

大连理工大学本科毕业设计(论文)

基于关系嵌入网络的小样本图像分类

Relational Embedding for Few-Shot Classification

学院（系）：电子信息与电气工程学部

专业：电子信息工程

学生姓名：杨题鸣

学号：201883016

指导教师：李培华教授

评阅教师：张立和

完成日期：2022.06.01

大连理工大学

Dalian University of Technology

原创性声明

本人郑重声明：本人所呈交的毕业设计(论文)，是在指导老师的指导下独立进行研究所取得的成果。毕业设计(论文)中凡引用他人已经发表或未发表的成果、数据、观点等，均已明确注明出处。除文中已经注明引用的内容外，不包含任何其他个人或集体已经发表或撰写过的科研成果。对本文的研究成果做出重要贡献的个人和集体，均已在文中以明确方式标明。

本声明的法律责任由本人承担。

作者签名： 杨题鸣 日 期： 2022.06.01

关于使用授权的声明

本人在指导老师指导下所完成的毕业设计(论文)及相关的资料(包括图纸、试验记录、原始数据、实物照片、图片、录音带、设计手稿等), 知识产权归属大连理工大学。本人完全了解大连理工大学有关保存、使用毕业设计(论文)的规定, 本人授权大连理工大学可以将本毕业设计(论文)的全部或部分内容编入有关数据库进行检索, 可以采用任何复制手段保存和汇编本毕业设计(论文)。如果发表相关成果, 一定征得指导教师同意, 且第一署名单位为大连理工大学。本人离校后使用毕业设计(论文)或与该论文直接相关的学术论文或成果时, 第一署名单位仍然为大连理工大学。

论文作者签名: 杨颖鸣 日 期: 2022.06.01

指导老师签名: 李春华 日 期: 2022.06.05

摘要

深度学习的各种模型取得成功的背后需要依靠海量的训练样本，然而在现实中数据样本的获取需要花费大量的人力物力成本，这使得获取海量的训练样本成为奢望。由于大量数据获取难的问题，小样本学习(few-shot learning)的概念被提出，其可以在极少样本情况下达到很好的效果。所以，小样本学习成为了人们关注的焦点，由此本文将围绕小样本学习为研究对象。

在本文中，首先，将构建共计 47951 张图片的垃圾分类数据集，其含有 117 个种类，每类图片数量范围在 208~1150 张之间。其次，基于关系嵌入网络的小样本图像分类(ReNet)模型，对其进行论文复现。在复现基础上，进行了两种类型的探究：1.ResNet-12 残差网络探究，引入局部自注意力与 BotNet 模块，在 CUB 鸟类数据集中最高提升了 0.17%。2.数据增强与快照集成探究，引入了 MixUp、CutMix、Random Erasing、Trivial Augment 等算法。最终在 CUB 和 CIFAR_FS 公开数据集和自制垃圾分类数据集 1/5-shot 中性能平均提升 4.1% 和 2.4%，在 CUB 和 CIFAR_FS 公开数据集中达到了 SOTA 水平。然后，以视觉变换器为小样本学习骨干模型，对 ViT、T2T、Distill-T2T 以及 CCT 模型进行论文复现。在复现基础上，提出 Res9ViT 以及其衍生模型，模型性能全面超过 ViT 在 1/5-shot 中性能平均提升 29.6% 和 23.2%。最后，将 ViT 模型以及 Res9ViT 模型应用到 ReNet 当中，在 CUB 与自制数据集 1/5-shot 中最高提升 1.8% 和 1.2%。

关键词：小样本学习；自建数据集；数据增强；残差网络；视觉变换器

Relational Embedding for Few-Shot Classification

Abstract

The success of various models of deep learning depends on a large number of training samples. However, in reality, the acquisition of data samples requires a large amount of human and material costs, which makes it impossible to obtain a large number of training samples. Due to the difficulty of obtaining a large amount of data, the concept of few-shot learning is put forward, which can achieve good results in the case of few samples. Therefore, few-shot learning has become the focus of people's attention, this thesis will focus on few-shot learning as the research object.

