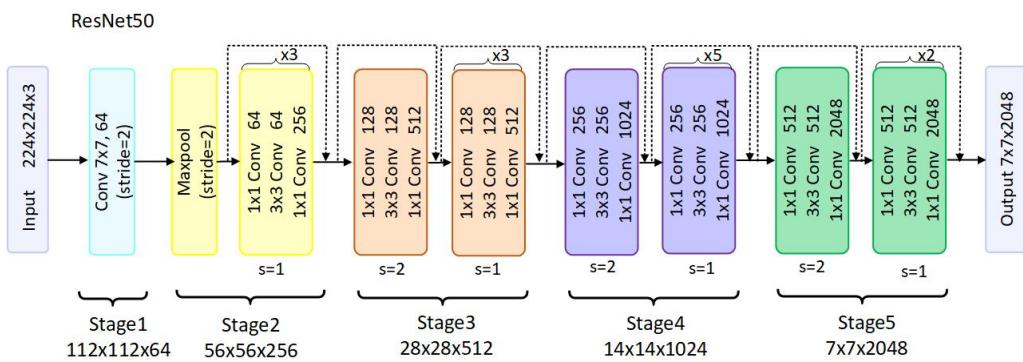


图 3.2 ResNet50 网络结构示意图

沙漏 (BottleNeck) 模块结构如图 3.3 所示，该网络第一个卷积层使用  $1 \times 1$  的卷积核来减少通道数量。第二个卷积层的卷积核大小为  $3 \times 3$ ，当  $\text{stride}=1$  时，特征图大小不变， $\text{stride}=2$  时，特征图大小变为原来的一半。第三个卷积层采用  $1 \times 1$  卷积恢复特征图的通道数。由于中间层的维度较小类似于沙漏，因此称为沙漏结

构。该结构可以有效降低网络的参数量与计算量。该结构还包括了一个残差模块，通过一个  $1 \times 1$  的卷积确保输入与输出的通道数相同后，在与输出直连相加。残差连接可以让网络更好的学习高层特征，同时避免网络浅层部分梯度消失或爆炸等问题。图中 BN 为批量归一化层、ReLU 为激活函数。

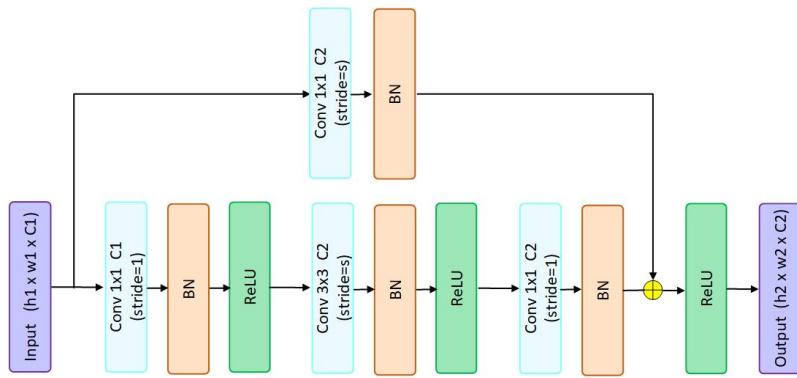


图 3.3 BottleNeck 模块结构示意图

### 3.2.2 特征金字塔网络

特征金字塔网络（Feature Pyramid Network, FPN）主要用来解决多尺度的目标检测问题。骨干网络特征提取器输出不同尺度的特征图，这些特征图的感受野不同。高层特征图感受野比较大，用来检测大尺寸目标，浅层的特征图感受野较小，用来检测小尺寸目标。但是浅层特征图的表达能力较弱，通常只有纹理、边缘形状、明暗等细节信息，而高层特征图则包含更加丰富的全局信息。为解决浅层网络特征表达能力有限的问题，Lin 等<sup>[55]</sup>引入了特征金字塔网络。该网络将顶层特征逐级向下传递并与浅层特征融合，使得浅层特征可以同时兼顾细节与整体具有更加丰富的特征表达。ResNet50 骨干网络构建特征金字塔的结构如图 3.4 所示。图中  $C_1, C_2, \dots, C_5$  为 ResNet 骨干网络生成的不同尺度特征图，相邻两个阶段的特征图在尺寸上为二倍缩放的关系。自顶向下的融合过程经过上采样与通道调整使得特征图的尺寸一致后再相加融合。以  $P_4$  特征图的生成为例， $P_5$  由  $C_5$  特征图经过  $1 \times 1$ 、通道数为 256 的卷积层得到。 $P_5$  经过二倍上采样（由反卷积实现）与  $C_4$  经过  $1 \times 1$ 、通道数为 256 的卷积的结果相加得到  $P_4$ 。其他特征图同样由上述过程生成。最后 FPN 使用  $3 \times 3$  卷积对融合后的特征图进行平滑处理消除直接相加可能导致的融合不充分问题。至此，完成了特征金字塔的构建。

### 3.2.3 区域举荐网络

区域举荐网络（Region Proposal Network, RPN）基于骨干网络提取的特征图生成一系列候选框区域，用于后续分类识别。RPN 网络结构如图 3.5 所示，包含了两个分支，上面一条分支用于预测每个锚框属于前景的分数。下面一条分支用于

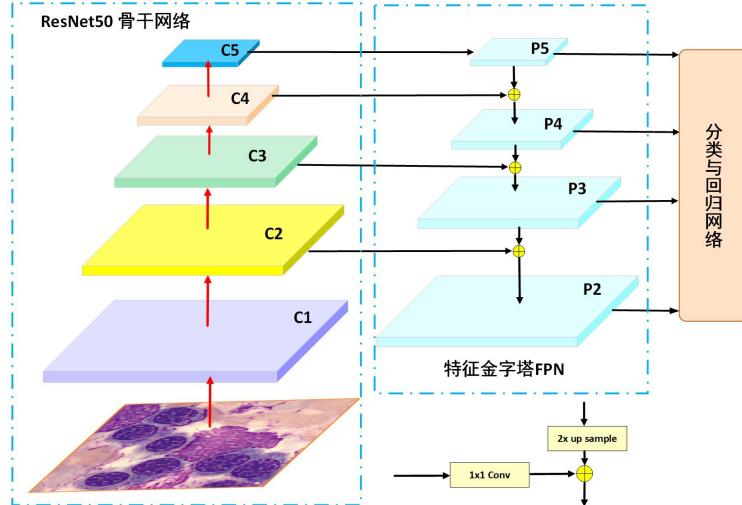


图 3.4 特征金字塔网络结构示意图

计算锚框边界坐标的回归信息，以生成更加精准的区域坐标。最后生成的候选框区域综合了前景分数与坐标修正，实现了目标的初步定位。

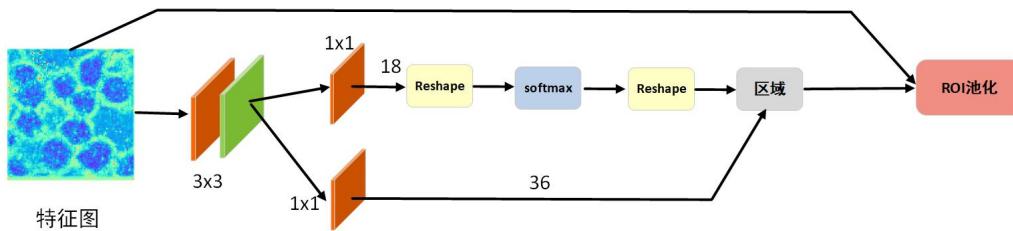


图 3.5 特征金字塔网络结构示意图

锚框是 RPN 网络根据预先定义的参数生成的一些列矩形区域，对于特征图上的每个锚点会生成  $k$  个锚框，通常将  $k$  设置为 9。如图3.6(a) 所示，九个锚框有三种尺寸与三种长宽比，锚框尺寸由数据集目标先验信息与特征图大小决定，尺度变化范围为  $2^0, 2^{\frac{1}{3}}, 2^{\frac{2}{3}}$ ，长宽比通常设定为 0.5, 1, 2。以骨髓血细胞图像为例，原图尺寸为  $363 \times 360 \times 3$ ，首先经过缩放与填充转大小换为  $832 \times 800 \times 3$ ，所有 anchor 的尺寸在  $32 \times 32 \sim 812 \times 812$ ，几乎可以覆盖各个尺寸的目标。以  $P_2$  特征图为例，该特征图的尺寸为  $208 \times 200$ ，缩放尺度为 4，九种 anchor 的尺寸分别为  $23 \times 45, 32 \times 32, 45 \times 23, 29 \times 57, 40 \times 40, 57 \times 29, 36 \times 72, 51 \times 51, 72 \times 36$ 。

特征图经过预测前景分支后输出的尺寸为  $h \times w \times 2k$ ，对应锚点的  $k$  个锚框，每个锚框有前景与背景两种分数。经过坐标回归分支后得到尺寸为  $h \times w \times 4k$  的坐标回归结果。我们采用  $(x, y, w, h)$  这样的四维向量来表示矩形框，分别代表矩形框的中心与宽高。如图3.6(b) 所示，红色框 A 为按照预先定义生成的锚框  $A = (A_x, A_y, A_w, A_h)$ ，绿色框代表目标的真值框  $GT = (G_x, G_y, G_w, G_h)$ 。坐标回归分支希望学习到一个映射  $F(A_x, A_y, A_w, A_h) = (G'_x, G'_y, G'_w, G'_h)$ ，使得原始的锚框 A 经过映射后得到的预测框  $G'$  能够更加接近真实框  $GT$ 。

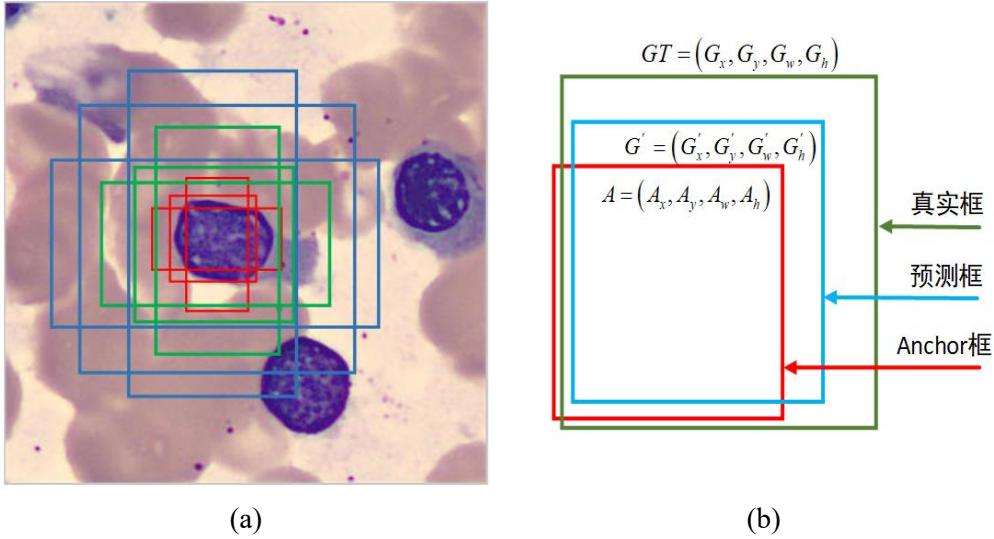


图 3.6 (a) Anchor 在图像中的示意图, (b) Anchor、预测框与真值框之间的关系

变换  $F$  中包含的参数通过坐标回归分支得到  $d_*(A) = W_*^T \cdot \phi(A)$ , 其中  $\phi(A)$  为锚点对应的特征向量,  $W_*$  为  $1 \times 1$  卷积层需学习的参数,  $d_*(A) = d_x(A), d_y(A), d_w(A), d_h(A)$ , 锚框  $A$  与预测框  $G'$  的坐标关系如式 3.1 所示。

$$\begin{aligned} G'_x &= A_w \cdot d_x(A) + A_x & G'_y &= A_h \cdot d_y(A) + A_y \\ G'_w &= A_w \cdot \exp(d_w(A)) & G'_h &= A_h \cdot \exp(d_h(A)) \end{aligned} \quad (3.1)$$

真实框 GT 与锚框 A 之间的平移量  $(t_x, t_y)$  与尺度因子  $(t_w, t_h)$  的变换计算如式 3.2 所示。

$$\begin{aligned} t_x &= (G_x - A_x) / A_w & t_y &= (G_y - A_y) / A_h \\ t_w &= \log(G_w / A_w) & t_h &= \log(G_h / A_h) \end{aligned} \quad (3.2)$$

坐标回归分支网络在训练时的优化目标是让预测值  $d_*(A)$  与真实变换系数  $t_*$  之间的差距最小, 选择 smooth-L1 损失函数进行优化。在测试时, 网络输出修正的坐标变换信息将锚框修正, 用于后续处理。

$$\hat{W}_* = \operatorname{argmin}_{W_*} \sum_i^n \operatorname{smooth}_{L_1}(t_*^i - W_*^T \cdot \phi(A^i)) + \lambda \|W_*\| \quad (3.3)$$

$$\operatorname{smooth}_{L_1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherswise,} \end{cases} \quad (3.4)$$

### 3.2.4 分类与回归网络

RPN 网络生成的候选区域大小形状各不相同, 需要经过 ROI 池化保证尺寸相同。首先把候选区域对应位置映射到特征图上, 再将对应特征图区域划分为  $pool_w \times pool_h$  的网格, 在每个网格区域内进行最大池化, 这样每个区域转化为  $pool_w \times pool_h$

固定大小输出。

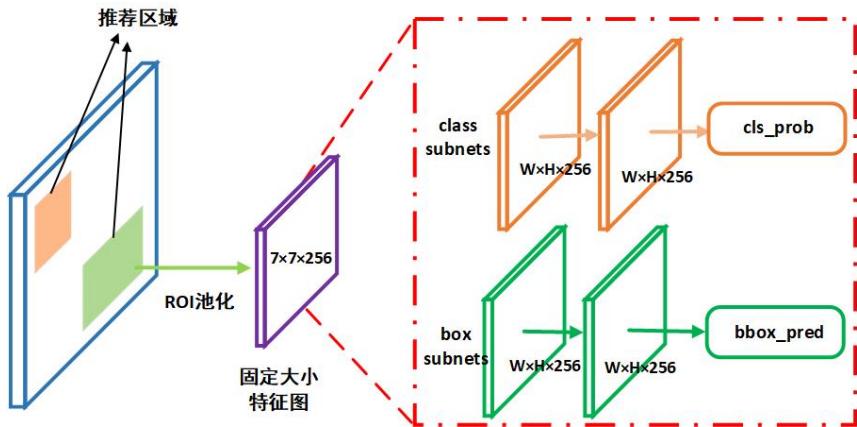


图 3.7 分类与回归网络结构示意图

分类网络使用 ROI 池化后的特征图，经过全连接层与 Softmax 函数输出分类概率向量，在训练时使用交叉熵损失函数进行优化。回归分支与 RPN 网络部分的坐标回归分支类似，目的是为了获得更加精确的检测框坐标。

### 3.3 单阶段目标检测网络

#### 3.3.1 网络结构

单阶段检测器如 SSD、RetianNet、YOLO 系列等具有较快的检测速度与较高的检测精度，通常用于实时目标检测。单阶段检测器在特征图上进行密集检测，对所有的锚框均进行分类与回归。在图像中大量的锚框都属于负样本，因此单阶段检测器在训练阶段面临着严重的正负样本不均衡问题，大量易分的负样本主导了损失函数的训练，使得网络难以对正样本进行学习，影响参数收敛，网络性能差。两阶段网络如 Faster-RCNN 通过区域举荐网络对所有锚框进行前景分数预测，并将可能包含目标的少量候选框传给后续的分类回归网络进行识别，有效的控制了正负样本比例，很好的解决了上述问题。

为解决单阶段检测器中极度不平衡的正负样本问题，Lin<sup>[19]</sup>等提出了一种简单且实用的焦点损失函数 (Focal Loss)，并设计了 RetianNet 网络。在该损失函数引导下，网络增加了对难以区分正样本的关注，减少了对易分辨背景样本的关注。RetinaNet 网络结构如图 3.8 所示，由骨干网络、特征金字塔网络与分类回归网络组成，网络主体结构在 3.2 节已进行详细阐述。

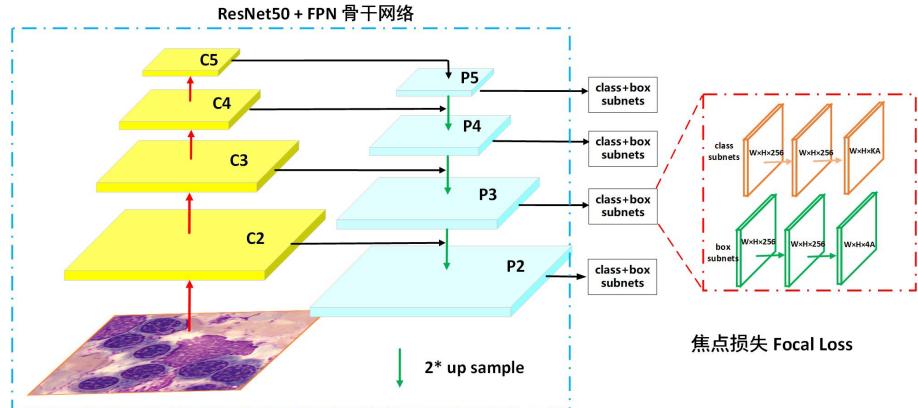


图 3.8 分类与回归网络结构示意图

### 3.3.2 焦点损失函数

焦点损失是基于交叉损失函数提出，在深度学习中，交叉熵用来衡量模型预测结果分布与真实结果分布之间的差异性，一般用于分类任务。假设有  $N$  个样本，类别数为  $C$ ， $y_{ij}$  表示第  $i$  个样本属于第  $j$  类的标签，取值为 0, 1。 $\hat{y}_{ij}$  表示模型预测第  $i$  个样本属于第  $j$  的概率，则交叉熵损失函数如式 3.5 所示：

$$\text{CE\_Loss}(y, \hat{y}) = -\frac{1}{N} \sum_{i=1}^N \sum_{j=1}^C y_{ij} \log(\hat{y}_{ij}) \quad (3.5)$$

在二分类问题中，只有两个正负样本，若  $p$  表示预测为正样本的概率， $1 - p$  表示预测为负样本的概率，样本标签取值为 0, 1，则二元交叉熵如式 3.6 所示：

$$\text{BCE} = \begin{cases} -\log(p), & \text{if } y = 1 \\ -\log(1 - p), & \text{if } y = 0 \end{cases} \quad (3.6)$$

常见解决正负样本不平衡的方法是引入  $\alpha$  参数平衡正负样本损失函数的权重。该参数由正负样本比例决定  $\frac{\alpha}{1-\alpha} = \frac{n}{m}$ ，式中  $n$  为负样本数量， $m$  为正样本数量。

$$\text{BCE} = \begin{cases} -\alpha \log(p), & \text{if } y = 1 \\ -(1 - \alpha) \log(1 - p), & \text{if } y = 0 \end{cases} \quad (3.7)$$

焦点损失在平衡二元交叉熵损失基础上加入了难易调整因子  $\gamma$ ，训练过程中，大量的锚框都属于易分的背景样本，这些置信度较高的样本对于模型效果提升影响较小。焦点损失对于高置信度样本进行惩罚，降低其损失函数中的权重  $(1 - p_t)^\gamma$ ，使得网络关注到难分的正负样本。焦点损失如式 3.8 所示。 $\gamma = 0$  时焦点损失等价于平衡二元交叉熵损失函数， $\gamma = 2$  时网络的效果较好，当分类概率  $p_t$  为 0.9 时，其损失函数权重降低 100 倍。而分类概率  $p_t = 0.5$  时，损失函数权重仅降低四倍。这使得网络在训练时能不断地聚焦在那些学习较差的样本上，梯度更新方向主要

由困难样本决定。 $\alpha$  是类别权重，正负样本的  $\alpha$  通常设置为 (0.25, 0.75)。

$$\text{Focal\_Loss}(p_t) = -\alpha(1-p_t)^\gamma \log(p_t)$$

$$p_t = \begin{cases} p, & \text{if } y = 1 \\ 1-p, & \text{if } y = 0 \end{cases} \quad (3.8)$$

### 3.3.3 网络预测

在网络的预测推理阶段，RetinaNet 在特征金字塔输出的多尺度特征图上进行预测。对于骨髓血细胞图像首先通过双线性插值缩放到  $832 \times 800$  大小，其特征图  $P_2, P_3, \dots, P_6$  的大小分别为  $208 \times 200, 104 \times 100, 52 \times 50, 26 \times 25, 13 \times 13$ ，特征图上的每个锚点输出 9 个锚框，对于一张骨髓血细胞图像，网络总共输出  $55250 \times 9$  个检测结果。每个检测结果包括了一个分类置信度向量与一个四维坐标回归向量，根据坐标回归向量可以计算出检测框在原始图像上的坐标。对于每个骨髓血细胞类别，大量的检测框均为背景区域，首先将置信度低于 0.05 的检测框滤除。然后对检测框按照置信度从高到低进行排序，选取前 1000 个检测框进行非极大值抑制 (Non-Maximum Suppresion, NMS)<sup>[56]</sup> 去除重复的检测框。NMS 的过程如下：首先从置信度最高的框开始，计算它与剩余框的交并比 (IOU)，并去除交并比大于阈值（通常为 0.5）的框。然后在剩余框中继续选择置信度最高的框重复上述过程，最后保留的框为非极大值抑制后处理后的结果。最终合并所有血细胞类别的检测结果，选择置信度最大的前 100 个框作为当前图片的检测框结果。

## 3.4 算法实现与实验结果分析

研究初期，我们需要对比不同目标检测网络的性能，确定骨髓血细胞检测的基线模型，然后在此基础上进行改进。在双阶段网络中，我们选择了 Faster-RCNN<sup>[17]</sup>、Cascade-RCNN<sup>[57]</sup> 网络。在单阶段网络中，我们选择 RetinaNet<sup>[19]</sup> 与 YOLOV3<sup>[58]</sup> 网络。上述网络都是经典的具有代表性的检测网络。我们比较了不同网络的参数量与浮点计算量与在骨髓血细胞数据集上的检测精度。此外，我们探索了网络在仅做检测与检测识别一体化任务上性能的差异。

### 3.4.1 实验环境

#### 3.4.1.1 数据集介绍

骨髓血细胞图像来自邃蓝智能科技 (上海) 有限公司合作医院提供，首先采用第 2.1 小节阐述的主动学习标注策略进行边界框的标注。我们总共标记了 6821 张血细胞图像，训练集与测试集按照 4:1 的比例进行随机划分，训练集包含了 5456

张图像，测试集包含了 1365 张图像。通常每个图像中包含 1 到 10 个有核血细胞，数据集总共标记了 11352 个血细胞，训练集有 9065 个血细胞，测试集有 2287 个血细胞。数据集的分布如表 3.1 所示：

表 3.1 骨髓血细胞检测数据集分布

序号	类别名	类别简写	训练集数量	测试集数量
1	原始细胞	Prim	1856	467
2	淋巴细胞	Lym	996	226
3	单核细胞	Mono	206	52
4	浆细胞	Plas	272	70
5	红细胞	Red	1880	503
6	早幼粒细胞	Promy	357	107
7	嗜中性中幼粒细胞	Myelo	701	150
8	嗜中性晚幼粒细胞	Late	503	144
9	嗜中性杆状核细胞	Rods	998	241
10	嗜中性分叶核细胞	Lobu	821	195
11	嗜酸性粒细胞	Eosl	475	132
总计			9065	2287

### 3.4.1.2 评价指标

在目标检测领域，通常使用检测框与真实框的交并比（IOU）来判断检测框是属于正样本还是负样本。交并比的定义如图 3.9 所示，对于预测框 A 与真实框 B，IOU 值等于两个矩形框的交集面积  $S(A \cap B)$  与并集面积  $S(A \cup B)$  的比值。IOU 的取值范围为 0 ~ 1，值越大表示两个矩形相似程度越高。

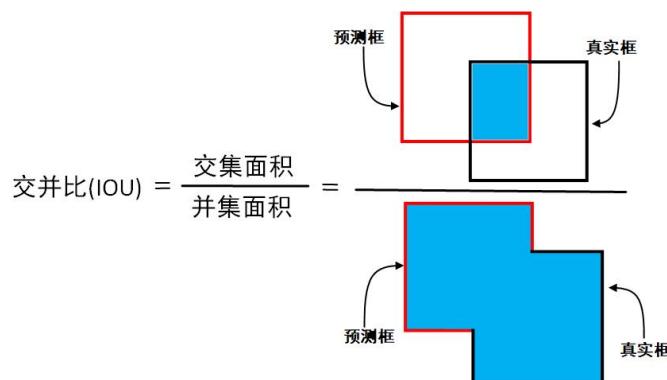


图 3.9 IOU 计算示意图

在深度学习中，TP (True Positive) 表示被正确检测为正类的正样本；FP (False

Positive) 表示被错误检测为正类的负样本; TN (True Negative) 表示被正确检测为负类的负样本。FN (False Negative) 表示被错误检测为负类的正样本。对于检测任务, 需要根据检测框的置信度分数与交并比来判断以图 3.10 为例, 绿色框为真实框, 红色框为网络检出框。第一张图中检测框与真实框 IOU 大于阈值, 且类别正确, 为真正例 TP。第二张图中 IOU 大于阈值, 但是类别检测错误, 分叶核 (Lobu) 真实类别被错误检测为杆状核 (Rods) 类别 (FP)。第三张图中, 检测框与所有真实框 IOU 均小于阈值, 背景类被错误检测为 Lobu 类 (FP)。第四张图片未检出, Lobu 类别被错误检测为背景类 (FN)。

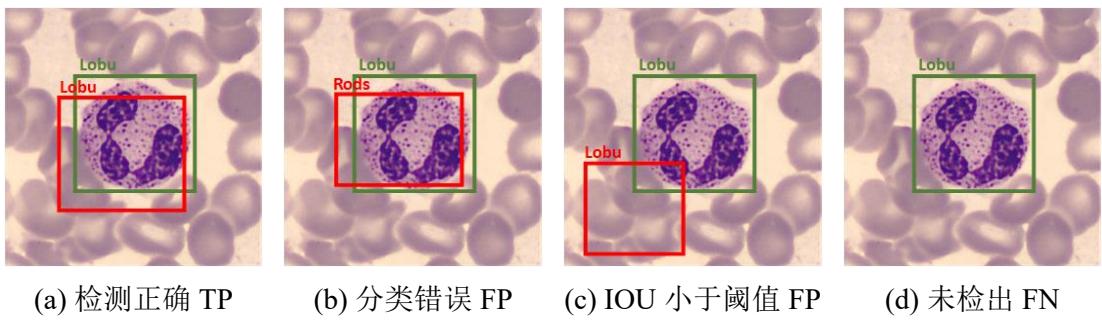


图 3.10 检测类型判别

目标检测常用的评价指标有精准率 (precision)、召回率 (recall)、PR 曲线、平均精度 (Average Precision, AP) 与平均精度均值 (Mean Average Precision, mAP)。

精准率是指预测正确的正样本数量与所有预测为正样本数量的比值, 如式 3.9 所示:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (3.9)$$

召回率是指预测正确正样本数量与所有正样本数量的比值, 如式 3.10 所示

$$\text{Precision} = \frac{TP}{TP + FN} \quad (3.10)$$

PR 曲线是以召回率为横坐标, 以精确率为纵坐标绘制的曲线。在绘制某一类别 PR 曲线时, 将检测框按照置信度从高到低进行排序, 设定多组置信度阈值, 根据交并比会得到多组精确率与召回率, 最后将这些点连接起来得到最终的 PR 曲线。随着召回率的提高, 检测的精确率会降低, 曲线越靠近右上角代表网络的检测性能越好平均精度是 PR 曲线下的面积, 即在多组召回率下的平均检测精确率。AP 值越大代表模型的性能越好。平均精度均值 (mAP) 是对所有类别的平均精度求均值, 通常用来衡量网络的检测效果。

### 3.4.1.3 实验设置

我们在 Linux 操作系统下的 NVIDIA TITAN V 显卡上训练模型。训练使用的深度学习框架为 Pytorch1.13.1，批量大小设置为 16。优化器使用随机梯度下降算法（SGD），动量设置为 0.9，权重衰减设置为  $5e-4$ 。学习率初始化为 0.001，网络总共训练 36 个轮次，在第 16 与第 28 个轮次，学习率变为原来的 1/10。RetinaNet 中 Focal Loss 中正负样本平衡参数  $\alpha = 0.25$ ，难易调整参数  $\gamma = 2$ 。数据集原始图像大小为  $363 \times 300$ ，为了更好的检测小目标，图像经过双线性插值与填充扩大到  $832 \times 800$ 。我们基于迁移学习的方式进行训练，单双阶段网络参数初始化均使用在 COCO 数据集上预训练的模型，然后使用骨髓学习检测数据集进行微调。

## 3.4.2 实验结果与分析

### 3.4.2.1 参数量、计算量与速度

Faster-RCNN、Cascade-RCNN、RetinaNet 与 YOLOV3 四种网络在输入图像大小为  $832 \times 800$  的情况下的参数量与计算量如表 3.2 所示。我们比较了四种网络在 NVIDIA TITAN V 硬件环境下的每秒帧率（Frame Per Second, FPS），即每秒可以处理的图像数量，来评估网络是否可以满足在生产环境实时性的要求。

表 3.2 不同网络结构参数量、计算量与速度对比

方法	骨干网络	参数量 (MB)	计算量 (GFLOPs)	FPS (TITAN V)
Faster-RCNN	ResNet50	41.17	139.25	27.7
Cascade-RCNN	ResNet50	69.17	167.24	21.9
RetinaNet	ResNet50	36.31	135.73	29.5
YOLOV3	DarkNet53	61.95	127.11	32.3

从表中可以看出，四种网络的 FPS 均大于 20，可以满足骨髓血细胞检测速度实时性的要求。单阶段网络的速度高于双阶段的目标检测网络。RetinaNet 网络参数量最小，计算消耗资源也相对较少，更适合在生产环境中进行部署。

### 3.4.2.2 检测算法精度对比

四种检测网络在骨髓血细胞测试集上的结果如表 3.3 所示，其中  $AP_{Prim}$  表示原始细胞在  $IOU = 0.75$  下的 AP 值，其他  $AP_*$  代表各类血细胞的 AP 值。 $mAP$  为所有类别 AP 的平均值，可以直观的衡量网络检测效果。

从表中可以看出，所有检测器针对单核细胞 (Mono) 的检测效果较差。AP 值均小于 0.5，针对有核红细胞的检测效果最好，AP 值大于 0.85，单核细胞在数据集

表3.3 不同目标检测方法在骨髓血细胞测试集上的检测结果

方法	$AP_{prim}$	$AP_{lym}$	$AP_{mono}$	$AP_{plas}$	$AP_{red}$	$AP_{promy}$
Faster-RCNN	0.861	0.690	0.262	0.715	0.871	0.640
Cascade-RCNN	0.860	0.621	0.409	0.683	0.875	0.620
RetinaNet	<b>0.879</b>	0.713	0.402	<b>0.721</b>	<b>0.928</b>	0.662
YOLOV3	0.840	0.784	0.385	0.705	0.908	0.540
方法	$AP_{myelo}$	$AP_{late}$	$AP_{rods}$	$AP_{lobu}$	$AP_{eosl}$	mAP
Faster-RCNN	0.678	0.656	0.672	0.712	0.754	0.682
Cascade-RCNN	0.721	0.724	0.787	0.846	0.775	<b>0.719</b>
RetinaNet	<b>0.750</b>	0.558	0.638	0.847	0.735	0.709
YOLOV3	0.714	0.664	0.769	0.793	0.766	0.699

中样本数量最少，有核红细胞在数据集中的样本数量最多，因此网络学习的效果与数据集中的样本量有关，需要增加类别较差样本的数量来提升该类血细胞的检测精度。平均检测精度（mAP）最高的网络为双阶段网络 Cascade-RCNN，单阶网络 RetinaNet 次之，Faster-RCNN 检测精度最差。RetinaNet 在多类血细胞如原始细胞、浆细胞、红细胞等平均检测精度最高，虽然平均检测精度稍低于 Cascade-RCNN，但相比于 Cascade-RCNN 的计算量更少与速度更快，具有更高的性价比。

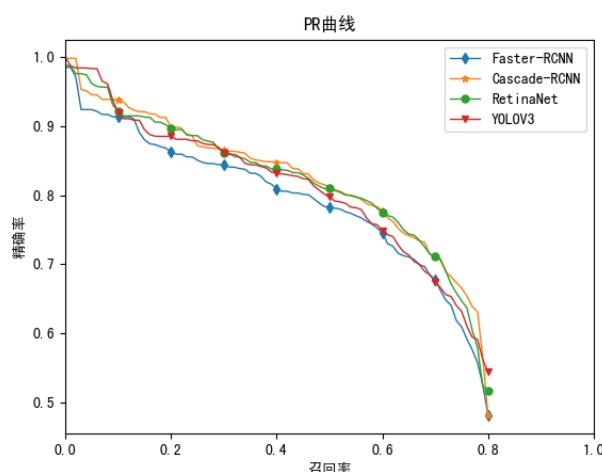


图 3.11 四种检测网络的 PR 曲线 (IOU=0.75)

骨髓血细胞检测是多类别检测问题，对于每一类血细胞都有自己的精确率与

召回率关系。为了更加直观的比较不同网络的结果，对于某一召回率，我们计算所有类别的平均精确率来绘制 PR 曲线。四种网络在  $IOU = 0.75$  的条件下 PR 曲线如图 3.11 所示。四种网络每类血细胞的 PR 曲线如图 3.12(a)-(d) 所示。

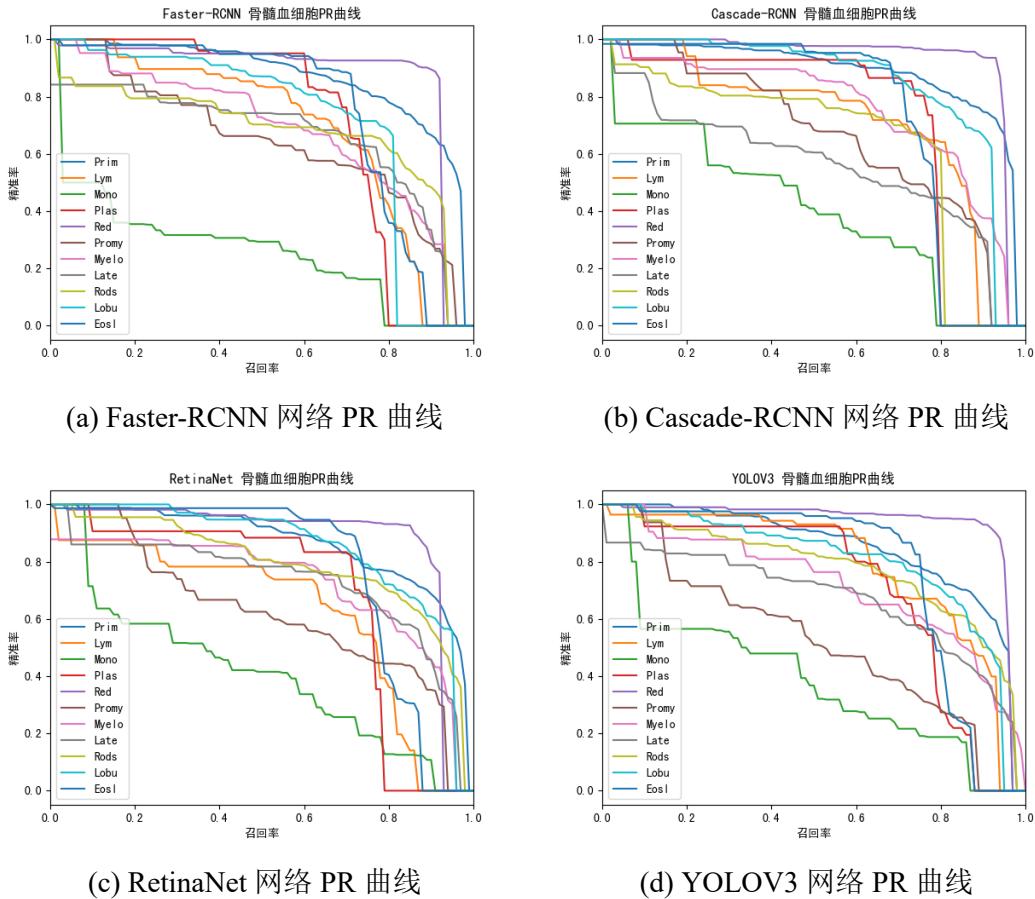


图 3.12 四种检测网络的 PR 曲线

为了进一步直观了解网络的检测性能，我们将检测框进行可视化绘制，如图 3.13 所示，图中仅展示置信度大于 0.6 的检测框结果。图中 (a) 为血细胞人工标注的结果，(b)-(e) 为网络检测的结果。从中我们可以看出原始细胞与嗜中性中幼粒细胞易发生混淆，在第二列图像中，只有 Cascade-RCNN 网络检测正确，Faster-RCNN 与 RetinaNet 均出现了虚警框。对于第四列图像中的破碎细胞，只有 RetinaNet 将其识别为背景，而其他网络均出现了虚警。实验结果表明，四种目标检测网络可以有效的检测并识别图像中的骨髓血细胞，但识别的准确率有待提升。

综合考虑计算量，参数量与检测精度等因素，我们选择性能较为优异 RetinaNet 作为基线模型，并基于骨髓血细胞特性对 RetinaNet 进行改进。

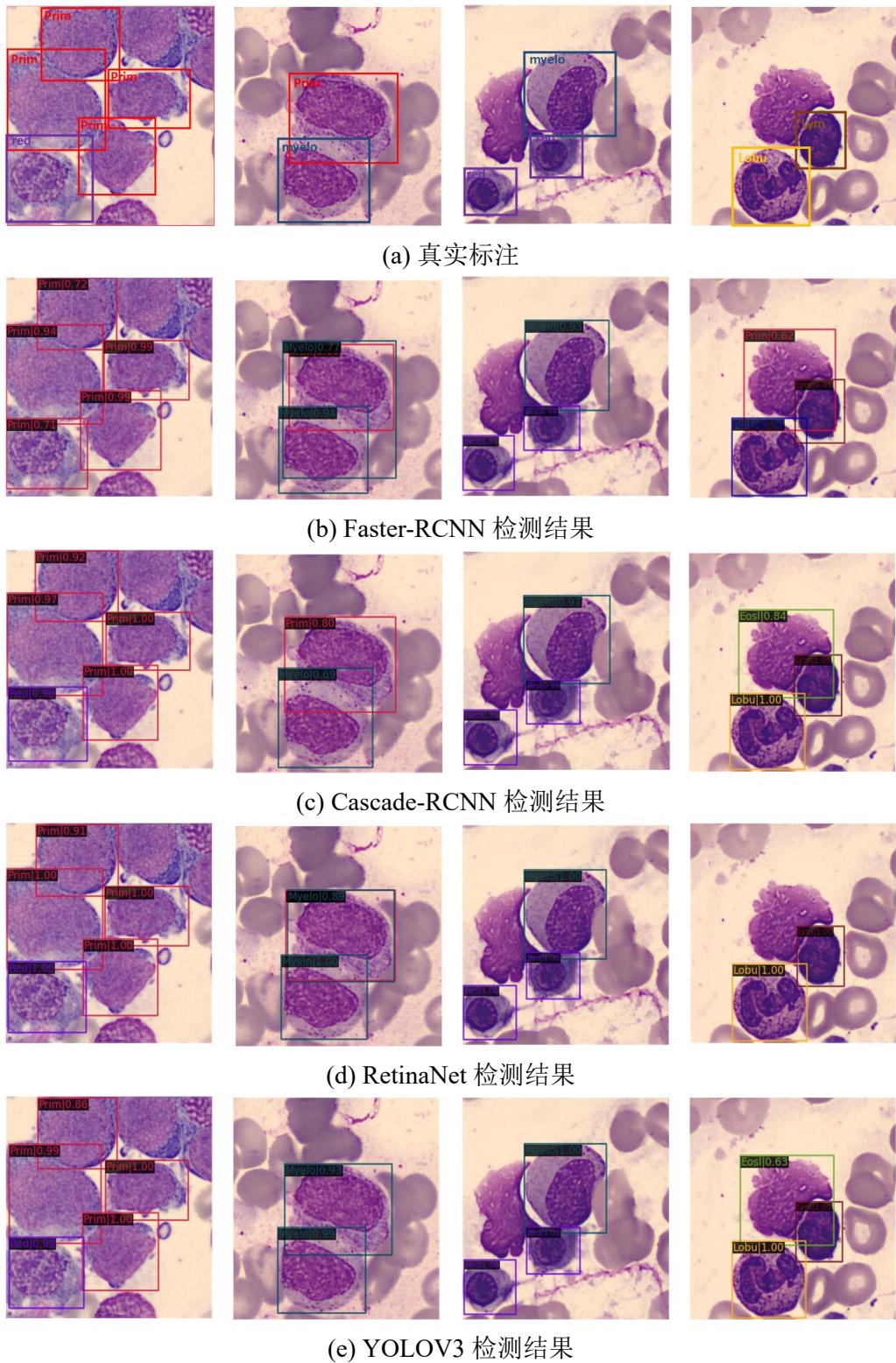


图 3.13 四种检测网络的可视化检测结果

### 3.4.2.3 检测网络与检测识别网络性能对比

骨髓血细胞检测识别网络需要给出骨髓血细胞的位置与类别，检测网络只需要进行前景检测输出血细胞的位置信息。我们以 RetinaNet 单阶段网络作为基线模型，对比了检测网络与检测识别网络在特征图与混淆矩阵上的差异。

骨髓血细胞检测识别的混淆矩阵的定义如下，首先对网络输出的检测框按照分类置信度进行过滤，只有分类置信度大于阈值的检测框才参与混淆矩阵的计算，然后基于 IOU 与标签信息来衡量检测结果是否正确。RetinaNet 检测识别网络在 score (分类置信度)  $> 0.3$ 、 $\text{IOU}=0.5$  条件下的混淆矩阵如图 3.14(a) 所示。各个类别的召回率与精确率如表 3.4 所示。单核细胞 (Mono)、杆状核细胞 (Rods) 与嗜酸性粒细胞 (Eosl) 存在比较严重的漏检问题，召回率小于 0.5。嗜中性中幼粒细胞 (myelo) 与嗜中性晚幼粒细胞 (Late) 的识别准确率较差。检测识别网络平均的检测正确率只有 0.581。RetinaNet 检测网络在 score (分类置信度)  $> 0.95$ 、 $\text{IOU}=0.5$  条件下的混淆矩阵如图 3.14(b) 所示，骨髓血细胞的召回率 0.964，正确率为 0.934。检测网络在  $\text{IOU} = 0.75$  条件下的平均精度 (AP) 为 0.954。实验结果表明，在仅做检测后，骨髓血细胞召回率与准确率极高，可以满足实际应用需求。

表 3.4 RetinaNet 网络各类别的准确率与召回率

	prim	lym	mono	plas	red	promy
Precision	0.644	0.621	0.553	0.783	0.787	0.512
Recall	0.780	0.682	0.236	0.591	0.888	0.454
	myelo	late	rods	lobu	eosl	平均
Precision	0.448	0.439	0.778	0.715	0.857	0.581
Recall	0.646	0.506	0.319	0.672	0.505	

检测网络与检测识别网络的骨干网络输出的均值特征图如图所示，第一行为检测网络在第 2 ~ 5 阶段输出的特征图，第二行为检测识别网络输出的特征图。从图中可以发现检测网络特征图主要关注细胞的边缘信息，而检测识别网络需要关注细胞核、细胞质等全局信息。我们认为在检测识别网络中，识别任务的难度要远高于坐标回归检测任务。识别任务需要提取精细、丰富的全局特征，坐标回归任务只需关注中心、边缘等局部特征。两类任务虽然在头部由不同的分支进行学习，但是共享骨干网络提取的特征，由于任务之间可能存在相互干扰，梯度更新方向不一致，导致骨干网络难以对精细分类特征进行学习，因此识别效果较差。

由于检测识别一体化的网络效果较差，不能满足实际应用的需求。因此，我们

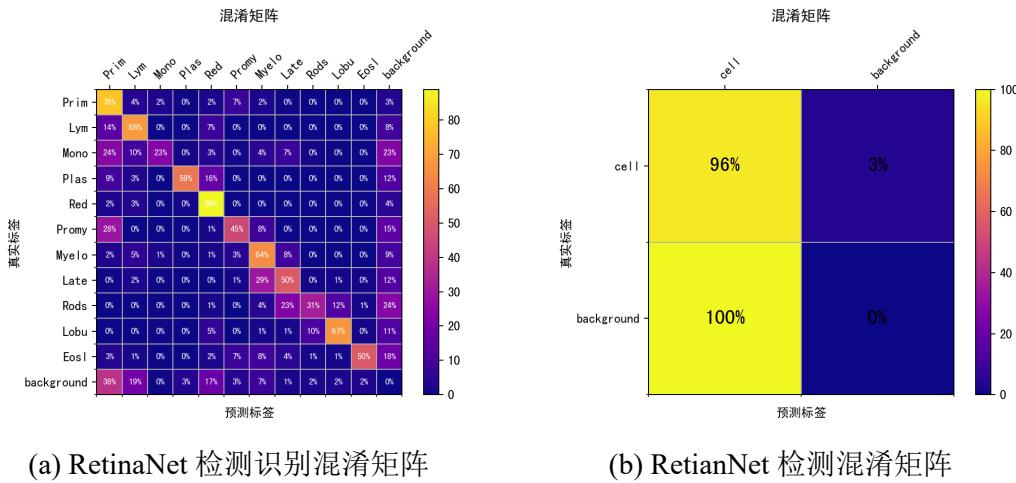


图 3.14 RetinaNet 检测识别与检测网络的混淆矩阵

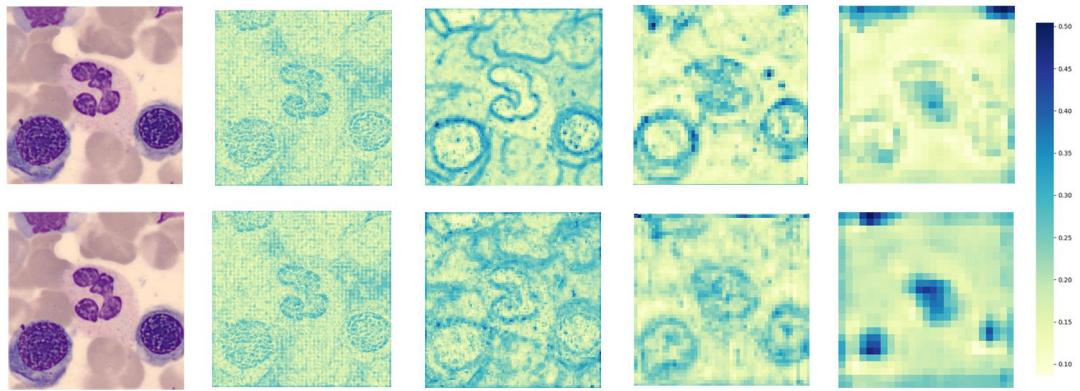


图 3.15 检测网络与检测识别网络特征图对比

先用目标检测网络进行血细胞前景检测与坐标回归，再将血细胞裁剪为切片，然后使用血细胞识别网络进行识别。通过将检测任务与识别任务解耦来提高血细胞分类计数任务的准确性。

### 3.5 小结

针对骨髓血细胞的检测，本章首先分析对比了单双阶段目标检测网络的差异，然后评估了不同检测网络在骨髓血细胞数据集上的检测精度、计算量与速度等性能。实验结果表明 RetinaNet 单阶段网络在速度与精度上要优于其他目标检测网络，因此我们选择将其作为骨髓血细胞检测的基线模型。此外我们对比了检测网络与检测识别网络在特征提取与识别准确率上的差异，实验结果表明检测识别网络的输出类别置信度较低，平均识别正确率只有 58.1%，易发生漏检与误检。检测网络的平均精度达到了 94.2%，网络检测的准确率与召回率都有大幅提升。因此我们确定了先检测再识别的骨髓血细胞处理流程，即先使用检测网络定位到血细胞的位

置，然后剪切为血细胞切片，再输入到识别网络中进行分类。

## 第4章 基于改进的 RetinaNet 骨髓血细胞检测算法设计与实现

### 4.1 引言

第三章中,我们对比了多个单阶段与双阶段检测网络在骨髓血细胞数据上的性能,综合考虑计算量,参数量与检测精度等因素,我们选择将性能优异的 RetinaNet 作为基线模型。此外我们确定了先检测再识别的骨髓血细胞处理流程,即检测网络只需要对血细胞进行前景检测与坐标回归。在 RetinaNet 基线模型中,尽管模型的检测精度已经很高,但仍然存在着漏检、密集与重叠的血细胞区域边界检测错误等问题,如图 4.1 所示。