In this thesis, firstly, a garbage classification dataset of 47951 images is constructed, which contains 117 categories, and the number of each category ranges from 208 to 1150 images. Secondly, based on Relational Embedding for Few-Shot Classification(ReNet) , This thesis will reproduce the model. On the basis of reproduction, two types of exploration are carried out: 1. Resnet-12 residual network exploration with introduction of LSA and BotNet module, the maximum improvement in CUB bird dataset is 0.17%. 2. Data augmentation and Snapshot Ensemble exploration with introduction of MixUp、CutMix、Radom Erasing、Trivial Augment and so on, the CUB and CIFAR_FS public datasets and self-made garbage classification datasets have improved by 4.1% and 2.4% in 1/5-shot settings , and SOTA has been achieved in CUB and CIFAR_FS public datasets. Thirdly, the ViT, T2T, Distil-T2T and CCT models are reproduced in this thesis as the few-shot backbone model. Based on reproduction, Res9ViT and its derived model are proposed. The performance of the model is better than ViT by 26% and 23% in 1/5-shot setting. Finally, ViT model and the Res9ViT model proposed in this thesis are applied to ReNet, and the CUB and garbage classification datasets are significantly improved by 1.8% and 1.2%.

Key Words: Few Shot Learning; Self-built Dataset; Data Augmentation; Residual Network; Vision Transformer

目 录

摘要	I
Abstract	II
1 文献综述	1
1.1 课题来源及研究目的和意义	1
1.2 研究方法分类和历史及其现状	1
1.2.1 基于模型微调的小样本学习	1
1.2.2 基于数据增强的小样本学习	2
1.2.3 基于迁移学习的小样本学习	4
1.3 小样本学习主流数据集介绍	7
1.4 主要研究内容与方案及其预期目标	7
1.5 本章小节	8
2 自制垃圾分类数据集	9
2.1 构建目的以及设计思想	9
2.2 数据集构建介绍	9
2.2.1 垃圾的种类选择	9
2.2.2 图片的获取方式	10
2.2.3 图片的清洗	10
2.3 数据集构建过程及其成果	11
2.3.1 图片爬取	11
2.3.2 数据集构建	12
2.3.3 数据清洗	13
2.3.4 数据集类别统计	14
2.4 本章小节	15
3 基于关系嵌入网络的小样本图像分类复现及其改进	16
3.1 基于关系嵌入网络的小样本图像分类复现	16
3.1.1 体系结构概述	16
3.1.2 自相关与互相关表示法	17
3.1.3 学习关系嵌入	19
3.1.4 复现过程及其结果评估	20
3.2 基于 ResNet 的 ResNet-12 骨干模型更改探究	21
3.2.1 探究目的与思路	21

3.2.2 模型介绍.....	22
3.2.3 模型更改探究过程及其结果.....	23
3.3 基于 ReNet 的主流数据增强与快照集成引入.....	25
3.3.1 数据增强图像融合算法 MixUp	25
3.3.2 数据增强图像拼接算法 CutMix	26
3.3.3 数据增强图像擦除算法 Random Erasing	27
3.3.4 数据增强图像随机增强算法 TrivialAugment	28
3.3.5 数据增强 Horizon-Filp, CenterCrop, RandomCrop.....	29
3.3.6 快照集成算法 Snapshot Ensemble	30
3.3.7 实验引入过程及其结果	31
3.4 本章小节	37
4 小样本学习 Vision Transformer 及衍生模型引入及改进	38
4.1 小样本学习 Vision Transformer 及衍生模型引入	38
4.1.1 探究目的与意义	38
4.1.2 基于纯 Transformer 的 ViT 模型	38
4.1.3 基于 Tokens 的 T2T 模型	39
4.1.4 基于知识蒸馏的 Distill-T2T 模型.....	39
4.1.5 基于卷积的 CCT 模型	41
4.1.6 模型复现过程及其结果	41
4.2 小样本学习 Vision Transformer 模型改进	44
4.2.1 探究目的与意义	44
4.2.2 结合卷积算法 Res9ViT 模型	45
4.2.3 结合空洞卷积算法 Dilated-Res9ViT 模型	47
4.2.4 结合 Attention 算法 ResT9ViT 模型	48
4.2.5 模型改进过程及其结果	49
4.3 基于 ReNet 的小样本学习 Vision Transformer 模型	52
4.3.1 探究目的与意义	52
4.3.2 ReNet 替换模型结构图.....	52
4.3.3 结果展示与分析	53
4.4 本章小节	55
结 论	56
参 考 文 献	58