骨髓血细胞检测主要有如下三个难点:(1)相比外周血红细胞、白细胞、血小板三类血细胞检测,骨髓血细胞种类繁多、形态丰富、尺寸大小不一。(2)在骨髓涂片制作过程中,由于染色剂与光照条件的变化,多个批次的数据存在着色彩差异。此外图像背景复杂,存在较多成熟红细胞的干扰。(3)对于骨髓细胞增生活跃的切片,存在大量血细胞密集堆叠,边缘黏连,易导致漏检、错检等问题。因此精准检测到骨髓血细胞是一项十分具有挑战性的课题。

针对上述难点与基线模型中存在的问题,本章提出了一种改进 RetinaNet 网络。该方法中,我们提出了一种基于全局注意力的自底向上的路径聚合网络结构,缩短了底层与顶层特征之间的信息传递路径,提升网络对位置特征的提取能力。此外探究了不同标签分配策略对检测性能的影响,提出基于最优输运的标签策略用于密集区域的血细胞的标签分配,避免了模糊分配样本的出现,提高网络对血细胞的召回能力。在骨髓血细胞数据集上的实验结果表明,本文提出的改进方法在检测准确率上有较大的提升,达到了较为先进的性能。

### 4.2 改进的 RetinaNet 骨髓血细胞检测网络

本章提出的改进 RetinaNet 网络结构如图 4.2 所示,整体网络结构基于第三章的 RetinaNet 基线模型进行改进。骨干网络为 ResNet50 用于图像特征提取,特征金字塔结构用于多尺度特征提取。锚框的尺寸、数量与分类回归网络结果与基线模型相同。为了提高网络检测的精度,我们在特征金字塔后引入了自底向上的路径聚合模块,该模块基于全局注意力将更浅层的特征与 FPN 深层的进行融合,提升网络定位特征的表达能力,此外我们引入了 IOU 预测分支,将检测框的定位质

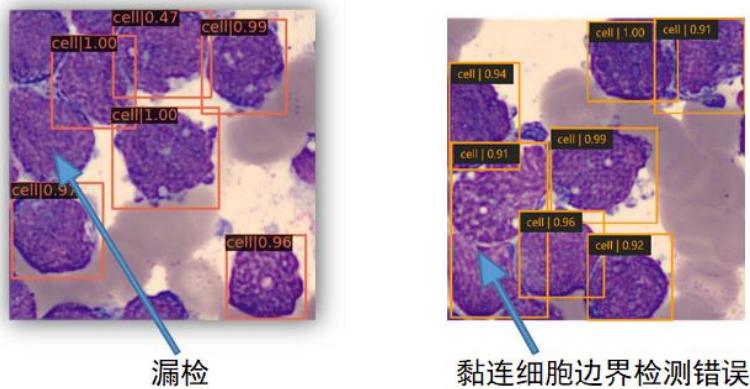


图 4.1 RetinaNet 基线模型检测错误示例

量也纳入到候选框的筛选中。在训练过程中，我们使用基于最优输运的策略进行标签分配。下面各个小节将详细介绍我们的改进点。

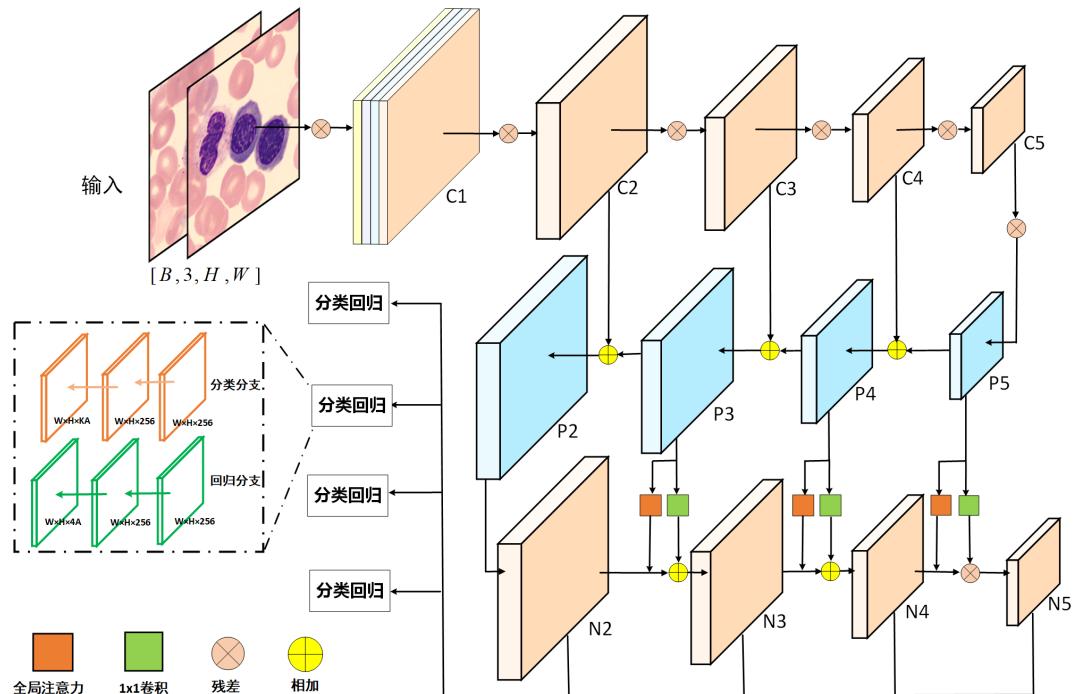


图 4.2 改进的 RetinaNet 网络结构示意图

#### 4.2.1 基于全局注意力的路径聚合网络

ResNet50 骨干网络提取了不同层级与尺度的特征图，高层次的特征图表达目标的抽象语义信息，低层次的特征图表达局部的纹理与模式信息，因此引入了特征金字塔网络，增加自上而下的路径来传播语义强的特征，从而增强所有层次特征的分类能力。在骨髓血细胞检测任务中，我们更加关注网络对于血细胞定位的准确性，这些定位信息主要存在浅层特征图的边缘、纹理信息中。我们构建了自

底向上的路径聚合网络，将浅层的特征与特征金字塔深层的特征图进行融合，通过特征直连缩短了底层到顶层特征之间的信息传递路径。原始结构中底层到顶层需要约 50 层网络，如图 4.3 红线所示。自下而上的路径聚合网络引入了特征直连，从底层到顶层的路径只有不到 10 层，如图 4.3 绿线所示，该路径使得浅层的纹理等高分辨定位信息可以更有效的传递到顶层，提升网络定位特征的表达能力。

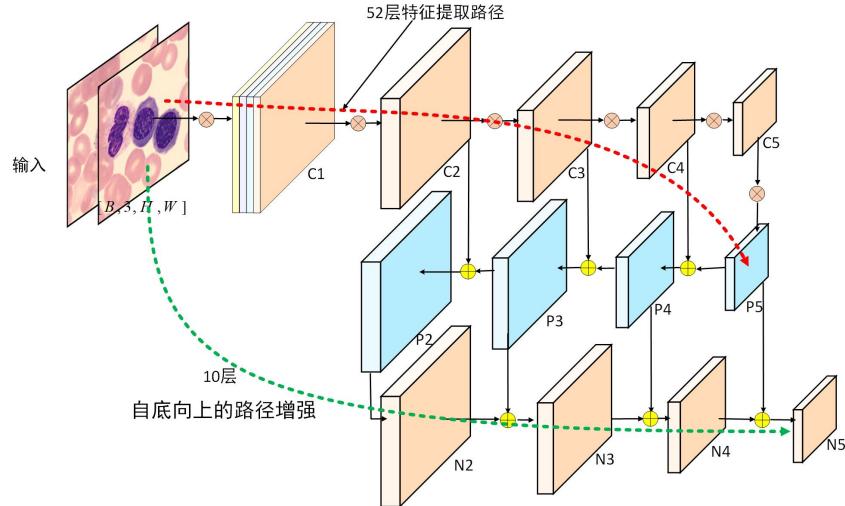


图 4.3 路径聚合网络结构示意图

图中  $C_2, C_3, C_4, C_5$  为 ResNet50 骨干网络不同阶段生成的特征图， $P_2, P_3, P_4, P_5$  为特征金字塔生成的不同级别的特征图。自底向上的路径聚合网络从最底层的  $P_2$  特征图开始，通过步长为 2 的卷积进行下采样并与高层的特征融合后生成新的特征图  $N_2, N_3, N_4, N_5$ 。路径构建模块中使用了全局通道注意力机制<sup>[59]</sup>，使用全局上下文信息的高层次特征指导浅层特征的筛选，该结构如图 4.4 所示。

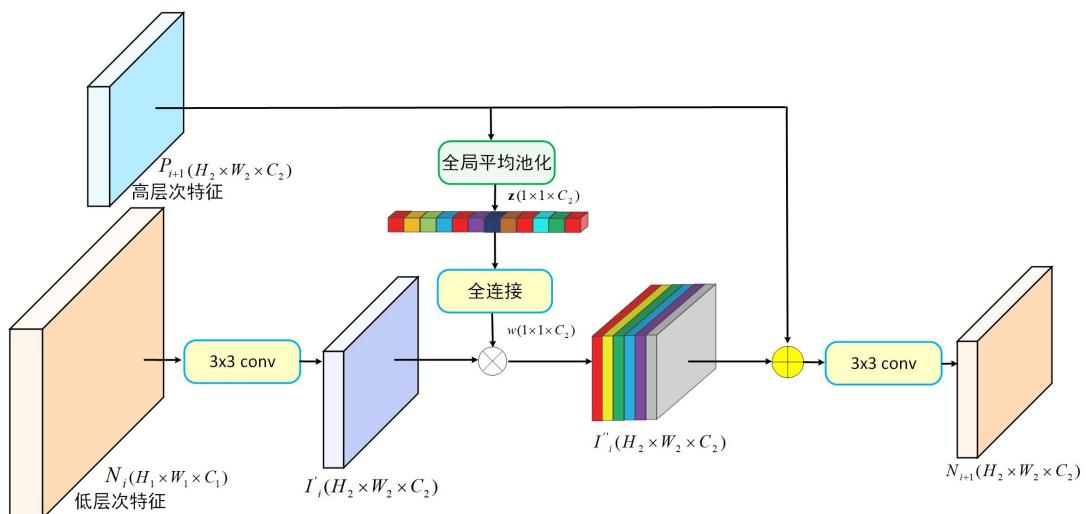


图 4.4 全局注意力模块

全局注意力模块的输入分别是路径聚合网络的低层级特征图  $N_i(H_1 \times W_1 \times C_1)$

与特征金字塔的高层级特征图  $P_{i+1}(H_2 \times W_2 \times C_2)$ , 输出新的特征图  $N_{i+1}$ 。首先对特征图  $N_i$  经过  $3 \times 3$ 、步长为 2 的卷积层降低特征图的尺寸得到  $I'_i(H_2 \times W_2 \times C_2)$ 。对高层级的特征图  $P_{i+1}$  进行全局平均池化, 每个通道压缩为一个值得到  $C$  维向量  $\mathbf{z}(1 \times 1 \times C_2)$ , 如式 4.1 所示:

$$\mathbf{z} = \mathbf{F}_{sq}(\mathbf{P}_{i+1}) = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W \mathbf{P}_{i+1}(i, j) \quad (4.1)$$

然后使用两层全连接结构来全局建模通道之间的依赖关系, 第一层全连接的输出使用 ReLU 激活函数, 第二层使用 Sigmoid 激活函数, 得到权重  $\mathbf{w}(1 \times 1 \times C_2)$ , 如式 4.2 所示。

$$\mathbf{w} = \mathbf{F}_{ex}(\mathbf{z}, \mathbf{W}) = \sigma(g(\mathbf{z}, \mathbf{W})) = \sigma(\mathbf{W}_2 \delta(\mathbf{W}_1 \mathbf{z})) \quad (4.2)$$

将权重向量  $\mathbf{w}$  与特征  $I'_i(H_2 \times W_2 \times C_2)$  按通道点乘得到加权后的特征  $I''_i(H_2 \times W_2 \times C_2)$ , 如式 4.3 所示。

$$\mathbf{I}''_i = \mathbf{F}_{scale}(\mathbf{I}'_i, \mathbf{w}) = w_c I'_{ic} \quad (4.3)$$

将  $\mathbf{I}''_i$  与  $P_{i+1}$  逐元素相加后再经过一个  $3 \times 3$  卷积层后得到路径聚合网络的下一层特征图  $N_{i+1}$ 。不断迭代上述过程, 直到生成  $N_5$  特征图为止。最终在融合后新的特征图  $N_2, N_3, N_4, N_5$  上进行区域提取与坐标回归。anchor 生成与分类回归结构与 Retinanet 相同, 在第三章已进行详细解释。

#### 4.2.2 IOU 预测分支

RetinaNet 在预测阶段会生成密集的检测框, 这些检测框会按照置信度高低进行非极大值抑制 (NMS), 去除重复的检测框。上述 NMS 方式默认了一种假设, 就是置信度高的锚框定位也会更加精确。但是部分细胞例如杆状核细胞不服从中心

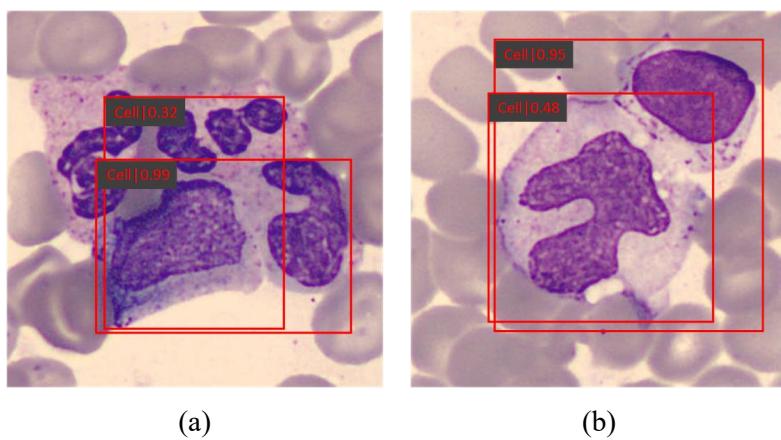


图 4.5 置信度高但交并比低的错误检测示例

分布，因此分类与回归这两种任务不一定严格正相关<sup>[60]</sup>，如图4.5所示，图中(a)无论置信度分数高低，血细胞检测框的坐标都不准确，(b)中置信度分数更高的检测框右侧与上侧的边界都是不准确的，而置信度较低的检测框边界正确。

我们认为需要将检测框的定位质量也纳入到非极大值抑制的考量当中，在挑选分数最大的检测框时同时考虑置信度与交并比。但是在预测阶段，没有目标真实的坐标信息，因此无法使用交并比来判断每个检测框定位质量的好坏。我们在网络的定位部分额外扩展出了一个子分支来预测每一个锚框可能对应真实框的交并比。该分支与定位分支共享特征图信息，使用一个卷积层对于每个锚框输出一个标量值，然后使用Sigmoid激活函数进行激活去得到一个零一之间的交并比信息，修改后的网络结构如图4.6所示。图中(a)为RetinaNet的分类回归分支网络结构。图(b)中在回归分支加入了一个额外的卷积层，去预测交并比信息。

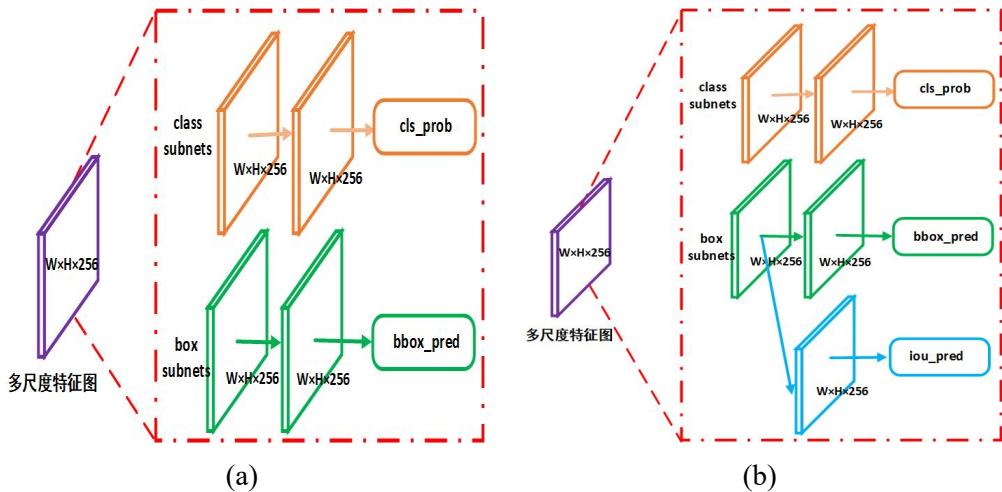


图4.6 交并比预测分支结构

在训练过程，我们优化的目标有分类，坐标回归与交并比，损失函数如式4.4所示，其中  $L_{IoU}(iou_i, iou_i^*)$  定义为预测 IoU 和真实 IoU 之间的二元交叉熵损失。

$$Loss = \sum_i L_{cls}(p_i, p_i^*) + \lambda_1 \sum_i p_i^* L_{reg}(t_i, t_i^*) + \lambda_2 \sum_i L_{IoU}(iou_i, iou_i^*) \quad (4.4)$$

#### 4.2.3 训练标签分配策略

在目标检测网络训练过程中，标签分配（Label Assignment）是非常重要的一个流程。其目的是将训练样本划分为正负样本，并分配分类与回归的目标，计算其与真值之间的损失来监督训练。标签分配方式为网络训练提供了判别性的监督信号，决定了网络学习与收敛的方向，直接影响模型性能的好坏。本节介绍了我们训练过程中采用的不同标签分配策略，根据样本标签分配的正负权重设计，可以将这些方法划分为硬标签分配方法与软标签分配方法。

#### 4.2.3.1 硬标签分配策略

硬标签分配策略假设每个锚框非正即负，若  $w_{pos}, w_{neg}$  分别表示样本属于正负样本的权重，则硬标签划分可以表示为  $w_{pos}, w_{neg} \in \{0, 1\}$  且  $w_{pos} + w_{neg} = 1$ 。这类方法的核心思想是找到一个最优划分边界，将锚框分割为正样本集合与负样本集合。边界划分规则可以分为静态规则与动态规则这两类。

#### 4.2.3.2 最大交并比

RetinaNet 与 Faster-RCNN 等网络采用的是最常用的基于交并比最大化的标签分配策略（MaxIoUAssigner）。该方法有以下几个步骤：(1) 初始化，将正样本集合  $P$ 、负样本集合  $N$  设置为空集，将所有锚框设置为忽略样本；(2) 计算多尺度特征图上所有锚框与所有真实框之间的交并比；(3) 获取每个锚框交并比最大的 GT 框，如果交并比大于正样本阈值（ $pos\_thres$ ），则设置为正样本。如果小于负样本阈值（ $neg\_thres$ ），则设置为负样本。(4) 如果 GT 框没有被锚框匹配到，则获得与 GT 框 IOU 最大的锚框，如果大于最小的正样本阈值，则将该锚框设置为正样本。

最大交并比标签分配策略是静态预先定义的标签分配方式。骨髓血细胞形状多变，比如杆状核粒细胞，通常呈现出 U 型，这导致目标的中心点通常为背景，并不能代表这个目标，而按照最大交并比的方式会被判定为正样本，导致网络的性能较差。

#### 4.2.3.3 自适应样本选择

自适应样本选择（Adaptive Training Sample Selection, ATSS）方法<sup>[61]</sup>基于 L2 距离与交并比动态计算分割阈值，是一种自适应划分目标正负样本的标签分配策略。具体步骤如下：(1) 对于网络输出的多个不同尺度特征图，在每个特征图上计算锚框中心坐标与目标中心坐标的  $L_2$  距离，选取  $K$  个  $L_2$  距离最小的锚框作为候选的正样本，如果有  $L$  个层级的特征图，那么可以得到  $K \times L$  个候选正样本。(2) 计算每个候选正样本与目标真实框的交并比，得到一组交并比的数据。计算这组交并比的均值  $m_g$  与标准差  $v_g$ ，将均值与标准差相加，得到交并比的分割阈值  $t_g = m_g + v_g$ 。(3) 在每个层级的特征图的候选正样本中根据阈值，选择真正的正样本加入正样本集合中进行训练。

图 4.7 为 ATSS 自适应计算阈值的示意图，柱状图的横坐标表示不同层级特征图，纵轴为交并比。柱子上的数字表示这个层级特征图上目标与最近  $K$  个锚框交并比的均值。均值  $m_g$  代表锚框与目标真实框的匹配程度，如果  $m_g$  比较大，则候

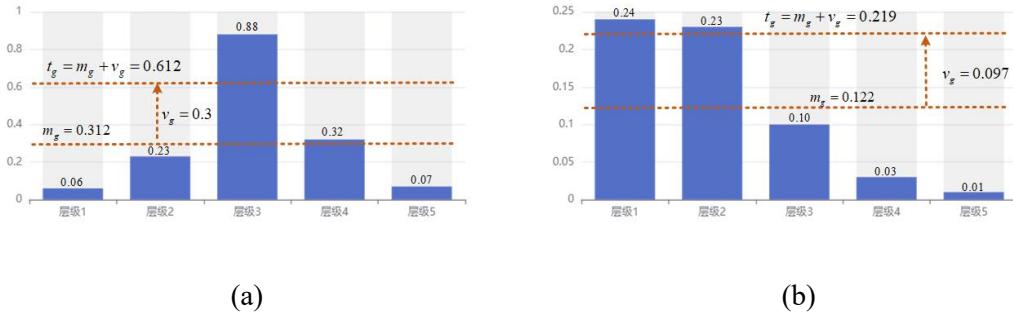


图 4.7 自适应样本选择阈值计算示意图

选正样本的质量都很好，可以适当提高阈值来挑选更好正样本。均值低则应当降低分割阈值。标准差  $v_g$  表示特征层次锚框与目标真实框的匹配程度。如果标准差比较高，则高质量的锚框集中在某一个层级的特征图中，应该提高阈值从最匹配的层级去挑选正样本。标准差比较低，说明所有层级的锚框匹配度都比较高，可以设置一个比较低的阈值广泛的从多个层级的特征图中选取。

#### 4.2.3.4 概率标签分配

概率标签分配 (Probabilistic Anchor Assignment, PAA)<sup>[62]</sup> 锚框的得分数视为概率分布，并通过最大似然来估计分布参数，然后自适应的计算分割阈值，来选取正负样本。锚框的分数用来衡量其与真实框的相似性，包括了分类得分与定位得分。分类得分是分类分支输出的置信度  $P_i(\text{cls}|\theta)$ ，定位得分为预测框与真实框之间的交并比  $\text{IOU}(f(a|\theta), g)$ ，为平衡这两种得分的权重，引入  $\lambda$  参数。锚框得分如式 4.5 所示：

$$S = P_i(\text{cls}|\theta) \times \text{IOU}(f(a|\theta), g)^\lambda \quad (4.5)$$

锚框可以分为正样本、负样本两组，因此可以使用双峰混合高斯分布来建模锚框的分数分布，如式 4.6 所示。

$$P(a|g, \theta) = \phi_1 N(a; \mu_1, \sigma_1) + \phi_2 N(a; \mu_2, \sigma_2) \quad (4.6)$$

然后根据最大期望算法 (Expectation-Maximization, EM) 使得似然最大化。首先对参数  $\phi_i, \mu_i, \sigma_i$  进行随机初始化，在 E 步中计算 Q 函数如式 4.7 所示，式中  $x_1, x_2, \dots, x_n$  为锚框的得分。

$$Q_i(z^{(i)} = k) = \frac{\phi_k \cdot \frac{1}{\sqrt{2\pi}\sigma_k} \exp\left[-\frac{(x^{(i)} - \mu_k)^2}{2\sigma_k^2}\right]}{\sum_{k=1}^K \phi_k \cdot \frac{1}{\sqrt{2\pi}\sigma_k} \exp\left[-\frac{(x^{(i)} - \mu_k)^2}{2\sigma_k^2}\right]} \quad (4.7)$$

在M步中根据式4.8计算混合高斯分布的参数 $\phi_1, \phi_2, \mu_1, \sigma_1, \mu_2, \sigma_2$ 。

概率标签分配的具体步骤如下：对于每个真实框，根据网络输出的置信度与坐标计算每个锚框的得分，在每个尺度的特征度选择K个得分最高的锚框。采用最大期望算法去估计这组锚框的双峰高斯分布参数，找到两个最高峰对应横坐标的分数 $s_1, s_2$ ，其中 $s_1 < s_2$ 。将得分低于 $s_1$ 的锚框作为负样本。得分位于 $s_1$ 与 $s_2$ 之间的锚框划分为忽略样本，得分大于 $s_2$ 的锚框作为正样本。最后那些没有参与分配的锚框均视为负样本。

$$\begin{aligned}\mu_k &= \frac{\sum_{i=1}^N Q_k^{(i)} x^{(i)}}{N_k} \\ \sigma_k &= \frac{\sum_{i=1}^N Q_k^{(i)} (x^{(i)} - \mu_{(k)}) (x^{(i)} - \mu_k)^T}{N_k} \\ \phi_k &= \frac{\sum_{i=1}^N Q_k^{(i)}}{N} \\ N_k &= \sum Q_k^N Q_k^{(i)}\end{aligned}\quad (4.8)$$

#### 4.2.3.5 最优运输标签分配

最优运输<sup>[63]</sup>标签分配将目标检测中的标签分配问题建模为将标签从真实框输送到锚框代价最小的最优运输策略（Optimal Transport）问题，如图4.8所示。

最优运输问题的定义如下：假设有m个供应商与n个需求方，其中第*i*个供应商有 $a_i$ 单元的货物，第*j*个需求方需要 $b_j$ 个单元的货物。每个单元的货物从第*i*个供应商运输到第*j*个需求方的运输代价是 $C_{ij}$ 。最优运输的目标是找到一种运输计划 $\mathbf{P}^* = \{P_{i,j} | i = 1, 2, \dots, m, j = 1, 2, \dots, n\}$ 将供应商的货物全部运输到需求方，并使得运输的代价最小，其中 $P_{i,j}$ 表示第*i*个供应商运输到第*j*个需求方单元货物的数量。

最优运输的优化目标如式4.9所示：

$$\min_{\mathbf{P}} \sum_{i=1}^m \sum_{j=1}^n C_{ij} P_{ij} \quad P_{ij} \geq 0, i = 1, 2, \dots, m, j = 1, 2, \dots, n. \quad (4.9)$$

其中第*i*个供应商输出的数量等于 $s_i$ ，即 $\sum_{j=1}^n P_{ij} = a_i$ 。第*j*个需求方接收的数量等于 $b_j$ ，即 $\sum_{i=1}^m P_{ij} = b_j$ 。供应商供应的数量等于需求方接收的数量 $\sum_{i=1}^m a_i = \sum_{j=1}^n b_j$ 。

最优运输问题可以在多项式的时间复杂度内，通过Sinkhorn快速迭代的方法进行求解。该方法的思想源于交叉熵与对偶问题，定义 $\mathbf{H}(\mathbf{P}) \stackrel{\text{def.}}{=} -\sum_{i,j} \mathbf{P}_{i,j} (\log(\mathbf{P}_{i,j}) - 1)$ 上述问题添加了拉格朗日算子和扰动项的形式如

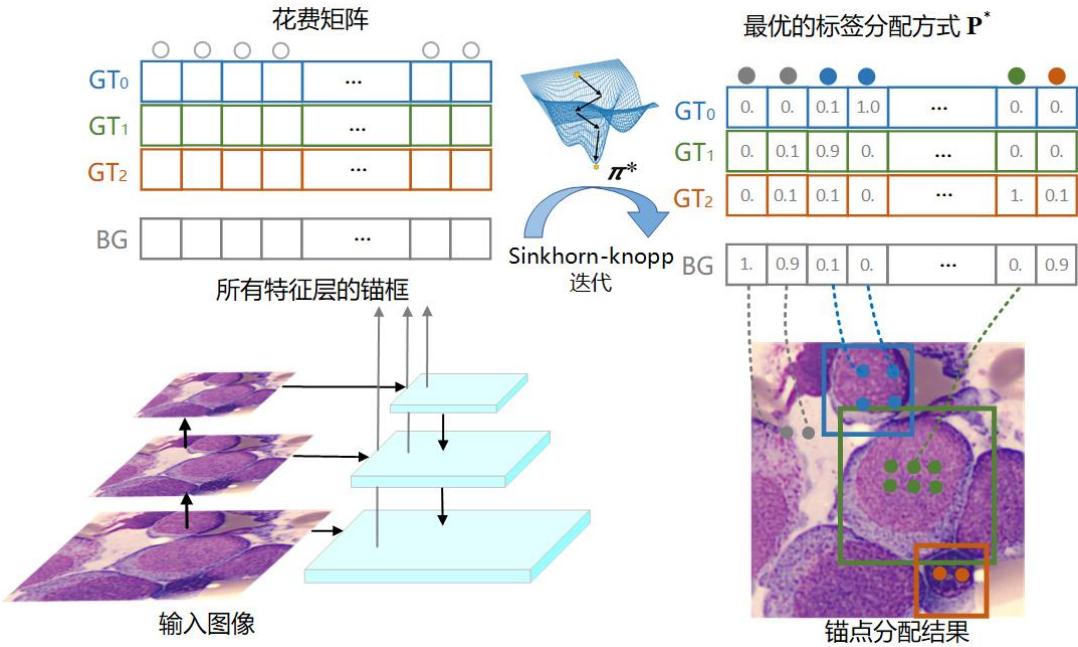


图 4.8 基于最优输运的标签分配策略

式 4.10 所示：

$$E(\mathbf{P}, \mathbf{f}, \mathbf{g}) = \sum_{i=1}^m \sum_{j=1}^n C_{ij} P_{ij} - \varepsilon H(\mathbf{P}) + f_j \left( \sum_{i=1}^m P_{ij} - b_j \right) + g_i \left( \sum_{j=1}^n P_{ij} - a_i \right) \quad (4.10)$$

其中  $\varepsilon$  是一个常量超参数，控制熵正则化项的强度，实验中设置为 0.1， $f_j (j = 1, 2, \dots, n)$  与  $g_i (i = 1, 2, \dots, m)$  是拉格朗日乘子。通过使得目标函数导数等于零求得最优的输运策略如式 4.11 所示：

$$\frac{\partial E(\mathbf{P}, \mathbf{f}, \mathbf{g})}{\partial P_{ij}} = C_{ij} + \varepsilon \log(P_{ij}) - f_j - g_i = 0 \quad (4.11)$$

因此最优输运策略为  $\mathbf{P}^*$  如式 4.12 所示：

$$P_{ij}^* = \exp\left(-\frac{f_j}{\varepsilon}\right) \exp\left(-\frac{C_{ij}}{\varepsilon}\right) \exp\left(-\frac{g_i}{\varepsilon}\right) \quad (4.12)$$

令  $s_j = \exp\left(-\frac{f_j}{\varepsilon}\right)$ ,  $U_{ij} = \exp\left(-\frac{C_{ij}}{\varepsilon}\right)$ ,  $t_i = \exp\left(-\frac{g_i}{\varepsilon}\right)$  上述变量满足如式 4.13 的约束关系：

$$\begin{aligned} \sum_i P_{ij} &= s_j (\sum_i U_{ij} t_i) = b_j \\ \sum_j P_{ij} &= (s_j \sum_i U_{ij}) t_i = a_i \end{aligned} \quad (4.13)$$

上式的约束关系需要同时满足，因此  $t_i$  与  $s_j$  的交替迭代公式如下：

$$s_j^{l+1} = \frac{b_j}{\sum_i U_{ij} t_i^l}, \quad t_i^{l+1} = \frac{a_i}{\sum_j U_{ij} s_j^{l+1}} \quad (4.14)$$

在目标检测的背景下，假设一张图片中有  $m$  个真实框，检测网络所有的 FPN 层总共输出了  $n$  个锚框。将真实框视为供应商，可以为  $k$  个锚框提供正样本的标签，等价于有  $k$  个单元的货物 i.e.,  $a_i = k, i = 1, 2, \dots, m$ 。每个锚框视为需求方，需要一个标签 i.e.,  $b_j = 1, j = 1, 2, \dots, n$ 。此外引入一个背景供应商，来为锚框提供负样本标签，数量为  $n - m \times k$ 。

将正样本标签从某一个真实框  $gt_i$  输运到一个锚框  $anchor_j$  的代价定义为分类与回归的加权损失，如式 4.15 所示：

$$C_{ij}^{fg} = L_{cls} \left( p_i^*, p_j^{cls}(\theta) \right) + \lambda L_{reg} \left( t_i^*, t_j^{box}(\theta) \right) \quad (4.15)$$

其中  $\theta$  代表模型参数， $p_j^{cls}(\theta)$  与  $t_j^{box}(\theta)$  分别代表模型预测的分类得分与坐标回归值。 $p_i^*$  与  $t_i^*$  代表了真实类别与坐标。 $L_{cls}$  与  $L_{reg}$  为交叉熵损失函数与 GIOU 损失函数。

背景供应商将一个负样本标签传递到锚框的花费为分类损失，如下式所示：

$$C_j^{bg} = L_{cls} \left( \Phi, p_j^{cls}(\theta) \right) \quad (4.16)$$

在定义好花费矩阵后，最优输运的策略  $\mathbf{P}^*$  可以通过 Sinkhorn-Knopp 迭代算法进行求解。每个 anchor 的标签为最大的标签所对应的供应商类别。算法流程如算法 4.1 所示。

## 4.3 算法实现与实验结果分析

### 4.3.1 实验环境

训练平台使用 linux，操作系统为 ubuntu20.04，GPU 为 NVIDIA RTX3090，CPU 为 Intel(R) Core(TM) i9-10940X。深度学习框架为 Pytorch1.13.1，训练的批量大小设置为 16。使用 SGD 随机梯度下降优化器对网络参数进行更新，动量设置为 0.9，权重衰减设置为 5e-4。初始化的学习率为 0.001。网络总共训练 36 个轮次，在第 16 与第 28 个轮次，学习率变为原来的 1/10。RetianNet 中 Focal Loss 中正负样本平衡参数  $\alpha = 0.25$ ，难易调整参数  $\gamma = 2$ 。主干网络使用在 coco 数据集上预训练的 ResNet50 网络。

#### 4.3.1.1 数据集介绍

骨髓血细胞图像来自邃蓝智能科技（上海）有限公司合作医院提供，首先采用第 2.1 小节阐述的主动学习标注策略进行边界框的标注。我们总共标记了 6821 张血细胞图像，训练集与测试集按照 4:1 的比例进行随机划分，训练集包含了 5456 张图像，测试集包含了 1365 张图像。通常每个图像中包含 1 到 10 个有核血细胞，

**算法4.1 最优输运标签分配****输入:**

- $I$ : 输入图像  
 $A$ : 网络输出的一组锚框  
 $G$ : 图像中目标真实框标注  
 $\epsilon$ : Sinkhorn-Knopp 迭代熵正则化项  
 $T$ : Sinkhorn-Knopp 迭代次数  
 $\lambda$ : 式 4.15 中的平衡因子

**输出:**

- $\mathbf{P}^*$  标签最优输运策略
- 1: set  $m = |G|, n = |A|$ ;
  - 2: 网络前向计算每个锚框的分数与坐标  $P^{cls}, P^{box}$
  - 3: 动态计算每个 GT 框的  $a_i$
  - 4: 计算得到背景供应商的标签数量  $a_{m+1} = n - \sum_{i=1}^m a_i$
  - 5:  $b_j (j = 1, 2 \dots n)$  使用 1 初始化
  - 6: 前景分类损失  $C_{ij}^{cls} = FocalLoss(P_j^{cls}, G_i^{cls})$
  - 7: 前景回归损失  $C_{ij}^{reg} = IoULoss(P_j^{box}, G_i^{box})$
  - 8: 背景花费  $C_{bg}^{cls} = FocalLoss(P_j^{cls}, \Phi)$
  - 9: 前景花费  $C_{fg} = C^{cls} + \lambda C^{reg}$
  - 10: 计算最终的花费矩阵将  $C_{bg}$  拼接到  $C_{fg}$  的最后一行
  - 11: 对  $s^0, t^0$  进行随机初始化
  - 12: **for**  $i = 1; i < T; i++$  **do**
  - 13:   计算  $s^{l+1}, t^{l+1} \leftarrow \text{SinkhornIter}(C, s^l, t^l, a, b)$
  - 14: **end for**
  - 15: 根据式 4.12, 计算并返回最优的标签分配策略  $\mathbf{P}^*$

数据集总共标记了 11352 个血细胞, 训练集有 9065 个血细胞, 测试集有 2287 个血细胞。数据集的分布如表 4.1 所示:

### 4.3.2 实验结果与分析

#### 4.3.2.1 评价指标

在 3.4.2.3 节中, 我们提出了先检测、再识别的骨髓血细胞计算处理流程。在本节中, 我们只关注血细胞检测的准确性, 网络性能评估指标采用准确率、召回率、平均精度均值 (mAP) 与 PR 曲线, 具体定义见 3.4.1.2 节。

#### 4.3.2.2 实验结果

为了验证本节提出的改进 RetinaNet 方法在骨髓血细胞检测上的有效性, 我们在上节所提的骨髓血细胞数据集上进行测试, 在不同交并比阈值条件下, 网络的准确率、召回率、f1-score、平均检测精度均值与 FPS 如表 4.2 所示, 其中准确率、召回率为 f1-score 最大条件下对应的数值, FPS 是网络每秒可计算的图像帧数量。

表 4.1 骨髓血细胞检测数据集分布

序号	类别名	类别简写	训练集数量	测试集数量
1	原始细胞	Prim	1856	467
2	淋巴细胞	Lym	996	226
3	单核细胞	Mono	206	52
4	浆细胞	Plas	272	70
5	红细胞	Red	1880	503
6	早幼粒细胞	Promy	357	107
7	嗜中性中幼粒细胞	Myelo	701	150
8	嗜中性晚幼粒细胞	Late	503	144
9	嗜中性杆状核细胞	Rods	998	241
10	嗜中性分叶核细胞	Lobu	821	195
11	嗜酸性粒细胞	Eosl	475	132
总计			9065	2287

从表中我们可以看到，本节提出的改进模型效果较好，在交并比为 0.75 的条件下，f1-score 为 0.9474，召回率为 0.950，准确率为 0.9448。随着交并比的逐渐增大，网络的准确率与召回率均开始下降，在交并比为 0.95 时，下降最为明显，平均检测精度只有 0.176。因此，对于高精度的定位 (IOU>0.9)，模型检测性能仍有待提升，我们认为未来可以在数据量上进行扩充，此外在标注上进行更加精细化的校对。

此外，我们对比了 SSD512<sup>[50]</sup>、Faster-RCNN<sup>[17]</sup>、Cascade-RCNN<sup>[57]</sup>、RetinaNet<sup>[19]</sup>、YOLOV3<sup>[58]</sup>与 DETR<sup>[21]</sup>等六种基于深度学习的主流目标检测网络，这些网络均使用在 COCO 数据集上预训练模型进行参数初始化，然后基于迁移学习在骨髓血细胞检测数据集上进行微调。表 4.3 为不同网络的平均检测精度对比，其中  $AP_{50}$  表示在交并比为 0.5 条件下的平均检测精度。

从表 4.3 中我们可以看到，SSD512 网络检测精度最差，这是由于 VGG16 骨干网络特征表达能力相比于 ResNet 等残差网络更差。在交并比为 0.5 的条件下，DETR 网络的平均精度最高，本节提出的网络略低于 DETR 方法。在交并比为 0.75 与 0.9 的条件下，本节方法平均检测精度均高于其他对比方法，特别是在交并比为 0.9 的条件下，本节方法相比于次优的 Cascade-RCNN 方法提升了约 6.93%。该结果说明 Cascade-RCNN 通过级联 RPN 网络的方式，可以有效的提升候选框的定位质量，而本文提出的改进点将 IOU 交并比纳入到检测网络的推理阶段，也可以帮

表 4.2 改进 RetinaNet 骨髓血细胞检测结果

IOU	准确率	召回率	f1-score	mAP	FPS
0.5	0.9624	0.96	0.9612	0.9864	27.2
0.55	0.9594	0.96	0.9597	0.9837	27.2
0.6	0.9594	0.96	0.9597	0.9837	27.2
0.65	0.9579	0.96	0.9589	0.9821	27.2
0.7	0.9559	0.95	0.9529	0.9735	27.2
0.75	0.9448	0.95	0.9474	0.9668	27.2
0.8	0.931	0.92	0.9255	0.9328	27.2
0.85	0.8738	0.86	0.8668	0.8463	27.2
0.9	0.7256	0.72	0.7228	0.6305	27.2
0.95	0.3537	0.33	0.3414	0.1766	27.2

助网络找到定位精度更好的候选框。在网络的计算量与参数量方面，DETR 的计算量最低，SSD512 的参数量最少，本文提出的改进方法相比于基础 RetinaNet 方法计算量增加了 3.8%，参数量增加了 1.58MB。总而言之，本节提出的改进 RetinaNet 方法在骨髓血细胞的检测精度上要优于其他方法，交并比为 0.75 条件下的平均检测精度提升了 1.26%。

本节提出的改进模型在骨髓血细胞数据集上检测的可视化结果如图 4.9 所示，图中将置信度阈值大于 0.5 的骨髓血细胞以红色检测框标出，左上角的数字为置信度分数，从图中我们可以看到，该网络可以有效的对密集与黏连的血细胞区域进行检测。

表 4.3 不同目标检测方法在骨髓血细胞测试集上的性能对比

方法	主干网络	$AP_{50}$	$AP_{75}$	$AP_{90}$	参数量大小 (MB)	运算次数 (GFLOPs)
SSD512	VGG16	0.9748	0.9178	0.3234	36.04	250.92
Faster-RCNN	ResNet50	0.9819	0.9458	0.4965	41.17	139.25
Cascade-RCNN	ResNet50	0.9806	0.9556	0.5896	69.17	167.24
RetinaNet	ResNet50	0.9801	0.9542	0.5215	36.31	135.73
YOLOV3	DarkNet53	0.9794	0.9508	0.4646	61.95	127.11
DETR	ResNet50	0.9888	0.9559	0.5844	41.30	59.82
本节提出的网络	ResNet50	0.9864	0.9668	0.6305	37.89	140.94

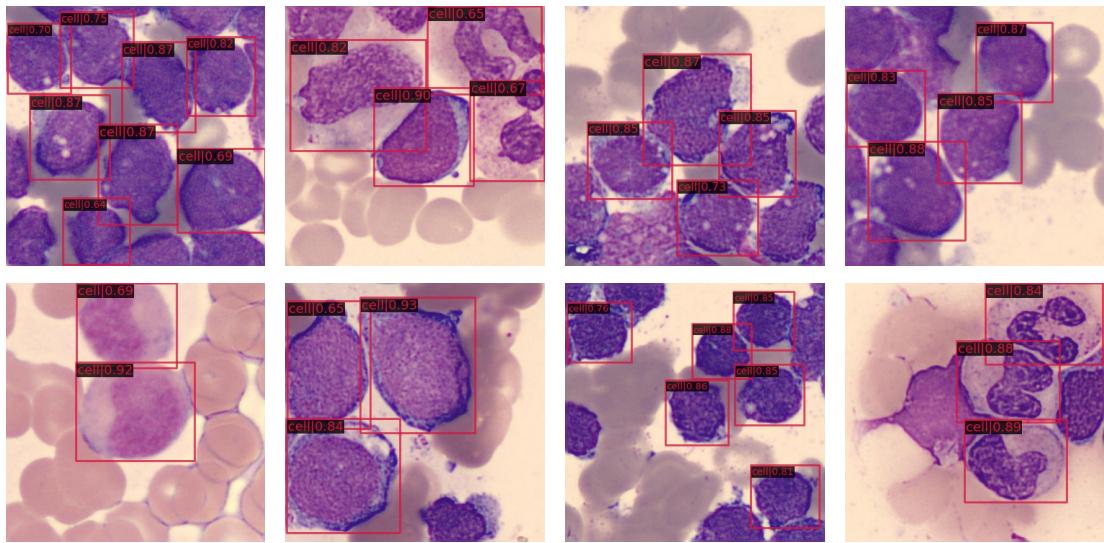


图 4.9 骨髓血细胞数据集上的检测结果

#### 4.3.2.3 改进网络结构

为了验证本节提出的全局注意力路径聚合网络模块与 IOU 预测分支网络模块的有效性，我们分别在基线模型上添加单一模块来验证这两种模块的有效性。不同模型在骨髓血细胞测试集的结果如表 4.4 所示， $AP_{50}$ 、 $AP_{75}$  与  $AP_{90}$  分别表示交并比为 0.5、0.75 与 0.9 条件下的平均检测精度。 $\checkmark$  代表网络添加该模块， $\times$  代表网络不包含该模块，上述模型均使用最大交并比（MaxIOU）的标签分配策略。

表 4.4 骨髓血细胞检测网络模块消融实验

主干网络	路径聚合模块	IOU 预测模块	$AP_{50}$	$AP_{75}$	$AP_{90}$
RetinaNet	$\times$	$\times$	0.9801	0.9542	0.5215
RetinaNet	$\checkmark$	$\times$	0.9893	0.9601	0.5951
RetinaNet	$\times$	$\checkmark$	0.9843	0.9575	0.5831
RetinaNet	$\checkmark$	$\checkmark$	0.9856	0.9629	0.6137

从表 4.4 中，我们可以看到，单独添加路径聚合网络模块，网络在交并比为 0.5 条件下的平均检测精度最高，在交并比为 0.75 与 0.9 时，平均检测精度相比于基线模型提升了 0.59% 与 7.36%。单独添加 IOU 预测模块，在交并比为 0.75 与 0.9 时，平均检测精度分别提升了 0.33% 与 6.16%，实验结果表明在网络结构中引入路径聚合网络与 IOU 预测网络均可以有效提高骨髓血细胞的检测精度。组合两种模块后的网络在交并比为 0.75 条件下的平均检测精度为 0.9629，相比于基线模型提升了 0.87%。在交并比为 0.90 条件下的平均检测精度为 0.6137，相比于基线模型提升了 9.22%。

#### 4.3.2.4 标签分配策略

在本小节，我们对比了不同的标签分配策略对骨髓血细胞检测精度的影响，实验中网络结构使用改进的 RetinaNet 网络，四种标签分配策略分别是最大交并比 (MaxIOU)、自适应样本选择 (ATSS)、概率标签分配 (PAA) 与最优运输标签分配 (OTA)。在骨髓血细胞检测数据集上的结果如表 4.5 所示，表中  $AP_{50}$ 、 $AP_{75}$  与  $AP_{90}$  的定义同表 4.4， $AR_{50}$ 、 $AR_{75}$  与  $AR_{90}$  分别表示置信度阈值大于 0.3、单张图像最大检测框数量为 100、交并比为 0.5、0.75、0.9 条件下的召回率。

表 4.5 标签分配策略消融实验

主干网络	标签分配策略	$AP_{50}$	$AP_{75}$	$AP_{90}$	$AR_{50}$	$AR_{75}$	$AR_{90}$
改进 RetinaNet	MaxIOU	0.9856	0.9629	0.6137	1.00	0.9784	0.7170
改进 RetinaNet	ATSS	0.9845	0.9657	0.6235	1.00	0.9825	0.7377
改进 RetinaNet	PAA	0.9843	0.9656	0.6246	1.00	0.9831	0.7361
改进 RetinaNet	OTA	0.9864	0.9668	0.6305	1.00	0.9841	0.7372

从表中可以看到，ATSS、PAA 与 OTA 的标签分配策略要优于基线模型的 MaxIOU 标签分配策略。在交并比为 0.50 的条件下四种标签策略的方法召回率均为 1.00，平均检测精度 OTA 最高，MaxIOU 次之。在交并比为 0.75 与 0.9 的条件下，OTA 标签分配方法在召回率与平均检测精度优于其他三种标签分配方法，ATSS 与 PAA 方法性能非常接近，MaxIOU 方法效果最差。在交并比为 0.75 时，OTA 标签分配方法相比于基线方法平均分类精度提升了 0.39%，在交并比为 0.9 时，召回率提升了 2.02%。上述实验结果表明，最优运输标签分配策略可以有效避免模糊样本的出现，提高网络的召回能力并进一步提升平均检测精度。

## 4.4 小结

针对密集、黏连骨髓血细胞区域存在漏检、误检等问题，本节提出了一种基于改进 RetinaNet 的骨髓血细胞检测网络。我们在基线模型上总共实现了三个创新改进点，第一点是基于全局注意力的自下而上的路径聚合模块，该模块通过特征直连缩短了底层到顶层特征之间的信息传递路径，使得浅层纹理等高分辨定位信息可以更有效的传递到顶层，该模块使得网络的  $AP_{75}$  提升了 0.59%。第二点，网络引入了 IOU 预测分支，对每个锚框与可能对应真实框交并比进行预测，在推理阶段将检测框的定位质量纳入到非极大值抑制的考量中，该模块使得网络的  $AP_{75}$  提升了 0.33%。第三点，本节探究了不同标签分配策略对检测性能的影响，将最优