附录 A 自建数据库类别详细目录.....	61
修改记录.....	63
致 谢.....	64

1 文献综述

在本部分首先介绍课题来源及研究小样本学习的目的和意义。其次，介绍研究小样本学习的方法以及主要的分类。最后介绍本论文的主要研究内容以及本论文的研究方案。

1.1 课题来源及研究目的和意义

科技的繁荣迎来数据量的梯度爆炸性增长，数据的增加使得人工智能时代的到来。此时，如今传统的人工智能方法依靠大量的数据，但是往往现实中无法获取如此大的数据量。在样本量很少的情况下，无法通过传统的深度学习网络模型实现不错的效果，常常在数据量太少时，模型会出现过拟合或者欠拟合等严重的问题。研究工作者常常会用各种方法去解决此类问题，比如：数据增强方法、Dropout 方法、数据迁移方法等等。然而除此之外，有一种全新的方法小样本学习就此诞生，其可以在很少的样本下，达到很好的效果。例如一个小孩从没见过大象，但是给他看一张大象的图片，他就能在动物园里快速的识别出大象。这就是人类学习与机器学习的巨大差别，由此受到了人类学习方式的启发。小样本学习核心思想在于教会模型学习的能力，使得它能从大量不同的任务中培养学习的能力，当遇到新的问题时，只需要通过很少的样本就可以学习到丰富的特征，由此实现准确的识别。深度学习当中，由于样本的数量的限制，除了文本分类与图像分类之外，面临着需要采用小样本学习的任务还有许多。小样本问题使得小样本学习的用途十分广泛，利用小样本任务当中场景主要有进行人脸识别，表情识别，手写字体识别，食品识别等等。在自然语言处理当中，也可以进行口语理解，自然语言处理，对话系统等等应用。由此可以发现，小样本学习在生活中处处用到，小样本学习的重要性使得本论文将围绕小样本学习展开研究。由于其诸多的优点并且目前有很多待研究的地方，为此本论文对小样本学习进行研究。

1.2 研究方法分类和历史及其现状

1.2.1 基于模型微调的小样本学习

相对传统的方法，小样本学习可以基于模型微调算法。模型微调的核心思想是在大规模样本上进行预训练，得到权重后在目标小样本学习数据集上对深度神经网络当中的全联接层或者顶端几层进行模型微调。这种模型微调的方式在当今小样本学习当中用得非常多，绝大部分模型都是采用此方法。方法也非常有效，一般能提升 1-2 个百分点。当原始数据集相比于目标数据集样本分布构造类似时，可以达到优秀的效果。Howard^[1]等人提出了一个通用模型微调语言模型(ULMFit)，它的独特特点是：ULMFit 使用了语

言模型而不是深度学习的方法。其模型如图 1.1 所示，可以分成 3 大阶段（1）语言模型预训练；（2）语言模型微调；（3）分类器微调。三大阶段采用的训练方法，使得 ULMFiT 可以很好的在小样本学习当中进行分类，适应不同类型的场景下的小样本学习的下游任务。

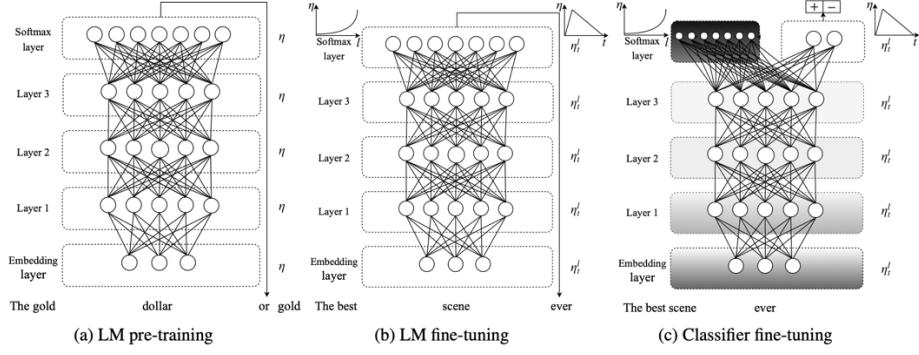


图 1.1 ULMFiT 三个不同阶段的组成

除此之外，Nakamura^[2] 等人提出一种微调的方法，一共可以分成 3 种运行机制：（1）在微调中采用自适 Optimizer；（2）在小样本类别训练中采用很低的学习率；（3）通过全网络调整实现解决原始数据集与目标数据集差异较大问题。

1.2.2 基于数据增强的小样本学习

在小样本学习的中，导致样本数据多样性降低的根本原因是样本量过少，数据增强 (Data Augmentation) 方法可以有效的提高样本丰富性。数据增强可以分成两类，一类是特征增强：在原特征空间中添加一些容易分类的特征。另一类是数据扩充：为增加特征多样性，数据扩充在原数据不变的情况下，根据原数据为基础衍生出带无标签数据数据或标签数据。下面将依次介绍三种主流方法

（1）基于无标签数据的方法

其核心思路是数据扩充将无标签数据利用起来，直推式学习以及半监督学习是其主流方法。近年来有许多研究方法被提出，利用无监督的元训练方法在 2016 年被 Wang^[3] 等人提出，算法使得几个顶层单元接触世界中真实大量的无标注样本从而实现无监督学习。2018 年，能有效进行半监督学习著名的算法 MAML 模型被 Boney^[4] 等人提出。同年 Ren^[5] 等，在原型网络的基础上进行改进，添加无标注样本，模型效果取得进一步提升。直推式学习中算法研究中，在 2019 年转导传播网络被 Liu^[6] 等人提出。随后，Cross Attention Network 被 Cai^[7] 等人提出，如图 1.2 所示。Cross Attention Network 的核心思想

是所有图像先进入一层 ResNet-12 网络后，得到的结果经过交叉注意力机制模块。最后将所得到的结果求出余弦夹角距离，最后计算损失函数的大小以更新模型。

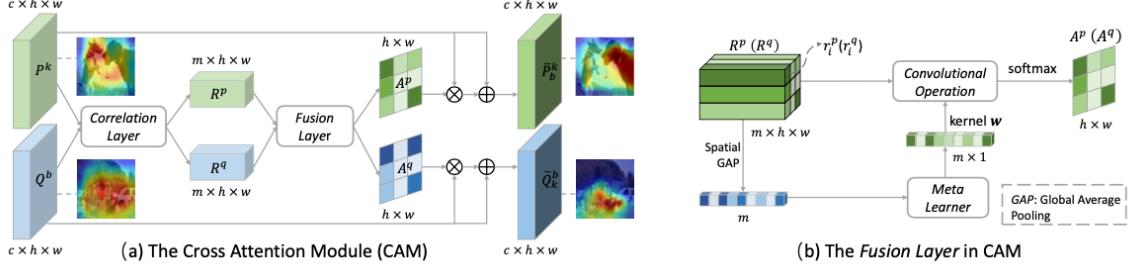


图 1.2 交叉注意力网络基本结构

(2) 基于数据合成的方法

通过带有标签数据实现数据扩充是数据合成的核心思想，生成对抗网络为常用的算法。近年来有许多研究方法被提出，生成对抗残差成对网络被 Mehrotra^[8] 提出，其核心思想是将 GAN 应用到小样本学习中，如图 1.3 所示。Wang^[9] 等人通过端到端方法共同训练生成模型和分类器，这也意味着将数据合成核心思想与元学习的核心理念融合，以丰富数据样本多样性。

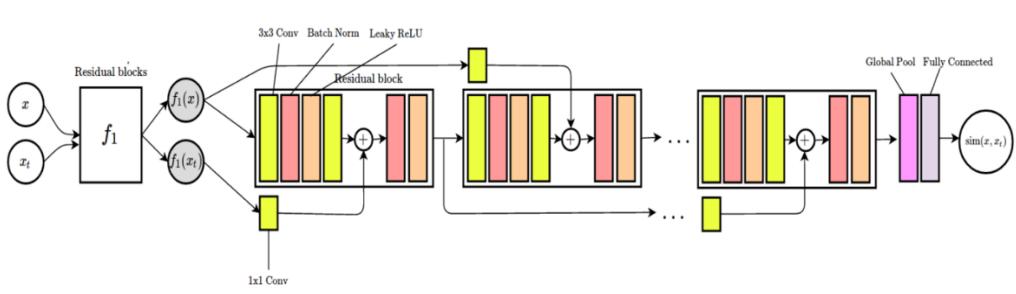


图 1.3 跳跃残差匹配网络(SRPN)

(3) 基于特征增强的方法

特征提取器的好坏与否常常是小样本学习最重要的一个部分，通过提升样本空间丰富性是特征增强的核心理念。Attributed-Guided Augmentation 模型学习被 Dixit^[10] 等人提出，此模型有效的利用特征增强的方法提升了模型的性能。此外，Schwartz^[11] 等人提出的 Delta 编码器成功的利用已有样本生成出新样本，将新得到的样本用于实际的训练过程当中。Chen^[12] 等人提出 TriNet，如图 1.4 所示，通过图像自带的空间特征与标签自带的语义特征的相互结合可以对图像的特征进行增强。