输运标签策略用于密集区域血细胞的标签分配，避免了模糊分配样本的出现，提高网络对血细胞的召回能力。在骨髓血细胞测试集上的实验结果表明，本文提出的改进方法相比于基线模型在平均检测精度上提升了1.26%，达到了较为先进的性能。改进网络单张图像的平均检测时间为37ms，可以用于实时自动化骨髓血细胞检测。

## 第5章 基于改进 Vision Transformer 骨髓血细胞检测算法设计与实现

### 5.1 引言

白血病是一种人体造血系统的恶性肿瘤，在所有恶性肿瘤中占比约 5%，是我国重点防治的十大恶性肿瘤之一。血细胞形态学检查是白血病诊断常规检查的一部分，各类骨髓血细胞经过染色后呈现出不同的形状、颜色与纹理，这些细胞由经验丰富的病理专家识别并计数，最终根据 FAB 标准给出白血病类型的诊断。上述人工镜检的诊断流程存在以下不足，人工分类计数繁琐费时，诊断结果具有较强的主观性。此外，细胞形态学人才资源紧缺，培养精通细胞病理诊断的医师要耗费大量的时间。通过研究骨髓血细胞自动化识别技术来辅助临床诊断，可以实现诊断流程的标准化、快速化与智能化，将医生从繁重的病理工作中解放出来，具有重要的临床意义和广阔的应用前景。

近年来，基于深度学习的方法在医学影像处理领域取得了巨大的成功，国内外学者纷纷开始探索基于深度学习的血细胞识别方法。基于深度学习的血细胞识别方法不再需要进行复杂的血细胞特征工程设计，直接将血细胞数据输入网络中进行端到端训练。通过优化损失函数，网络可以自动挖掘数据的潜在特征，并基于这些特征进行预测，相比于传统识别方法具有更高的识别准确率。根据文献调研，血细胞识别算法主要基于计算机视觉领域的目标识别网络并通过迁移学习进行微调，这些识别网络包括了 ResNext、ResNet 与 EfficientNet 等，在血细胞数据集上获得了很好的识别结果。

骨髓血细胞自动化识别是一项非常具有挑战性的任务，主要存在着以下难点：(1) 缺少大规模的骨髓血细胞识别任务数据集，不同批次的骨髓血细胞存在染色、光照等差异，导致切片图像外观、颜色变化的多样性。(2) 各类骨髓血细胞在人体中的所占比例不同，导致数据集中各个类别样本数量分布不均衡，数量较少类别的骨髓血细胞难以有效的进行特征学习。(3) 骨髓血细胞种类繁多，例如粒细胞有原始、早幼、中幼与晚幼等阶段，相邻发育阶段的血细胞在形态上非常类似，骨髓血细胞子类之间差异较小增加了细粒度识别的难度。

针对骨髓血细胞识别过程中的难点，本章提出了一种基于改进 Vision Transformer 的骨髓血细胞识别方法。首先，使用第四章提出的改进 RetinaNet 网络从图像中检测出细胞边界并进行裁剪，去除背景等干扰。接着，本章提出一种重叠图像块划分方法将裁剪后的图像分割为多个图像块并学习嵌入向量表示。然后，嵌

向量经过多个编码层进行特征提取。本章基于多头自注意机制提出了稀疏注意力模块，该模块可以捕捉图像中的辨识性区域，提取图像中的细粒度特征，并将筛选后的特征输入到编码层。最后，网络输出的分类特征用于骨髓血细胞的识别。在训练过程中，本章采用对比损失进一步增加分类特征的类内一致性与类间差异性。最后，在邃蓝智能骨髓血细胞数据集与开源的慕尼黑血细胞形态学数据集实验结果表明，我们提出的方法具有良好的精细分类性能，相比于其他识别方法具有更高的识别准确率。

本章内容源自我以第一作者发表的文章《基于改进 Vision Transformer 的血细胞图像识别方法研究》<sup>[64]</sup>

## 5.2 改进的 Vision Transformer 骨髓血细胞识别网络

本文提出的基于 Vision Transformer 的骨髓血细胞识别网络框架如图 5.1 所示。首先将输入的血细胞图像分割为  $N$  个  $P \times P$  大小的图像块，接着将图像块线性映射为序列化嵌入向量，其次加入可学习的分类向量与位置编码信息。然后嵌入向量被输入到多个堆叠的编码模块中进行特征提取。在最后一层编码模块前，使用辨识性区域选择模块来寻找图像中的区分性像素块并将其对应的隐含特征作为输入。最后编码器输出的分类特征经过全连接层得到骨髓血细胞的类别概率信息。

### 5.2.1 重叠图像块划分

Vision Transformer 模型接收的输入为序列化数据，因此需要将图像划分为图像块并线性映射为序列化向量（token）。Vision Transformer 模型将图像划分成大小为  $P \times P$  且互不重叠的像素块，但这样划分会破坏图像的局部结构，例如辨识性区域被划分到两个相邻的图像块中。为避免该问题，本文采用滑动窗的方法来生成有重叠的图像块。当输入图像的尺寸为  $H \times W \times C$ 、图像块大小为  $P$ 、滑动窗的步长为  $S$  时，图像将会被划分为  $N$  个像素块，其中  $N$  如式 5.1 所示。

$$n = n_h \times n_w = \left\lfloor \frac{h-p}{s} + 1 \right\rfloor \times \left\lfloor \frac{w-p}{s} + 1 \right\rfloor \quad (5.1)$$

通过滑动窗的方式，两个相邻像素块的重叠面积为  $(P-S) \times P$ ，更好的保留了图像的局部信息。当  $S$  越小，局部结构保存的越完整，但会增加序列化向量的数量导致计算开销变大。综合利弊，在实验中将  $S$  的大小设置为  $2P/3$ 。图像划分完成后，需要将 2-D 的图像块转化为 1-D 的序列向量，首先将图像块展平为一组向量  $x_p \in \mathbb{R}^{N \times P^2 C}$ ，然后通过线性变换将其映射到  $D$  的维度大小。上述转化在具体实现上等价于对原图像进行  $D$  个  $P \times P \times P^2 C$  尺寸的卷积核、步长为  $2P/3$  的卷

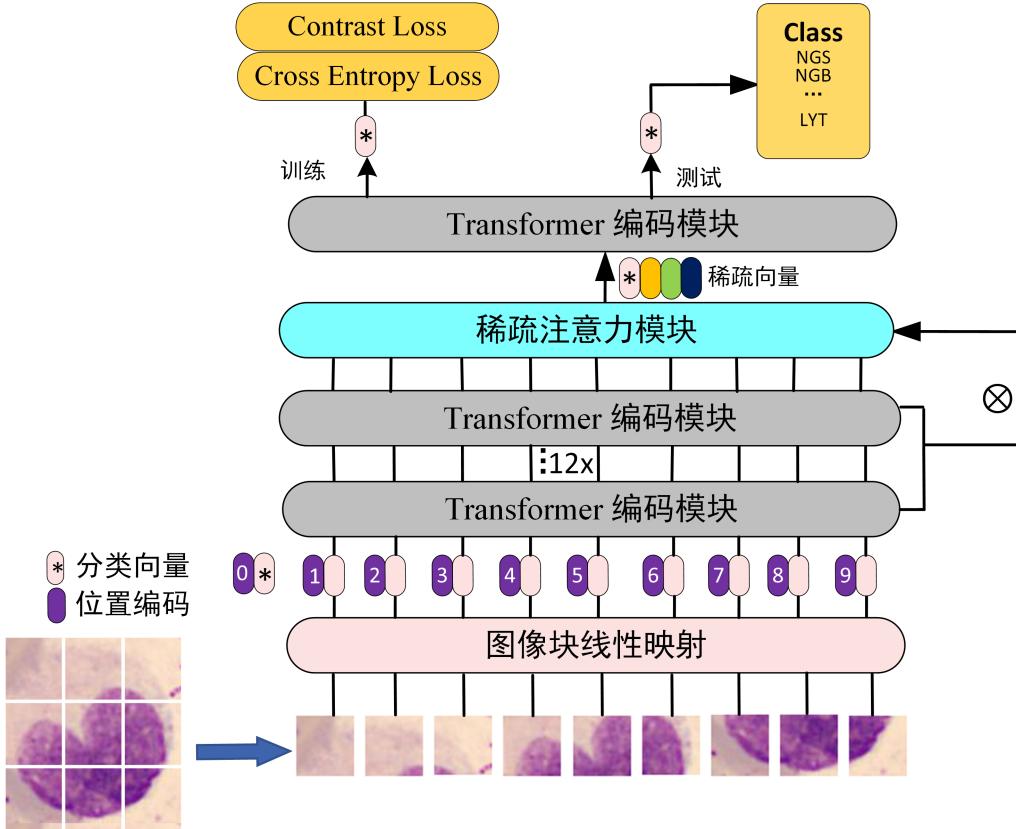


图 5.1 基于改进 Vision Transformer 骨髓血细胞识别网络结构

积操作。由于嵌入后的向量不包含位置信息，需要加入一个特殊的可学习位置编码。此外还加入可学习分类向量作为最终的输出特征用于图像分类。嵌入后的序列数据  $\mathbf{z}_0$  如式 5.2 所示，其中  $\mathbf{E}$  为投影矩阵、 $\mathbf{E}_{\text{pos}}$  为位置编码、 $\mathbf{x}_{\text{class}}$  为分类向量。

$$\mathbf{z}_0 = [\mathbf{x}_{\text{class}}; \mathbf{x}_p^1 \mathbf{E}; \mathbf{x}_p^2 \mathbf{E}; \dots; \mathbf{x}_p^N \mathbf{E}] + \mathbf{E}_{\text{pos}} \quad (5.2)$$

图片序列化之为一组向量后，网络需要知道每一个 token 在序列中的绝对位置，并且不同 token 之间的相对位置也需要保持一致。因此引入位置编码（Positional Embedding, PE）来表示 token 的位置关系。位置编码采用连续有界的正弦、余弦函数来表示位置信息，受到二进制编码的启发，位置编码向量的低位角频率较大，模拟二进制低位 0、1 的高频交互，位置编码高位角频率小，对位置  $t$  的变动不敏感，模拟二进制的高位变化较小。为了可以用线性变换表示相对位置变动，采用正弦与余弦一组的方式对位置编码进行表示。定义  $t$  为这个 token 在序列中的实际位置， $\mathbf{PE}_t \in \mathbb{R}^d$  表示这个 token 的位置编码向量， $\mathbf{PE}_t^{(i)}$  表示位置编码向量中的第  $i$  个元素， $d_{\text{model}}$  为位置编码的维度，则位置编码  $\mathbf{PE}_t \in \mathbb{R}^d$  如式 5.3 所示：

$$\mathbf{PE}_t^{(i)} = \begin{cases} \sin(w_i t), & \text{if } k = 2i \\ \cos(w_i t), & \text{if } k = 2i + 1 \end{cases} \quad (5.3)$$

其中  $w_i = \frac{1}{10000^{2i/d_{\text{model}}}}$ ,  $i = 0, 1, 2, 3, \dots, \frac{d_{\text{model}}}{2} - 1$ 。token 序列长度为 50, 向量维度为 128 的位置编码可视化结果如图 5.2 所示。

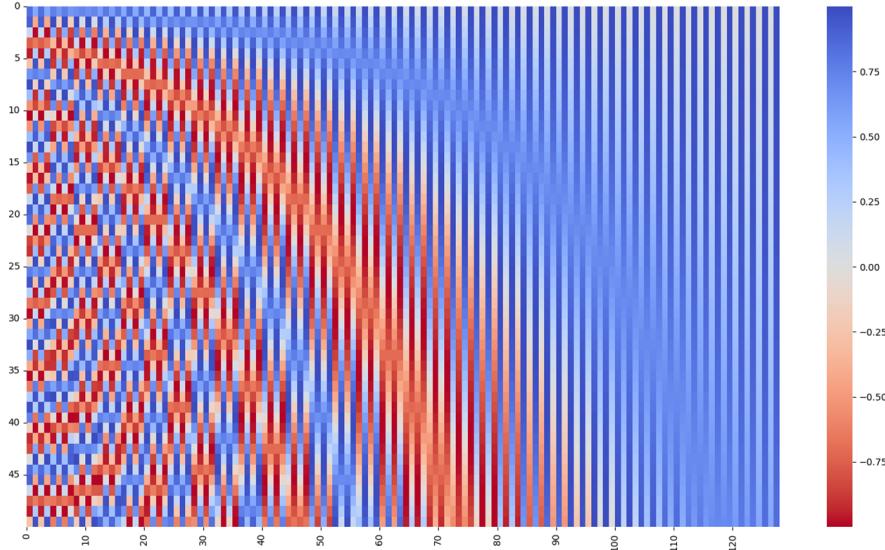


图 5.2 位置编码可视化

### 5.2.2 编码层

Vision Transformer 的编码器由  $L$  个结构相同的编码模块堆叠而成, 编码模块结构如图 5.3 所示:

编码模块包含了多头自注意 (Multi-head Self-Attention, MSA) 与多层次感知机 (Multi-Layer Perceptron, MLP)。多头自注意模块由  $N_h$  个单头自注意单元 (Self-Attention, SA) 组成。对于单头自注意单元, 首先将输入  $\mathbf{z}_p \in \mathbb{R}^{(N+1) \times D}$  经过线性变换得到查询矩阵  $\mathbf{Q}$ 、键矩阵  $\mathbf{K}$ 、值矩阵  $\mathbf{V}$ 。线性变换如式 5.4 所示:

$$\begin{aligned}\mathbf{Q} &= \mathbf{z}_p \cdot \mathbf{W}^Q \quad \mathbf{W}^Q \in \mathbb{R}^{D \times d_k} \\ \mathbf{K} &= \mathbf{z}_p \cdot \mathbf{W}^K \quad \mathbf{W}^K \in \mathbb{R}^{D \times d_k} \\ \mathbf{V} &= \mathbf{z}_p \cdot \mathbf{W}^V \quad \mathbf{W}^V \in \mathbb{R}^{D \times d_k}\end{aligned}\tag{5.4}$$

其中  $d_k = \frac{D}{N_h}$ , 得到  $\mathbf{Q}$ 、 $\mathbf{K}$ 、 $\mathbf{V}$  后, 注意力权重矩阵  $\mathbf{A}$  的计算如式 5.5 所示:

$$\mathbf{A} = \text{softmax} \left( \frac{\mathbf{Q} \cdot \mathbf{K}^T}{\sqrt{d_k}} \right), \quad \mathbf{A} \in \mathbb{R}^{(N+1) \times (N+1)}\tag{5.5}$$

矩阵  $\mathbf{A}$  中的元素  $A_{ij}$  表示第  $i$  个特征与第  $j$  个特征之间的相关性, 值越大则相关性越强,  $\sqrt{d_k}$  是缩放因子。注意力权重矩阵  $\mathbf{A}$  点乘值矩阵  $\mathbf{V}$  得到单头自注意单元的输出  $\mathbf{z}'$ 。

$$\mathbf{z}' = \mathbf{A} \cdot \mathbf{V}, \quad \mathbf{z}' \in \mathbb{R}^{(N+1) \times d_k}\tag{5.6}$$

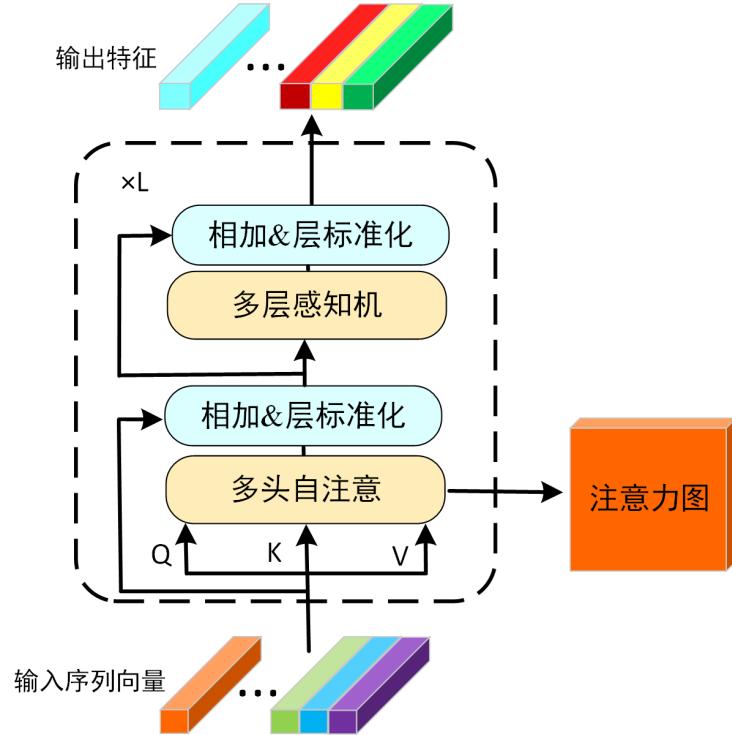


图 5.3 Vision Transformer 编码模块结构

不同的单头自注意单元在互不干扰、独立的特征子空间中学习相关特征，最终多头自注意模块对单头自注意单元输出的结果进行拼接，再经过线性变换得到该模块的输出。该输出与  $\mathbf{z}'$  进行残差连接，经过层标准化（Layer Normalization, LN）作为下一个多层感知机模块的输入。

$$\text{MSA}(\mathbf{z}_p) = \text{concat}_{i \in N_h} (\text{SA}(\mathbf{z}_p^i)) \mathbf{W}_{\text{out}} + \mathbf{b}_{\text{out}} \quad (5.7)$$

其中  $\mathbf{W}_{\text{out}} \in \mathbb{R}^{D \times D}$  为权重， $\mathbf{b}_{\text{out}} \in \mathbb{R}^{(N+1) \times D}$  为偏置。多层感知机模块由两个全连接层组成，第一个全连接层的激活函数为 ReLU，第二个全连接层不使用激活函数，计算公式如下：

$$\text{MLP}(\mathbf{X}) = \text{ReLU}(\mathbf{X} \cdot \mathbf{W}_1 + \mathbf{b}_1) \cdot \mathbf{W}_2 + \mathbf{b}_2 \quad (5.8)$$

若  $\mathbf{z}_{p-1}$  为第  $p$  个编码模块的输入，该编码模块的输出如下式所示：

$$\begin{aligned} \mathbf{z}'_p &= \text{LN}(\text{MSA}(\mathbf{z}_{p-1}) + \mathbf{z}_{p-1}) \\ \mathbf{z}_p &= \text{LN}(\text{MLP}(\mathbf{z}'_p) + \mathbf{z}'_p) \end{aligned} \quad (5.9)$$

### 5.2.3 稀疏注意力模块

血细胞分类中的关键问题是能否准确定位到图像中的辨识性区域，以图 5.4 中的粒细胞为例，不同发育阶段的粒细胞差异较为细微，原始粒细胞与早幼粒细胞

染色质都较为细致，区别主要是细胞质中是否存在非特异性颗粒。中幼粒细胞与晚幼粒细胞染色质都呈现出聚集的索块状，区别主要是细胞核形态是否存在凹陷。在卷积神经网络中主要通过区域推荐网络或者弱监督的分割掩码来定位图像中的辨识性区域，而在Vision Transformer模型中，其多头自注意机制可以自主学习不同图像块的权重。为了充分利用此权重信息实现辨识性区域的定位，本文提出了稀疏注意力模块。

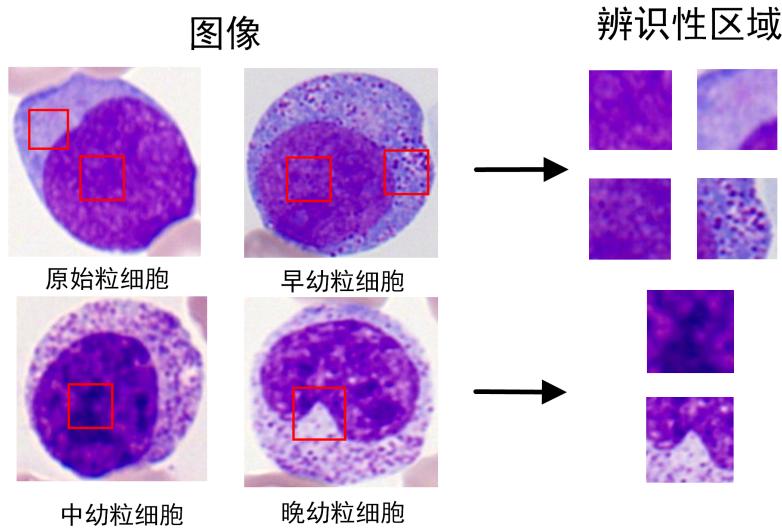


图 5.4 不同类别血细胞的辨识性区域

由于高层次特征的抽象性，其注意力图不一定能代表对应输入图像块的重要性，因此我们利用先前所有编码模块的注意力图信息并结合压缩激发模块来自主学习每个注意力图的权重。该模块首先将注意力图全局平均池化为一个描述符，接着使用两个全连接层建模注意力图间的相关性，最终得到每个注意力图的权重值  $\alpha$ 。将权重值归一化后与注意力图加权求和得到最终的注意力权重  $\mathbf{A}_{\text{attn}}$ ，如式 5.10 所示。整个流程如图 5.5 所示：

$$\mathbf{A}_{\text{attn}} = \sum_{i=1}^{L-1} \alpha_i \mathbf{A}_i \quad (5.10)$$

$\mathbf{A}_{\text{attn}}$  包含了低层特征与高层特征全部的注意力权重信息，相比于单层的注意力权重  $\mathbf{A}_{L-1}$  更适合筛选辨识性区域。我们使用  $\mathbf{A}_{\text{attn}}$  中分类向量对应的权重  $A_{\text{attn}}^{\text{class}} = [\mathbf{a}_{\text{final}}^1, \mathbf{a}_{\text{final}}^2, \dots, \mathbf{a}_{\text{final}}^{N_h}]$ ，在  $N_h$  个自注意头中筛选出最大权重所对应的隐含特征。最后，将这些隐含特征与分类向量进行拼接作为最后一层编码模块的输入。

$$\mathbf{z}_{L-1}^{\text{attn}} = \left[ \mathbf{z}_{L-1}^{\text{class}}; z_{L-1}^{a_1}; z_{L-1}^{a_2}; \dots; z_{L-1}^{a_{N_h}} \right] \quad (5.11)$$

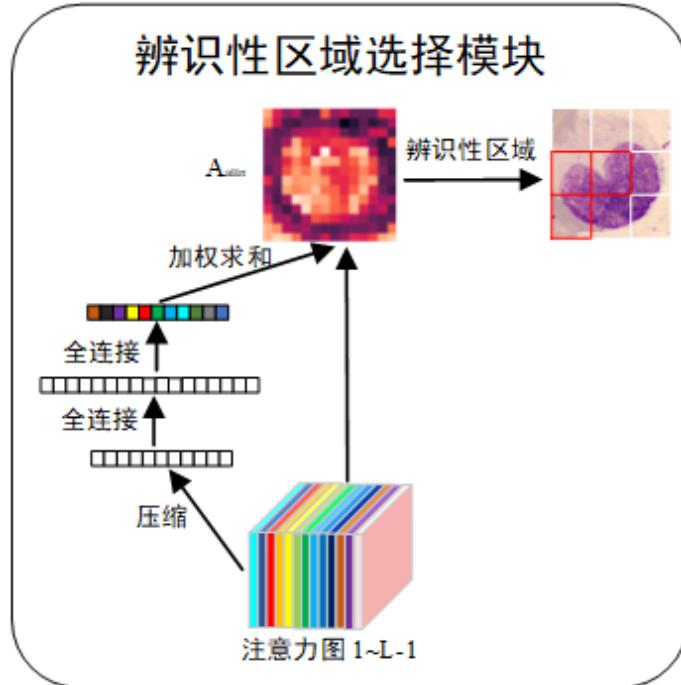


图 5.5 稀疏注意力模块结构示意图

稀疏注意力模块将全部序列向量替换为辨识性区域对应的特征向量并与分类向量进行拼接，然后输入到最后一个编码模块中，这样不仅保留了全局分类特征信息，还强制让最后一个编码层关注到不同类别之间的细微差异部分，同时舍弃了大量区分度较低的区域信息如背景、超类共同特征等，从而提升了网络的细粒度特征表达能力。

#### 5.2.4 损失函数

像 Vision Transformer 一样，我们将网络输出的第一个向量即分量向量用于图像分类。网络的损失函数包括了交叉熵损失  $L_{cross}$  与对比损失  $L_{con}$ ，如式 5.12 所示：

$$L = (\mathbf{y}, \mathbf{y}') + L_{con}(\mathbf{z}) \quad (5.12)$$

交叉熵损失用于衡量真实标签  $\mathbf{y}$  与网络预测标签的  $\mathbf{y}'$  的相似性，定义如式 5.13 所示：

$$L_{cross}(\mathbf{y}, \mathbf{y}') = - \sum_{i=1}^c y_i \log(y'_i) \quad (5.13)$$

为了进一步增加网络提取特征的类内相似性与类间差异性，我们加入了对比损失  $L_{con}$ <sup>[65]</sup>。对比损失使得不同标签对应的分类特征相似度最小，相同标签的分类特征相似度最大。为了使正负样本均衡，防止损失被简单的负样本（相似度很

小的不同类别特征) 所支配, 我们引入阈值  $t_{\text{con}}$ , 只有不同类别样本特征的相似度大于  $t_{\text{con}}$  时才计入到损失中, 当输入数据的批量大小为  $N$  时, 对比损失定义如式 5.14 所示:

$$L_{\text{con}} = \frac{1}{N^2} \sum_{i=1}^N \left[ \sum_{j:y_i=y_j} \left( 1 - \frac{\mathbf{z}_i \cdot \mathbf{z}_j}{\|\mathbf{z}_i\| \|\mathbf{z}_j\|} \right) + \sum_{j:y_i \neq y_j} \max \left( \frac{\mathbf{z}_i \cdot \mathbf{z}_j}{\|\mathbf{z}_i\| \|\mathbf{z}_j\|} - t_{\text{con}}, 0 \right) \right] \quad (5.14)$$

## 5.3 实验结果分析

### 5.3.1 数据集介绍

我们采用了 The Cancer Imaging Archive 平台上开源的血细胞形态学数据集 (The Munich AML Morphology Dataset, TMAMD)<sup>[66]</sup> 该数据来自慕尼黑医院 2014 年至 2017 年间 100 位被诊断为急性白血病的患者与 100 位无血液恶性肿瘤的患者。数据集包含了 15 类由专家标记的 18635 张单细胞图像。由于数据集存在较严重的类别不平衡问题, 部分类别的数量小于 30 使得网络难以有效的进行特征学习。本文只关注了样本数量大于 30 的十个类别。在选择的十个类别中, 对于数量较多的类别采用随机欠采样减少样本数量, 对于数量较少的类别, 采用水平、垂直翻转与旋转 90°、180° 的方式进行数据扩充, 表 5.1 为 TMAMD 数据集的分布情况与数据增强<sup>[67]</sup>后的样本分布情况。

### 5.3.2 实验环境与评价指标

实验中图像块的大小设置为  $16 \times 16$ , 滑动窗步长大小为 12, 式 5.14 中的阈值  $t_{\text{con}}$  大小为 0.4, 批尺寸大小设置为 32。我们在 Linux 操作系统下的 NVIDIA GeForce RTX 3090 显卡上训练模型。训练使用的深度学习框架为 Pytorch 1.10.1, 优化器使用随机梯度下降算法 (SGD), 动量设置为 0.9, 权重衰减设置为 5e-4, 学习率初始化为 0.001, 在第 40、70、90 个 epoch 时变为原来的 1/10。整个训练过程在第 100 个 epoch 停止。

为了定量评估分类算法的性能, 我们使用五折交叉验证与精确率 (Precision)、召回率 (Recall)、准确率 (Accuracy) 等评价指标, 定义如式 5.15- 5.17 所示

$$\text{Precision} = \frac{TP}{TP + FP} \quad (5.15)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (5.16)$$

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \quad (5.17)$$

表 5.1 TMAMD 血细胞数据集数据分布情况

序号	血细胞类别名	图像数量	是否选择	数据增强
1	分叶核嗜中性粒细胞 (NGS)	8484	√	1000
2	杆状核嗜中性粒细胞 (NGB)	109	√	545
3	典型淋巴细胞 (LYT)	3937	√	1000
4	非典型淋巴细胞 (LYA)	11	×	
5	单核细胞 (MON)	1789	√	1000
6	嗜酸性粒细胞 (EOS)	424	√	848
7	嗜碱性粒细胞 (BAS)	79	√	395
8	原始粒细胞 (MYO)	3268	√	1000
9	早幼粒细胞 (PMO)	70	√	350
10	二分裂早幼粒细胞 (PMB)	18	×	
11	中幼粒细胞 (MYB)	42	√	210
12	晚幼粒细胞 (MMZ)	15	×	
13	原始单核细胞 (MOB)	26	×	
14	有核红细胞 (EBO)	78	√	390
15	破碎细胞 (KSC)	15	×	
总计		18365		6738

### 5.3.3 实验结果

本文方法在慕尼黑 TMAMD 数据集上各类精确率与召回率如表 5.2 所示，混淆矩阵如图 5.6 所示。网络的 top-1 平均分类准确率为 91.96%，top-5 平均分类正确率为 99.48%。我们注意到对于最常见的血细胞类型，例如分叶核、杆状核嗜中性粒细胞、典型的淋巴细胞、嗜酸性粒细胞、有核红细胞，网络预测结果与医生标注达到了极好的一致性，精确率与召回率均高于 90%。而其他类别例如不同发育阶段的粒细胞以及嗜碱性粒细胞，由于相邻发育阶段的粒细胞差异较为细微并且原始样本数量较少，识别更具挑战性，存在误分类是可以容忍的。

此外，我们将本文提出的方法与其他深度学习方法例如 VGG<sup>[68]</sup>、ResNet<sup>[52]</sup>、SE-ResNet<sup>[69]</sup>、ResNext<sup>[70]</sup>、EfficientNet<sup>[71]</sup>、Vision Transformer<sup>[72]</sup>进行了对比。表 5.3 为不同方法在 TMAMD 数据集上的识别结果，表 5.3 第三列结果表明我们的方法在 TMAMD 数据上的识别准确率优于其他的方法，取得了有竞争力的性能。具体而言，我们改进的 Vision Transformer 与卷积神经网络相比识别准确率提升了 1.5% ~ 3.0%，本文图像块非重叠的模型与其基础框架相比识别准确率提升了

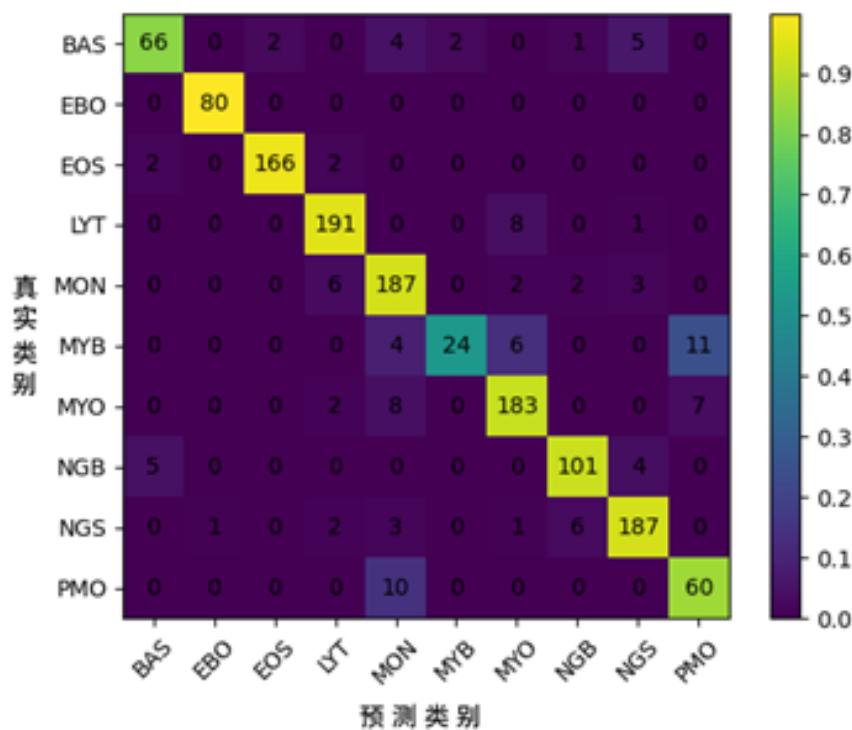


图 5.6 改进 Vision Transformer 的识别混淆矩阵

表 5.2 改进 Vision Transformer 方法的识别精确率与召回率

序号	血细胞类别名	精确率 (%)	召回率 %)	测试图像数量
1	嗜碱性粒细胞 (BAS)	90.41	82.50	80
2	有核红细胞 (EBO)	98.77	100.00	80
3	嗜酸性粒细胞 (EOS)	98.81	97.65	170
4	典型淋巴细胞 (LYT)	94.09	95.50	200
5	单核细胞 (MON)	86.57	93.50	200
6	中幼粒细胞 (MYB)	92.31	53.33	45
7	原始粒细胞 (MYO)	91.50	91.50	200
8	杆状核嗜中性粒细胞 (NGB)	91.82	91.81	110
9	杆状核嗜中性粒细胞 (NGS)	93.50	93.50	200
10	早幼粒细胞 (PMO)	76.92	85.71	70
总计				1355

0.74%，而模型的浮点运算次数仅增加 0.09GFLOPS、参数量增加 7.08MB。

表 5.3 不同识别方法性能对比

方法	骨干网络	准确率 (%)	运算次数 (GFLOPs)	参数量大小 (MB)
VGG	VGG16	88.85	15.53	134.31
ResNet	ResNet50	89.01	4.12	23.53
ResNet	ResNet152	89.22	11.58	78.63
SENet	SE-ResNet50	89.88	4.13	26.06
SENet	SE-ResNet101	89.96	7.86	47.3
ResNext	ResNext50	89.22	4.27	23.00
ResNext	ResNext152	90.25	11.80	57.92
EfficientNet	EfficientNet-B0	90.47	0.04	19.34
Vision Transformer	vit-base-p16	91.14	16.86	85.81
本文方法（图像块非重叠）	vit-base-cell-p16	91.88	16.95	92.89
本文方法（图像块重叠）	vit-base-cell-p16	91.96	19.44	92.92

图 5.7 中展示了本文模型的自注意力图。我们在数据集中随机选取了八个血细胞（细胞 1-细胞 8），其中，第一列为原始图像，第二列为整合后的注意力图，每种颜色表示不同头部的注意力。第三到八列为稀疏注意力模块多头注意力单元前六个头部所对应的注意力图。从整合注意力图中我们可以看到，细胞核、细胞质与背景分别被不同头部标记为不同的颜色。因此，本文模型的自注意机制有能力区分目标的不同区域。此外，我们将稀疏注意力模块每个头部最大权重对应的图像块进行标记，如图 5.8 所示。第一行图像中的红色边框为稀疏注意力模块所挑选的图像块区域，第二行为整体注意力图。我们看到模型主要关注了细胞核与细胞质等辨识性区域。以上可视化结果表明，本文模型成功捕捉到了细胞中的辨识性区域。

从表 5.3 中可以看到，本文最优模型的浮点运算次数与参数量大小均大于卷积神经网络方法，而卷积网络中的 EfficientNet 模型通过神经架构搜索，具有更优运算次数与识别准确率。因此，本文进一步探究了 Transformer 结构中嵌入向量的维度、编码层数量、与多头注意力的头数与模型准确率的关系，从而找到更好地模型速率与准确率的平衡。我们设计不同的超参数如表 5.4 所示，编码层数量为 4、头数量为 4，嵌入向量维度为 256 时，模型最小只有 7MB，运算次数为 1.18GFLOPS，但准确率较低仅有 78.08%。当将编码层的层数从 12 减少到 2 层，多头自注意的数

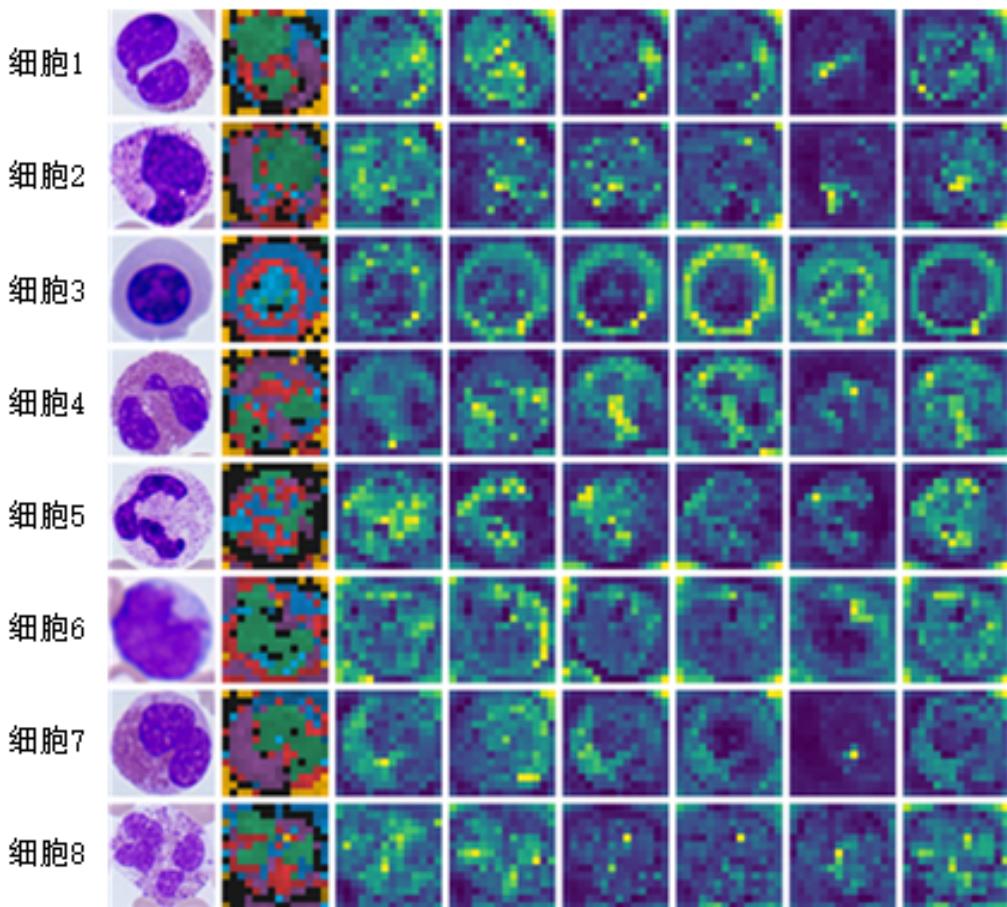


图 5.7 可视化自注意力图

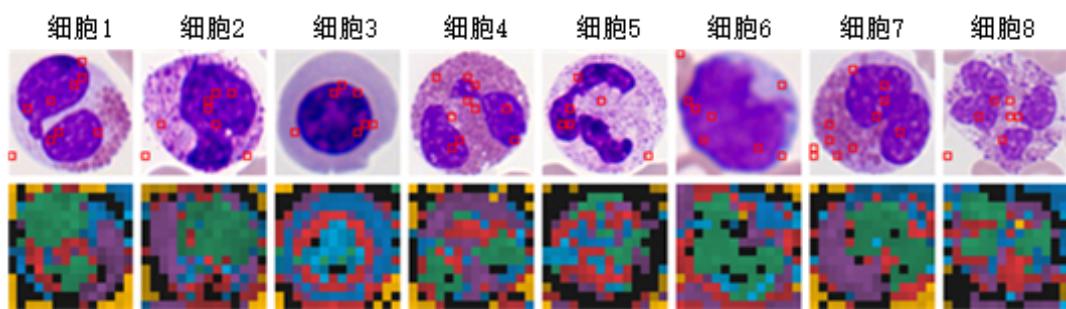


图 5.8 稀疏注意力模块选择的图像块

量与嵌入向量维度不变时，模型的准确度由 91.88% 下降到了 83.17%。当编码层数相同时，嵌入向量维度越低，模型的识别准确率也相应降低。整体来说，当本文模型识别准确率高于卷积网络时，参数量与运算次数也高于卷积网络。未来我们会关注于 Transformer 网络的结构搜索，从而找到更加精准的模型速率与准确率的平衡。

表 5.4 不同 Transformer 结构的速率与准确率

编码层数量	多头自注意数	嵌入向量维度	运算次数 (GFLOPs)	参数量大小 (MB)	准确率 (%)
12	12	768	16.95	92.89	91.88
10	12	768	14.16	78.73	90.85
8	12	786	11.37	64.54	89.81
6	12	768	8.58	50.37	87.64
4	12	768	5.79	36.28	85.01
2	12	768	3.0	22.02	83.17
12	8	512	10.04	55.13	84.12
8	8	512	6.73	38.32	82.41
4	8	512	2.8	17.57	80.85
12	4	256	4.39	24.18	78.64

### 5.3.4 消融实验

我们将本文模型进行消融实验研究来分析不同模块对细胞图像识别的影响，我们分别评估了图像块的划分方法，稀疏注意力模块，对比损失的影响。

#### 5.3.4.1 图像块划分方法

我们探究了图像块划分大小与图像块是否重叠对模型识别准确率的影响，实验结果如表 5.5 所示。图像块大小为 32 相比于大小为 16 的划分方式，嵌入后向量数量与模型的计算量都大幅降低，训练与推理的时间也减少约 3/4，但是模型的识别准确率较差。无论块大小是 16 还是 32，重叠划分方式相比非重叠划分方式模型识别准确率均有提高，而由此引入的额外计算成本也是可以接受的。图 5.9 实验结果表明，图像块划分越小，图像块之间存在重叠可以使得图像的局部细节保留的更加完整，模型的识别准确率越高。

表 5.5 不同图像块划分方式的消融研究

块大小	划分方式	嵌入向量数量	准确率 (%)	运算次数 (GFLOPs)
32	非重叠	50	88.92	4.36
32	重叠	82	89.88	7.16
16	非重叠	197	91.88	16.95
16	重叠	226	91.96	19.44

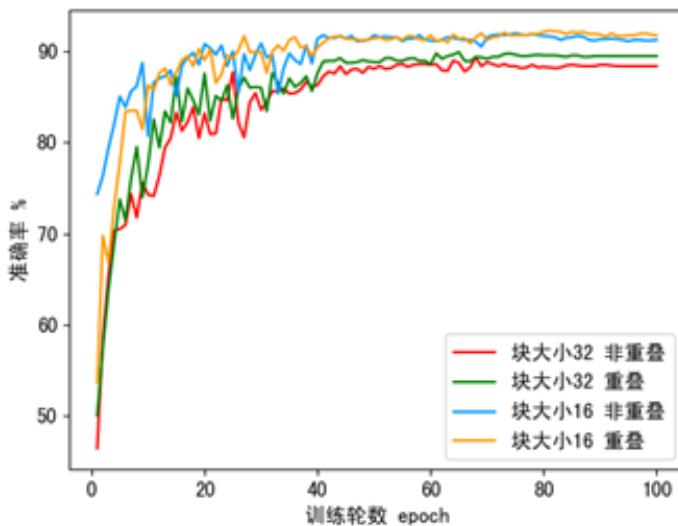


图 5.9 图像块划分方式对模型性能的影响

#### 5.3.4.2 稀疏注意力模块

通过稀疏注意力模块来选择显著性图像块作为最后编码层的输入，模型的识别准确率从 91.14% 提高到 91.8%。我们认为通过稀疏注意力的方式，模型将采样最具辨别力的图像块作为输入，从而明确丢弃一些无用的图像块并迫使网络对重要的部分进行学习。

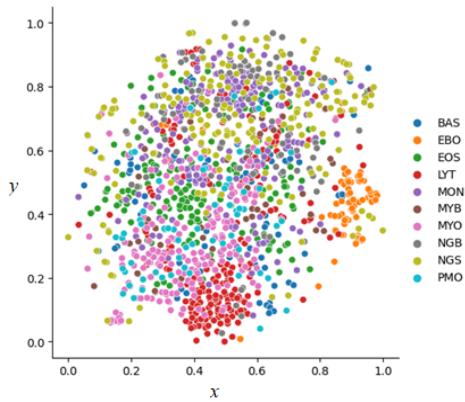
#### 5.3.4.3 对比损失

Vision Transformer 与本文模型在有无对比损失情况下的识别性能如表 5.6 所示。实验结果表明，通过加入对比损失，Vision Transformer 的识别准确率提升了 0.52%，本文模型的识别准确率提升了 0.59%。图 5.10 为测试集图像与本文模型输出分类特征的 t-SNE<sup>[73]</sup> 降维可视化结果，我们发现加入对比损失后，不同类别的分类特征在嵌入到二维空间后距离增大，相同类别的分类特征距离减小。综合上述结果，我们认为加入对比损失可有效扩大相似子类之间的特征距离，并减少相

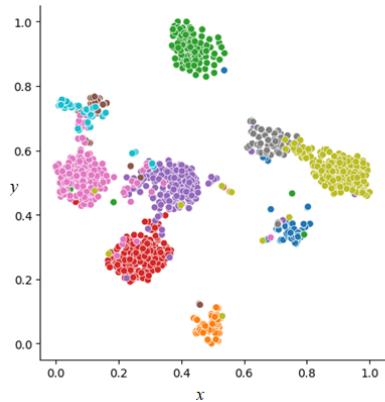
同类别之间的特征距离，从而提升模型的识别性能。

表 5.6 对比损失的消融研究

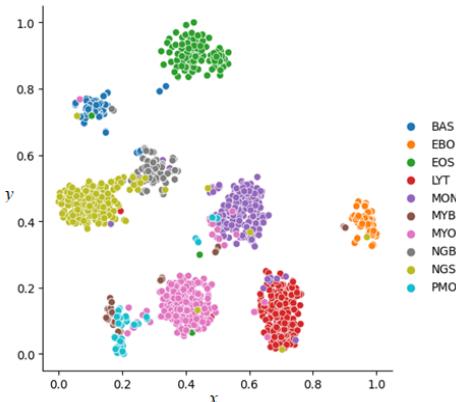
方法	对比损失	准确率 (%)
Vision Transformer	×	90.62
Vision Transformer	√	91.14
本文模型	×	91.29
本文模型	√	91.88



(a) 测试集图像 t-SNE 降维



(b) 无对比损失分类特征 t-SNE 降维



(c) 使用对比损失分类特征 t-SNE 降维

图 5.10 t-SNE 降维可视化结果

## 5.4 小结

目前血细胞识别研究主要侧重于五类血细胞的粗分类，很少有研究关注血细胞大类中子类的识别。针对上述问题，本章提出了一种基于改进 Vision Transformer 的血细胞识别模型。我们基于 Vision Transformer 中的自注意机制提出了稀疏注意

力模块，该模块综合利用了所有编码层的注意力权重信息，捕捉到图像中的辨识性区域，提升了模型的细粒度特征表达能力。本文采用对比损失进一步增加了网络学习特征的类内一致性与类间差异性。本文方法在 TMAMD 数据集上取得了先进的性能，定性与定量的可视化结果均表明了本方法的有效性与可解释性。相较于其他识别方法，本文方法具有更高的识别准确率，有望为医生临床诊断提供参考依据，具有潜在的临床应用前景。

## 第6章 骨髓血细胞检测与识别软件设计

本章介绍基于深度学习的骨髓血细胞检测与识别软件的设计。首先分析了软件的需求，包括了功能性需求与非功能性需求。接着，介绍了软件的架构设计与各个模块的流程图与数据库表设计。软件模块包括了用户登录注册模块、骨髓血细胞检测模块、骨髓血细胞识别模块、患者数据管理模块与日志模块。最后展示了软件实际实现的功能界面，并设计相关测试用例对软件功能进行测试。

### 6.1 需求分析

本章的目的是实现一个骨髓血细胞自动化检测识别软件，通过第三、四章介绍的深度学习算法将血细胞硬件采集设备收集的图像自动完成血细胞的定位、分类计数。最终根据 FAB 分类标准给出病情的诊断。本软件主要解决人工镜检流程复杂、枯燥、主观性强等问题。医生可以将患者数据一键上传，在患者信息管理界面可以查询相关患者的单张血细胞图像分类结果与整体血细胞分布的柱状图，此外还可以对错分类的血细胞重新进行标注。上述血细胞数据均落入到云端的数据仓库中，通过数据的不断积累，未来可以进一步提升模型的识别性能。