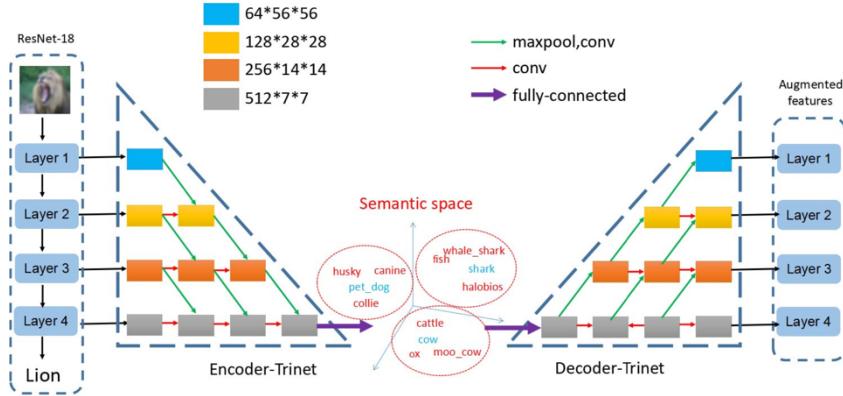


图 1.4 双向网络(TriNet)

1.2.3 基于迁移学习的小样本学习

利用已经学会的知识学习新的知识是迁移学习的核心思想，通过旧知识快速地迁移 到完全新的领域当中是迁移学习主要任务。越来越多的科研人员开始研究迁移学习，许多优秀的算法模型被提出。在迁移学习中，数据集被划分为三部分：(Training Set) 也名叫训练集、(Support Set) 也名叫支持集和(Query Set) 也名叫查询集。迁移学习的三类方法下面进行工作介绍。

(1) 基于度量学习的方法

评判各个要素之间相似性的函数叫做度量，概念常出现在数学领域中。采用 Episodic Training 是基于度量学习的核心方法，其将数据集在训练过程中分为多个 Task，每个 Task 从 Training Set 中任意的选出 C 个类别，每类含有 K 个样本(C -way K -shot)，多个 Task 在多次构建当中被建立起来。训练时，在模型中将多任务依次进行训练。训练样本结构图，如图 1.5 所示。



图 1.5 Training Set 结构图

在测试时，从剩余的样本中选取一个任务进行测试，如图 1.6 所示。



图 1.6 One-shot 预测结构图

Siamese Neural Networks 在 2015 年被 Koch^[13] 等人最先提出。在 2016 年当中，Matching Networks 被 Vinyals^[14] 等人提出。Jiang^[15] 等人基于匹配网络的思路将嵌入函数改进为 4 层的卷积网络。Snell^[16] 等人在 2017 年提出了 Prototypical Networks，模型主要是用深度学习的方法将图像的特征空间变成可以度量的向量，对于同一类别样本计算出向量的平均值即可当成此样本的原型向量。通过不断最小化损失函数和训练模型，使得不同类别的样本向量更为远离，类别相同则相互靠近。原型网络样例示意图如下图 1.7 所示。

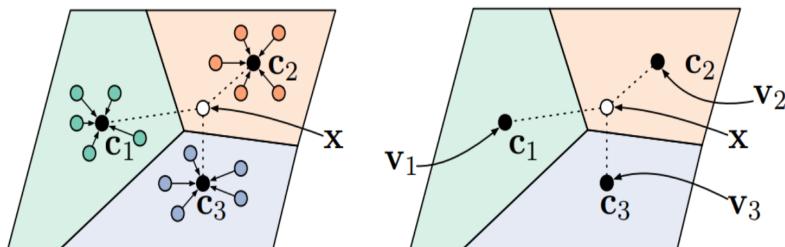


图 1.7 原型网络样例示意图

(2) 基于元学习的方法

Learning to Learn 是元学习的核心，在机器学习领域是一个前沿课题。采用元学习的方法，可以使得模型具有类似于人的能力去学习新的元知识，不仅可以学习模型参数以及结构，还可以学习优化器等等。基于记忆增强的神经网络(Memory-Augmented Neural Networks, MANN)的方法由 Santoro^[17] 等人在 2016 年提出。在 2017 年，Finn^[18] 等提