#### 6.1.1 功能需求分析

骨髓血细胞检测识别软件包含以下几种功能，用户登录/注册功能、骨髓血细胞图像上传/检测功能、骨髓血细胞图像识别功能、患者数据管理查看分析功能。

##### 1) 用户登录/注册功能

软件的首页为登录页，用户只有登录后才能使用网站的全部功能。注册用户仅为医院医生，注册后可以使用骨髓血细胞检测、骨髓血细胞识别功能，并能对检测识别结果进行修订与更改。医生注册后可以对个人信息如昵称、电话、邮箱、部门、性别进行修改。

##### 2) 血细胞检测功能

医生输入患者的 ID 后，将扫描设备拍摄的骨髓血细胞数字化图像上传，上传后图像可在界面实时显示，并展示检测到的血细胞切片，提供切片图像压缩包下载的功能。在检测精度方面，交并比 (IOU) 为 0.75 时，检测的平均精度 (AP75) 不低于 0.90。在 IOU 阈值为 0.50~0.95 时，检测平均精度 (AP50:75) 不低于 0.85。

##### 3) 血细胞识别功能

医生在输入患者 ID 后，选择切片图像上传，软件可以将识别的结果呈现给医

生。在识别精度方面，整体分类正确率不低于 0.90，针对常见的骨髓血细胞类型如嗜中性粒细胞、淋巴、红细胞的 f1-score 大于 0.95。

#### 4) 患者数据管理分析功能

医生可以搜索某患者的骨髓血细胞图像切片，并展示算法识别的类别，医生可以对血细胞识别的类别进行修改，并人工确认。软件可以对患者的血细胞类别数量分布以直方图或饼状图的形式进行展现，根据 FAB 标准对血液疾病类型给出初步诊断结果。

### 6.1.2 非功能性需求分析

软件的非功能性需求主要包含以下几个方面。1) 性能与速度，需要每小时可完成 20 张涂片约 2000 个骨髓血细胞的检测与识别。2) 数据安全性与可靠性，数据计算时，需要保证算法的正确与稳定性，使用数据库维护与更新数据，保证数据的完整性和可追溯性。3) 软件的稳定性，软件在使用中不会因为某些异常或错误而崩溃或无法正常运行。4) 软件的易用性，需要有良好用户交互体验的界面设计，用户能够方便的使用软件骨髓血细胞检测识别任务。5) 软件的可扩展性，有清晰的模块化与接口化设计，可以方便的进行检测识别算法的升级。架构上采用可扩展的架构，可方便的支持新增功能。

## 6.2 软件设计

### 6.2.1 软件架构设计

骨髓血细胞检测与识别软件开发使用的技术框架为 B/S 架构。开发环境为 Windows 10、前端使用 Vue 框架与 Element UI 组件库。后端使用的框架 Django 2.4、深度学习模型部署工具 ONNX、数据库为 Mysql 8.0。

骨髓血细胞检测识别架构如图 6.1 所示，根据用户需求分析，主要划分为四个模块：用户模块、骨髓血细胞检测模块、骨髓血细胞识别模块与患者数据管理模块。

软件采用前后端分离的架构，分别进行独立开发与部署。前端与后端的数据交互与通信使用 axios 网络请求库。后端服务部署在 GPU 服务器上，前端服务可以部署在 CDN 或者云服务器上，可以方便的进行系统升级、扩容与扩展。软件无需用户进行安装，打开浏览器输入网页域名即可使用，具有非常好的跨平台性，在 windows、mac、linux 操作系统下均可使用。数据与计算服务均在云端实现，降低了对使用者本地计算机资源的依赖。

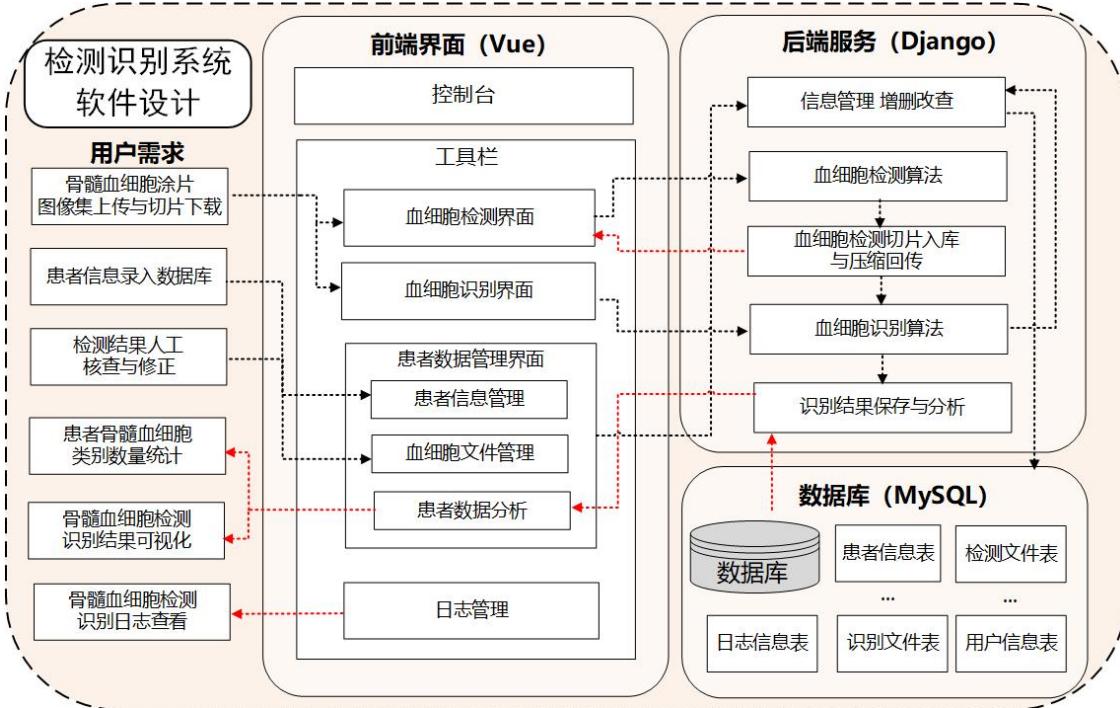


图 6.1 骨髓血细胞检测识别软件架构图

### 6.2.2 软件数据库设计

数据库是骨髓血细胞检测识别软件中的重要部分，其用于存储与管理血细胞数据，为医生提供方便快捷的数据查询、修改与更新等功能。MySQL 是一个开源且功能强大的关系型数据库管理系统，支持并发读写，可以处理大规模数据，可靠性强，能够保证数据的完整性与安全性，因此我们使用 MySQL 数据库<sup>[74]</sup>用于软件的数据存储与管理。

软件数据库表主要包括了用户信息表、检测文件表、识别文件表、日志表等，其设计的合理性对于软件性能稳定性与可扩展性至关重要，下面将对各数据表设计进行详细介绍。

1) 用户信息表，用户信息表如表 6.1 所示，其记录了用户基本信息如用户名、密码、头像、邮箱、手机等。用户密码通过 MD5 加密算法生成一个长度为 128 位的哈希值存储在数据库中，在登录时，通过比较用户密码的哈希值是否相同对用户进行校验。用户登录后可以对自己的个人信息进行修改。超级用户如医生可以对骨髓血细胞检测与识别的结果进行人工核对与修改。

2) 检测文件表，表 6.2 为检测文件表，该表用于存储医生上传的患者骨髓血细胞图像以及检测算法检测到的血细胞的坐标。首先根据上传的文件内容根据 MD5 算法计算出文件的 128 位哈希值，并将该名称作为文件名称落入到数据库中。血细胞检测结果坐标以 json 字符串的形式进行存储。

表 6.1 用户信息表

字段名称	数据类型	约束条件	字段说明
<code>id</code>	INTEGER	主键/自增	唯一用户 ID
<code>username</code>	varchar(150)	非空	用户名(登录)
<code>password</code>	varchar(128)	非空	用户登录密码
<code>email</code>	varchar(255)	非空	用户邮箱
<code>nickname</code>	varchar(40)	-	用户昵称
<code>mobile</code>	varchar(255)	-	用户电话
<code>avatar</code>	varchar(255)	-	用户头像
<code>gender</code>	INTEGER	-	用户性别
<code>dept_id</code>	bigint	外键	部门 ID
<code>create_datetime</code>	datetime	-	用户创建时间
<code>update_datetime</code>	datetime	-	最近一次修改时间
<code>last_login</code>	datetime	-	最近一次登录时间
<code>is_superuser</code>	bool	-	是否超级用户

3) 识别文件表, 表 6.3 为识别文件表, 该表用于存储医生上传的患者骨髓血细胞切片图像与识别的血细胞类别信息。根据上传的文件内容根据 MD5 算法计算出文件的 128 位哈希值, 并将该名称作为文件名称落入到数据库中。`is_confirmed` 字段表示该条识别结果是否被医生人工核对。

### 6.3 各个模块设计

本节将详细介绍四大模块的计算流程设计与 UI 界面设计。用户主界面如图 6.2 所示, 界面左侧为工具栏, 功能选项有控制台、骨髓血细胞检测、骨髓血细胞识别、患者数据管理与日志管理。患者数据管理菜单下有三个子菜单分别是患者信息管理、患者数据分析、血细胞文件管理。Tab 栏展示了当前界面所属的窗口, 可以方便的进行不同功能页面的切换。主窗口显示用户的操作界面, 展示用户上传的骨髓血细胞数据与检测识别的结果, 并将患者骨髓血细胞的数据分布以柱状图或饼状图的形式展示。右上角的用户管理部分可查看与修改当前用户信息、注销当前用户登录, 系统通知消息也在此部分进行展示。

表 6.2 检测文件表

字段名称	数据类型	约束条件	字段说明
id	INTEGER	主键/自增	唯一文件 ID
filename	varchar(200)	非空	上传图像名称
url	varchar(100)	非空	文件存储路径
md5sum	varchar(36)	非空	文件 MD5 哈希值
patient_id	bigint	非空	患者 ID
creator_id	bigint	外键	上传者 ID
create_datetime	datetime	-	上传时间
update_datetime	datetime	-	最近一次修改时间
description	varchar(1024)	-	检测结果 json 描述

表 6.3 识别文件表

字段名称	数据类型	约束条件	字段说明
id	INTEGER	主键/自增	唯一文件 ID
filename	varchar(200)	非空	上传图像名称
url	varchar(100)	非空	文件存储路径
md5sum	varchar(36)	非空	文件 MD5 哈希值
cell_id	INTEGER	-	骨髓血细胞类别 ID
cell_name	varchar(200)	-	骨髓血细胞名称
patient_id	bigint	非空	患者 ID
creator_id	bigint	外键	上传者 ID
create_datetime	datetime	-	上传时间
update_datetime	datetime	-	最近一次修改时间
is_confirmed	bool	-	是否人工核对

### 6.3.1 用户模块

用户登录与修改个人信息界面如图 6.3 所示，用户模块是网站最基础的模块，用户在注册后，用户信息会写入数据库的用户信息表中，在登录时需要填写注册时的用户名、密码与验证码信息，用户在点击登录按钮后，后端会对用户名与密码在数据库中进行核验，如果正确则会跳转到软件的首页的控制台页面，否则会提示用户输入的用户名或密码错误。用户在登陆后，可以点击右上角的头像去修改

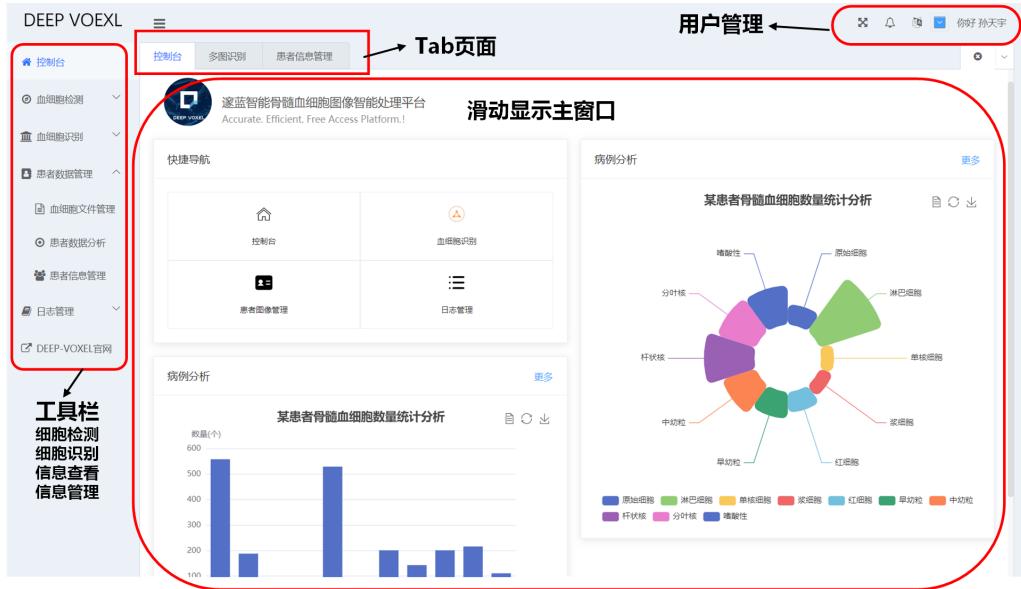


图 6.2 骨髓血细胞检测识别软件界面

并完善个人信息，如密码、电话、邮箱与昵称等。



图 6.3 用户模块界面示意图

### 6.3.2 骨髓血细胞检测模块

骨髓血细胞检测界面如图 6.4 所示，首先需要填写患者 ID，然后点击“+”号，上传患者骨髓血细胞图像，该过程调用后端 /api/system/detect\_file 接口，将细胞图像写入到数据库中，调用成功后将文件的 url 列表返回前端。然后点击检测按钮调用后端 /api/recognition/save\_detect 接口，传输参数为待检测的 url 文件列表，后端调用基于 ONNX 部署的改进的 RetinaNet 骨髓血细胞检测模型，然后将检测得到的坐标信息返回给前端，在检测结果区域以红色框进行标出。最后可以点击下载

切片按钮，获得剪切后的血细胞图像切片文件。

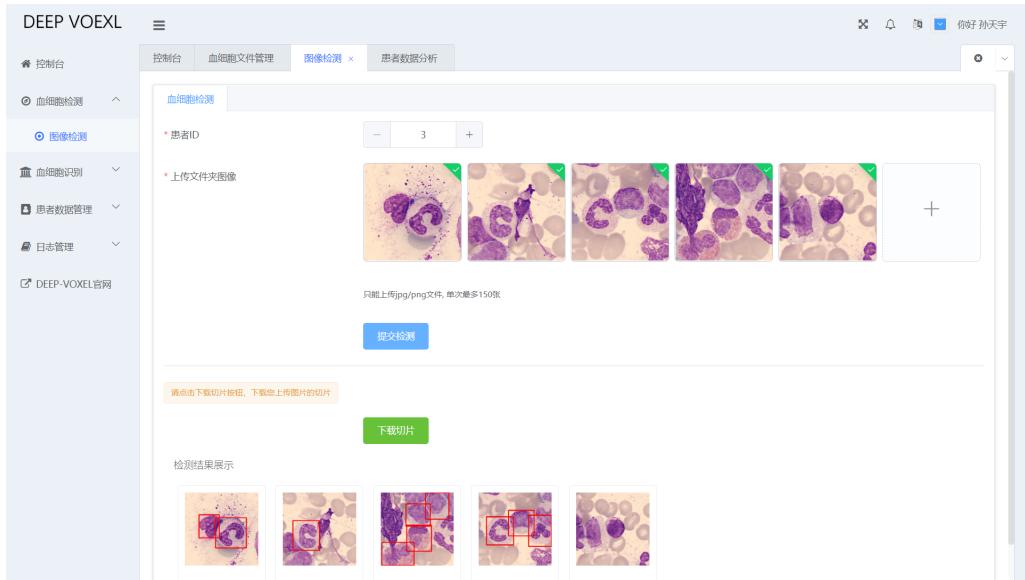


图 6.4 骨髓血细胞检测界面

### 6.3.3 骨髓血细胞识别模块

骨髓血细胞识别界面如图 6.5 所示，首先输入患者的 ID，接着点击“+”号，选取检测部分下载的骨髓血细胞切片图像，类似骨髓血细胞检测界面部分，图像会存储到数据库中。点击识别提交按钮，后端会调用 ONNX 部署的改进 Vision Transformer 模型对骨髓血细胞类别进行识别，将血细胞的类别与名称以 json 串的形式返回，前端在页面下方的文本框中会按照上传图像的顺序显示各个图像的类别编号与类别名称。每个血细胞的具体信息可以到患者数据管理菜单下的血细胞文件管理页面进行查看。

### 6.3.4 患者数据管理模块

患者数据管理模块下总共有三个子页面，分别是血细胞文件管理页面、患者数据分析页面与患者信息管理页面。

#### 6.3.4.1 血细胞文件管理

血细胞文件管理界面将骨髓血细胞识别界面上传的血细胞图像数据与识别结果以表格的形式进行展现，如图 6.6 所示，表格从左到右列名分别是患者 ID、文件名称、缩略图、血细胞类别、血细胞名称、上传/核验医师、是否人工确认、更新时间与操作。点击血细胞类别下拉列表，可以对类别进行更改，同时自动更新血细胞名称。点击是否人工确认下拉列表选择 true 表示该条数据已经通过人工确认。

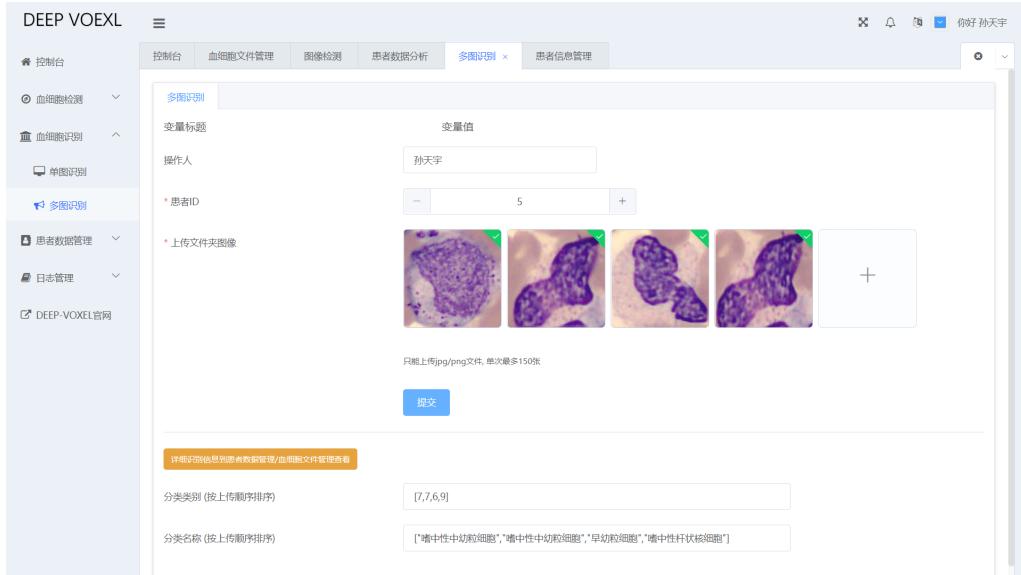


图 6.5 骨髓血细胞识别界面

在最后一列的操作选项中，可以对血细胞其他信息进行编辑或者删除。

DEEP VOEXL									
控制台 血细胞文件管理 图像检测									
患者ID 患者ID 文件名称 搜索文件名称 搜索 重置									
序号	患者ID	文件名称	缩略图	血细胞类别	血细胞名称	上传/检验医师	是否人工确认	更新时间	操作
1	3	ER0181_163_x231_y2_47_2.jpg		8	嗜中性粒细胞	孙天宇	false	2023-03-29 18:15:06	
2	3	ER0181_163_x123_y1_36_1.jpg		9	嗜中性杆状核细胞	孙天宇	false	2023-03-29 18:15:06	
3	3	ER0181_157_x135_y1_23_1.jpg		7	嗜中性中幼粒细胞	孙天宇	false	2023-03-29 18:15:06	
4	3	ER0181_162_x2_y239_2.jpg		6	早幼粒细胞	孙天宇	false	2023-03-29 18:15:05	
5	3	ER0181_158_x125_y1_40_1.jpg		11	嗜酸性粒细胞	孙天宇	false	2023-03-29 18:15:05	

图 6.6 医生核验识别结果界面

### 6.3.4.2 患者数据分析

患者数据分析界面如图 6.7 所示，输入患者 ID，点击查询后，在下方患者详情的文本框中会以文本的形式展示各血细胞类别的名称与数量。各类血细胞的分布以柱状图与饼状图的方式呈现给医生，辅助医生对白血病的类型进行诊断。决策算法根据 FAB (French-American-British) 分型标准给出初步的白血病类型诊断建议。在 FAB 分类标准中，原始细胞占比大于 30% 作为急性白血病的诊断标准，根

据原始细胞的类型分为急性淋巴细胞白血病（acute lymphoblastic leukemia, ALL）与急性髓系白血病（acute myeloid leukemia, AML）。当原始/幼稚淋巴细胞占比超过 50% 为急性淋巴细胞白血病（ALL）。AML 白血病根据不同细胞的占比分为 M0~M7 型，例如颗粒过多早幼粒细胞占比大于 30% 为 M3 型的急性早幼粒细胞白血病。

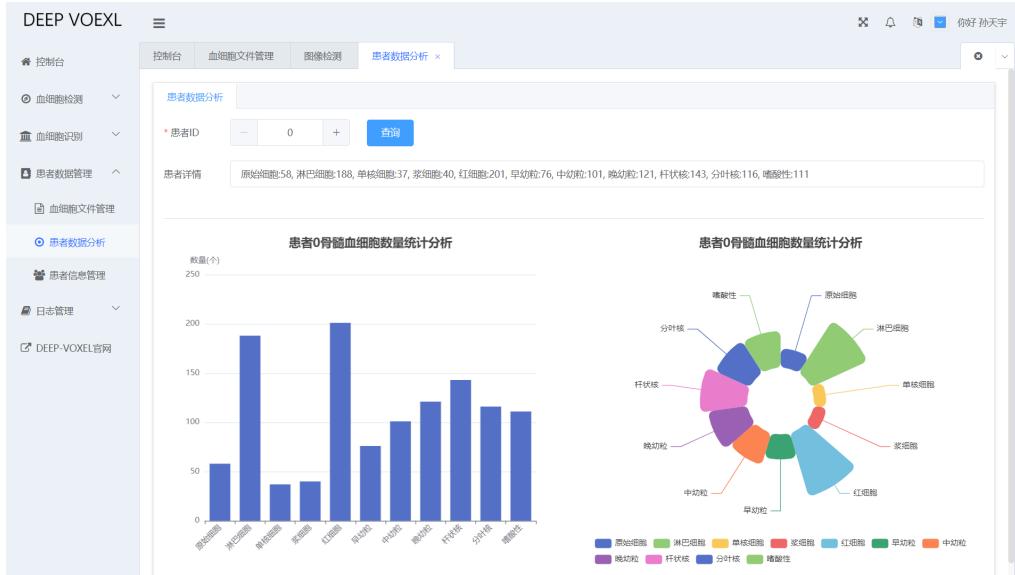


图 6.7 患者数据分析界面

#### 6.3.4.3 患者信息管理

患者信息管理界面如图 6.8 所示，在该界面中医生可以新增、删除、编辑、导入与导出患者数据。点击 + 新增按钮，弹出子页面中依次填入患者的姓名、性别、身份证号码等信息，点击确定即可完成患者信息的录入。在页面的最上方可以根据姓名、关键词等信息进行患者信息的搜索。

## 6.4 小结

本文基于 B/S 架构设计并开发了骨髓血细胞检测与识别软件。前端使用 Vue 框架与 Element UI 组件库。后端使用的框架为 Django2.4，深度学习模型部署工具使用 ONNX 进行模型部署，数据存储使用 Mysql8.0。软件实现了用户模块、骨髓血细胞检测模块、骨髓血细胞识别模块与患者数据管理模块。通过该软件，医生可以将患者骨髓血细胞图像数据一键上传，软件自动完成血细胞的定位、分类计数，医生可以查询相关患者的单张血细胞图像分类结果与整体血细胞分布的柱状图，并对血细胞类别进行校对与修改，软件最终根据 FAB 分类标准给出诊断建议。

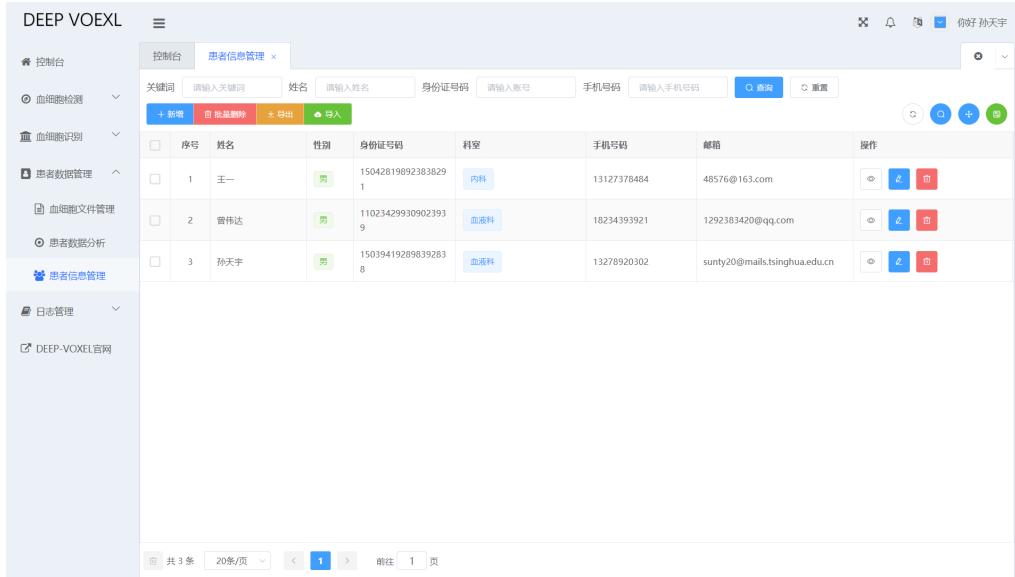


图 6.8 患者信息录入页面

实测中，软件的运行速度与检测识别精度均达到了项目指标的要求，未来有望应用于白血病的临床辅助诊断。

## 第7章 总结与展望

### 7.1 全文总结

过去的十年中，计算机科学技术飞速发展，人工智能广泛赋能于医学行业，为药物研发、疾病诊疗等领域带来了深刻的变革。如今，各大医院已经积累了海量患者数据，为基于数据驱动的深度神经网络提供了数据支撑。白血病是一种常见多发的血液疾病，目前该疾病的诊断主要依靠医生在显微镜下对骨髓血细胞进行形态学检查。全自动化的骨髓血细胞分类计数，可以将医生从繁重的细胞病理工工作中解放出来，具有非常重要的临床辅助诊断意义。

复杂背景下的骨髓血细胞检测与识别是实现全自动化骨髓血细胞形态学检查的重要研究课题，本文以邃蓝智能科技（上海）有限公司提供的骨髓血细胞样本为例，深入分析了目前骨髓血细胞检测与识别研究面临的技术难点，并研究了基于深度学习的骨髓血细胞检测识别算法，最后设计并实现了骨髓血细胞检测识别软件将相关算法进行落地，本文的主要工作包含了以下的四个方面：

- 1) 提出主动学习的数据标注迭代框架、先检测再识别的算法策略。

本文使用数据来源于实践基地邃蓝智能科技有限公司合作医院提供的脱敏数据。原始数据无标签，首先在医生协作下完成数据集血细胞边界框与类别信息的标注。大量的数据标注耗时费力，本文采用主动学习技术，标注少量数据训练模型，对未标注的数据根据网络输出的熵值降序排序，去发掘数据集中高信息量的样本，提高标注的效率与精度，降低标注成本。在骨髓血细胞的检测方面，本文对比了多种单阶段与双阶段目标检测网络的检测精度、计算量与速度等性能，并确定将 RetinaNet 作为骨髓血细胞检测的基线模型。本文探索了 RetinaNet 网络在仅检测与检测识别一体化任务上特征提取与识别准确率上的差异，实验结果表明检测识别一体化任务的输出类别置信度较低，平均识别正确率只有 58.1%，易发生漏检与误检。仅检测任务的平均精度达到了 94.2%，检测的准确率与召回率都有大幅提升。本文认为两种任务共享骨干网络提取的特征，导致骨干网络难以对精细分类特征进行学习，因此确定了先检测再识别的骨髓血细胞处理流程，即先使用检测网络定位到血细胞的位置，然后剪切为血细胞切片，再输入到识别网络中进行分类的流程。

- 2) 提出了基于改进 RetinaNet 的骨髓血细胞检测算法

确定了先检测再识别的流程后，骨髓血细胞检测部分更加关注血细胞定位精度。为了解决密集、堆叠与黏连血细胞区域的漏检、错检等问题，本文提出了一种

基于改进 RetinaNet 的骨髓血细胞检测网络。该网络采用了基于全局注意力的自下而上的路径聚合模块，通过特征直连缩短了底层到顶层特征之间的信息传递路径，使得浅层的纹理等高分辨定位信息可以更有效的传递到顶层，提升网络定位特征的表达能力。网络引入了 IOU 预测分支，对每个锚框与可能对应真实框交并比进行预测，在推理阶段将检测框的定位质量也纳入到非极大值抑制的考量当中。此外，本文探究了不同标签分配策略对检测性能的影响，提出基于最优输运的标签策略用于密集区域的血细胞的标签分配，避免了模糊分配样本的出现，提高网络对血细胞的召回能力。在骨髓血细胞数据集上的实验结果表明，本文提出的改进方法在检测精度（mAP）上相比于主流的目标检测网络提升了 1% 以上，达到了较为先进的性能。网络单张图像的平均检测时间为 37ms，可以用于实时全自动化骨髓血细胞检测。

### 3) 提出了基于改进 Vision Transformer 的骨髓血细胞识别算法

针对骨髓血细胞种类繁多，相邻发育阶段的血细胞在形态上非常类似，骨髓血细胞子类之间差异较小等识别难点，本文将性能优异的 Vision Transformer 作为基线模型，提出了重叠图像块划分方法、辨识性区域选择模块与对比 loss 对基线模型进行改进。其中重叠图像块划分方法可以更好的保留图像的局部信息、避免破坏图像的局部结构。辨识性区域选择模块采用压缩激发结构学习所有编码层的注意力图权重，并将每个注意力头最大权重对应的隐含特征用于分类，该模块让网络关注到不同类别之间的细微差异部分，同时舍弃大量区分度较低的背景、超类共同特征区域，从而提升网络的细粒度特征表达能力。为了进一步增加分类特征的类内一致性与类间差异性。本文引入了对比损失，对特征的间隔进行约束，使得不同标签特征相似度最小，相同标签的特征相似度最大。实验结果表明，本文方法在慕尼黑骨髓血细胞数据集上的平均分类准确率为 91.96%，与卷积神经网络相比识别准确率提升了 1.5% ~ 3.0%，相比于基线 Vision Transformer 模型识别准确率提升了 0.74%，有望为医生临床诊断提供参考依据，具有潜在的临床应用前景。

### 4) 将上述算法落地，研究并开发了骨髓血细胞检测与识别软件

本文基于 B/S 架构设计并开发了骨髓血细胞检测与识别软件。前端使用 Vue 框架与 Element UI 组件库。后端使用的框架 Django2.4，深度学习模型部署工具 ONNX，数据库为 Mysql 8.0。软件实现了用户模块、骨髓血细胞检测模块、骨髓血细胞识别模块与患者数据管理模块。通过该软件，医生可以将患者骨髓血细胞图像数据一键上传，软件自动完成血细胞的定位、分类计数。最终根据 FAB 分类标准给出病情的诊断。医生可以查询相关患者的单张血细胞图像分类结果与整体血细胞分布的柱状图，并对血细胞类别进行校对与修改。血细胞数据均落入到云端

的数据库中，通过数据的不断积累，未来可以进一步提升模型的检测与识别性能。

## 7.2 工作展望

未来工作展望，骨髓血细胞自动化检测识别技术集成了计算机视觉、深度学习、目标检测、目标识别等技术，其研究对于实现全自动的骨髓血细胞形态学分析具有重要的价值。本文实现了多类骨髓血细胞的检测与识别，但在实际应用中，骨髓血细胞成分非常复杂，细胞种类繁多、背景多变，未来仍需在以下三个方面展开研究：

(1) 持续收集骨髓血细胞数据集，并对血细胞图像类别进行人工标注。目前单核细胞、早幼粒细胞等由于数据较少，识别效果较差，后续需要特别进行扩充。目前医院临床诊断报告中包含约 30 余类血细胞的计数信息，需要继续扩充目前训练集中不存在的骨髓血细胞类别，使模型可以达到进行临床诊断的标准。

(2) 尽管 Vision Transformer 实现了非常高的骨髓血细胞分类准确率，但是模型参数量为 93MB，运算量为 19.44GFLOPs，对部署的硬件环境有较高的要求。未来需要进一步精简其结构，提高模型的识别速度。可以采用神经网络结构搜索方式，设计超网络结构，采用基于可微分的搜索策略如 DARTS<sup>[75]</sup>、GDAS<sup>[76]</sup>等，对网络中不重要的分支与结构进行裁剪，找到更加精准的模型速率与准确率的平衡。

(3) 有标签骨髓血细胞数据获取的成本极其昂贵，目前医院有海量的无标签骨髓血细胞数据，未来研究应该聚焦于如何将利用大量的无标签数据来提升模型的性能。例如利用无监督学习<sup>[77]</sup>通过大量的无标签数据进行预大模型的预训练，然后基于少量的有标签数据进行下游任务的迁移。或者采用伪标签的半监督学习<sup>[78]</sup>方式给无标签的数据打上标签信息，然后先利用这些伪标签进行训练，再进行微调。半监督/无监督学习可以降低标注数据的需求、通过使用未标记数据来学习更多的特征，使得模型对噪声和异常值更加鲁棒，泛化性能更强。

## 参考文献

- [1] 黄治虎, 陈宝安, 欧阳建, 等. 我国白血病流行病学调查的现状和对策[J]. 临床血液学杂志, 2009, 22(2): 166-167.
- [2] Musto P, Statuto T, Valvano L, et al. An update on biology, diagnosis and treatment of primary plasma cell leukemia[J]. Expert review of hematology, 2019, 12(4): 245-253.
- [3] Heimpel H. Conventional morphological examination of blood and bone marrow cells in the diagnosis of preleukemic syndromes[C]//Preleukemia. Springer, 1979: 4-11.
- [4] Bennett J M, Catovsky D, Daniel M T, et al. Proposals for the classification of the acute leukaemias french-american-british (fab) co-operative group[J]. British journal of haematology, 1976, 33(4): 451-458.
- [5] Parmar C, Barry J D, Hosny A, et al. Data analysis strategies in medical imaging[J]. Clinical Cancer Research, 2018, 24(15): 3492-3499.
- [6] Shen D, Wu G, Suk H I. Deep learning in medical image analysis[J]. Annual review of biomedical engineering, 2017, 19: 221-248.
- [7] Cseke I. A fast segmentation scheme for white blood cell images[C]//Iapr International Conference on Pattern Recognition. 1992.
- [8] Jiang K, Liao Q M, Dai S Y. A novel white blood cell segmentation scheme using scale-space filtering and watershed clustering[C]//Proceedings of the 2003 International Conference on Machine Learning and Cybernetics (IEEE Cat. No.03EX693). 2003.
- [9] Wu J, Zeng P, Zhou Y, et al. A novel color image segmentation method and its application to white blood cell image analysis[C]//2006 8th international Conference on Signal Processing: volume 2. IEEE, 2006.
- [10] 马建林, 崔志明, 吴健, 等. 一种新的基于区域增长的 ROI 分割算法[J]. 计算机应用研究, 2008, 25(5): 1582-1585.
- [11] Sadeghian F, Seman Z, Ramli A R, et al. A framework for white blood cell segmentation in microscopic blood images using digital image processing[J]. Biological Procedures Online, 2009, 11(1): 196-206.
- [12] Theera-Umporn N. White blood cell segmentation and classification in microscopic bone marrow images[C]//Fuzzy Systems and Knowledge Discovery: Second International Conference, FSKD 2005, Changsha, China, August 27-29, 2005, Proceedings, Part II 2. Springer, 2005: 787-796.
- [13] Ramoser H, Laurain V, Bischof H, et al. Leukocyte segmentation and classification in blood-smear images[C]//2005 IEEE Engineering in Medicine and Biology 27th Annual Conference. IEEE, 2006: 3371-3374.
- [14] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 580-587.

- [15] He K, Zhang X, Ren S, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition[J]. *IEEE transactions on pattern analysis and machine intelligence*, 2015, 37(9): 1904-1916.
- [16] Girshick R. Fast r-cnn[C]//Proceedings of the IEEE international conference on computer vision. 2015: 1440-1448.
- [17] Ren S, He K, Girshick R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[J]. *Advances in neural information processing systems*, 2015, 28.
- [18] Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection [C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 779-788.
- [19] Lin T Y, Goyal P, Girshick R, et al. Focal loss for dense object detection[C]//Proceedings of the IEEE international conference on computer vision. 2017: 2980-2988.
- [20] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. *Advances in neural information processing systems*, 2017, 30.
- [21] Zhu X, Su W, Lu L, et al. Deformable detr: Deformable transformers for end-to-end object detection[A]. 2020.
- [22] Fang Y, Liao B, Wang X, et al. You only look at one sequence: Rethinking transformer in vision through object detection[J]. *Advances in Neural Information Processing Systems*, 2021, 34: 26183-26197.
- [23] Wang W, Xie E, Li X, et al. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2021: 568-578.
- [24] Xia T, Jiang R, Fu Y Q, et al. Automated blood cell detection and counting via deep learning for microfluidic point-of-care medical devices[C]//IOP conference series: materials science and engineering: volume 646. IOP Publishing, 2019: 012048.
- [25] Dhib N, Ghazzai H, Besbes H, et al. An automated blood cells counting and classification framework using mask r-cnn deep learning model[C]//2019 31st international conference on microelectronics (ICM). IEEE, 2019: 300-303.
- [26] Shakarami A, Menhaj M B, Mahdavi-Hormat A, et al. A fast and yet efficient yolov3 for blood cell detection[J]. *Biomedical Signal Processing and Control*, 2021, 66: 102495.
- [27] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation[C]// Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 3431-3440.
- [28] Lu Y, Qin X, Fan H, et al. Wbc-net: A white blood cell segmentation network based on unet++ and resnet[J]. *Applied Soft Computing*, 2021, 101: 107006.
- [29] 伍柏青, 傅新文. 当代五分类血细胞分析仪技术原理分析[J]. *实验与检验医学*, 2011, 29 (4): 391-394.
- [30] Jain A K, Duin R P W, Mao J. Statistical pattern recognition: A review[J]. *IEEE Transactions on pattern analysis and machine intelligence*, 2000, 22(1): 4-37.

- 
- [31] Ghosh M, Das D, Mandal S, et al. Statistical pattern analysis of white blood cell nuclei morphometry[C]//2010 IEEE Students Technology Symposium (TechSym). IEEE, 2010: 59-66.
  - [32] Rezatofghi S H, Soltanian-Zadeh H. Automatic recognition of five types of white blood cells in peripheral blood[J]. Computerized Medical Imaging and Graphics, 2011, 35(4): 333-343.
  - [33] 孙凯, 姚旭峰, 黄钢. 基于机器学习的白细胞六分类研究[J]. 软件, 2020, 41(10): 5.
  - [34] 袁满. 血细胞图像白细胞的自动检测与识别[D]. 东南大学, 2017.
  - [35] Matek C, Schwarz S, Spiekermann K, et al. Human-level recognition of blast cells in acute myeloid leukaemia with convolutional neural networks[J]. Nature Machine Intelligence, 2019, 1(11): 538-544.
  - [36] Mori J, Kaji S, Kawai H, et al. Assessment of dysplasia in bone marrow smear with convolutional neural network[J]. Scientific reports, 2020, 10(1): 1-8.
  - [37] Fu X, Fu M, Li Q, et al. Morphogo: an automatic bone marrow cell classification system on digital images analyzed by artificial intelligence[J]. Acta Cytologica, 2020, 64(6): 588-596.
  - [38] Yoon J, Kwon J A, Yoon S Y, et al. Diagnostic performance of cellavision dm96 for plasmodium vivax and plasmodium falciparum screening in peripheral blood smears[J]. Acta tropica, 2019, 193: 7-11.
  - [39] Huang P, Wang J, Zhang J, et al. Attention-aware residual network based manifold learning for white blood cells classification[J]. IEEE Journal of Biomedical and Health Informatics, 2020, 25(4): 1206-1214.
  - [40] 从玉隆. 骨髓细胞形态学检验技术与临床[M]. 骨髓细胞形态学检验技术与临床, 2002.
  - [41] 付鑫焱, 王巍, 高海燕. 数字病理技术在骨髓细胞形态学检验中的应用进展[J]. 中国医疗器械杂志, 2022.
  - [42] Ren P, Xiao Y, Chang X, et al. A survey of deep active learning[J]. ACM computing surveys (CSUR), 2021, 54(9): 1-40.
  - [43] Shorten C, Khoshgoftaar T M. A survey on image data augmentation for deep learning[J]. Journal of big data, 2019, 6(1): 1-48.
  - [44] Rosenblatt F. The perceptron: a probabilistic model for information storage and organization in the brain.[J]. Psychological review, 1958, 65(6): 386.
  - [45] Lydia A, Francis S. Adagrad—an optimizer for stochastic gradient descent[J]. Int. J. Inf. Comput. Sci, 2019, 6(5): 566-568.
  - [46] Kingma D P, Ba J A, Adam J. A method for stochastic optimization. arxiv 2014: volume 106 [A]. 2020.
  - [47] Li Z, Liu F, Yang W, et al. A survey of convolutional neural networks: analysis, applications, and prospects[J]. IEEE transactions on neural networks and learning systems, 2021.
  - [48] Nelson B. Getting to know vue.js[J]. Apress, New York, 2018.
  - [49] Rubio D. Beginning django[M]. Springer, 2017.
  - [50] Liu W, Anguelov D, Erhan D, et al. Ssd: Single shot multibox detector[C]//Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. Springer, 2016: 21-37.

- [51] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition [A]. 2014.
- [52] 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. 2016: 770-778.
- [53] Huang G, Liu Z, Van Der Maaten L, et al. Densely connected convolutional networks[C]// Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 4700-4708.
- [54] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 1-9.
- [55] Lin T Y, Dollar P, Girshick R, et al. Feature pyramid networks for object detection[J]. IEEE Computer Society, 2017.
- [56] Neubeck A, Van Gool L. Efficient non-maximum suppression[C]//18th international conference on pattern recognition (ICPR'06): volume 3. IEEE, 2006: 850-855.
- [57] Cai Z, Vasconcelos N. Cascade r-cnn: Delving into high quality object detection[C]// Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 6154-6162.
- [58] Redmon J, Farhadi A. Yolov3: An incremental improvement[A]. 2018.
- [59] Li H, Xiong P, An J, et al. Pyramid attention network for semantic segmentation[A]. 2018.
- [60] He Y, Zhu C, Wang J, et al. Bounding box regression with uncertainty for accurate object detection[C]//Proceedings of the ieee/cvf conference on computer vision and pattern recognition. 2019: 2888-2897.
- [61] Zhang S, Chi C, Yao Y, et al. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 9759-9768.
- [62] Kim K, Lee H S. Probabilistic anchor assignment with iou prediction for object detection[C]// Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXV 16. Springer, 2020: 355-371.
- [63] Villani C, et al. Optimal transport: old and new: volume 338[M]. Springer, 2009.
- [64] 孙天宇, 朱庆涛, 杨健, 等. 基于改进 Vision Transformer 的血细胞图像识别方法研究[J/OL]. 生物医学工程学杂志, 2022, 39(6): 1097-1107. DOI: 10.7507/1001-5515.202203008.
- [65] Hadsell R, Chopra S, LeCun Y. Dimensionality reduction by learning an invariant mapping [C]//2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06): volume 2. IEEE, 2006: 1735-1742.
- [66] Matek C, Schwarz S, Marr C. A single-cell morphological dataset of leukocytes from aml patients and non-malignant controls[EB/OL]. 2022. <https://wiki.cancerimagingarchive.net/pages/viewpage.action?pageId=61080958>.
- [67] Van Dyk D A, Meng X L. The art of data augmentation[J]. Journal of Computational and Graphical Statistics, 2001, 10(1): 1-50.
- [68] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition [A]. 2014.

- [69] Hu J, Shen L, Sun G. Squeeze-and-excitation networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 7132-7141.
- [70] Xie S, Girshick R, Dollár P, et al. Aggregated residual transformations for deep neural networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 1492-1500.
- [71] Tan M, Le Q. Efficientnet: Rethinking model scaling for convolutional neural networks[C]// International conference on machine learning. PMLR, 2019: 6105-6114.
- [72] Dosovitskiy A, Beyer L, Kolesnikov A, et al. An image is worth 16x16 words: Transformers for image recognition at scale[A]. 2020.
- [73] Van der Maaten L, Hinton G. Visualizing data using t-sne.[J]. Journal of machine learning research, 2008, 9(11).
- [74] Christudas B, Christudas B. Mysql[M]. Springer, 2019.
- [75] Liu H, Simonyan K, Yang Y. Darts: Differentiable architecture search[A]. 2018.
- [76] Dong X, Yang Y. Searching for a robust neural architecture in four gpu hours[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 1761-1770.
- [77] Berry M W, Mohamed A, Yap B W. Supervised and unsupervised learning for data science[M]. Springer, 2019.
- [78] Ouali Y, Hudelot C, Tami M. An overview of deep semi-supervised learning[A]. 2020.

## 致 谢

2016 年我怀揣着对未来的美好憧憬踏入了清华大学，转眼间已经在清华学习生活了七年。在这七年的时光中，我收获了知识与成长，开阔了视野与认知。在漫漫求索的路途中，遇到诸多艰难险阻，是老师、同学、朋友与家人的帮助与陪伴让我倍感温暖并激励我不断前行。值此毕业之际，向他们表达最诚挚的感谢。

首先，感谢我的导师杨健教授，在本科时期，他是我的班主任，在学业方面给予我很多指导，让我明确的学习的目标与方向。在大三时，非常荣幸保研到了杨健教授师门下，杨健教授在确定科研方向、文献阅读与代码编写等方面对我进行悉心指导，是我科研道路上的引路人。在研究生三年时光中，杨健教授为我提供了企业、医院与研究所等实践机会，指导我参与并完成了多项科研项目，让我的认知与技术能力得到了极大提升。在生活方面，杨健教授亲切随和、亦师亦友，对我们的关心无微不至。杨健教授渊博的学识、精益求精的治学态度、严谨认真的科研作风，为后辈树立了伟岸的学习榜样。

然后，感谢实践基地邃蓝智能科技公司首席曾亮老师，感谢他与公司对本项目的大力支持。曾亮老师在骨髓血细胞检测与识别算法研究中为我提供了专业的理论指导，让我顺利的完成了硕士课题。

接着，感谢实验室的师兄师姐师弟师妹们，与你们在实验室的相处，让我的生活更加丰富多彩，平时的研究交流也让我受益匪浅。感谢王洪淼师兄在深度学习与机器学习方向给予的指导；感谢何耀民师兄在读硕期间的指导与帮助，疫情期间我们被封控在宿舍内，相互扶持鼓励，共同度过了最艰难的一段时光；感谢朱庆涛师兄在数据集处理与文献调研方面给予的帮助与建议。祝大家未来前程似锦，江湖再见！

最后，感谢我的父母，感谢他们多年来的辛勤付出与培养，他们在生活中无微不至的关心与照顾是我勇敢前行的坚实后盾。

## 声 明

本人郑重声明：所呈交的学位论文，是本人在导师指导下，独立进行研究工作所取得的成果。尽我所知，除文中已经注明引用的内容外，本学位论文的研究成果不包含任何他人享有著作权的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。

签 名： \_\_\_\_\_ 日 期： \_\_\_\_\_

## 个人简历、在学期间完成的相关学术成果

### 个人简历

1998年08月30日出生于内蒙古自治区赤峰市。

2016年9月考入清华大学电子工程系电子信息科学与技术专业，2020年7月本科毕业并获得工学学士学位。

2020年9月免试进入清华大学电子工程系攻读电子信息专业硕士至今。

### 在学期间完成的相关学术成果

#### 学术论文：

- [1] 孙天宇, 朱庆涛, 杨健, 曾亮. 基于改进 Vision Transformer 的血细胞图像识别方法研究 [J]. 生物医学工程学杂志, 2022, 39(6):1097-1107(EI, CSCD)

## 指导教师评语

论文提出了……

## 答辩委员会决议书

论文提出了……

论文取得的主要创新性成果包括：

1. .....
2. .....
3. .....

论文工作表明作者在xxxxx具有xxxxx知识，具有xxxx能力，论文xxxx，  
答辩xxxx。

答辩委员会表决，（×票/一致）同意通过论文答辩，并建议授予×××（姓名）  
×××（门类）学博士/硕士学位。