出了未知模型的元学习方法(Model-Agnostic Meta-Learning, MAML) , 其从很少的数据中进行较少训练即可达到好的效果,如图 1.8 所示。

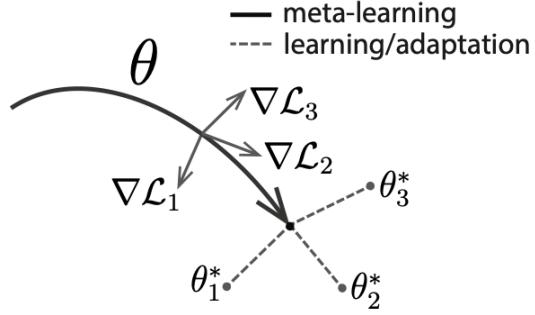


图 1.8 元学习算法(MAML)的示意图

(3) 基于图神经网络的方法

图很早起源于日常生活中, 图在计算机学科当中, 常常出现在数据结构当中, 随着近年来机器学习的兴起, 图也逐渐的被重视起来。图神经网络有常用的有图神经网络(Graph neural networks (GNNs))是通过图 1.9 所示来捕获图的依赖关系的神经网络模型 [19]。近年来, 许多 GNN 的变体, 例如: 图注意力网络(graph attention network (GAT))、图循环网络(graph recurrent network (GRN))、图卷积神经网络(graph convolutional network (GCN))等。

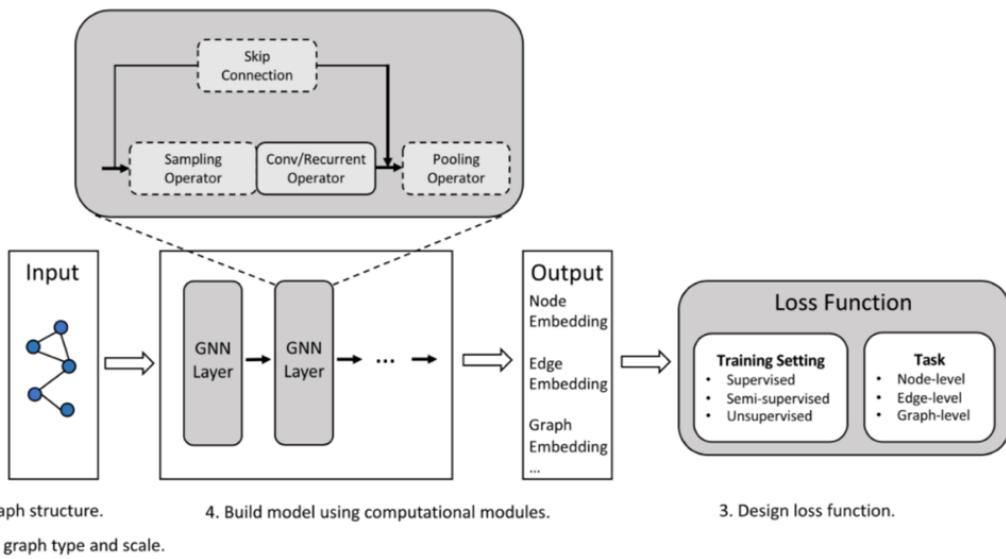


图 1.9 GNN 模型的一般设计流程

1.3 小样本学习主流数据集介绍

一些标准数据集在小样本图像分类任务中被广泛使用。

单样本学习最常用的是数据集 Omniglot^[20]。其包含 50 个字母的 1623 个手写字符, 每字符是由 20 个不同的人在线绘制的。

小样本学习最常用的是 miniImageNet^[21]数据集。其是从 ImageNet 分割得到的, 包含数据集 ImageNet^[22] 每个类别中含有 600 个图像, 一共 100 个类别。一般 64 类用于构成元训练数据集, 16 类用于元验证数据集, 20 类用于元测试数据集。

此外, 常用数据集还有 CUB^[23]、tieredImageNet^[24]等。鸟类图像数据集 Caltech-UCSD Birds(CUB) 共计 11788 张图像, 包含鸟类 200 种, 一般 130 类用于元训练数据集, 20 类用于元验证数据集, 50 类用于元测试数据集。Mengye 等在 2018 年提出的新数据集 tieredImageNet, 其是从 ImageNet 分割得到的, 与 miniImageNet 不同的是, tiered 中有 608 种类别, 类别更多。

1.4 主要研究内容与方案及其预期目标

(1) 研究内容:

为了解整个深度学习以及小样本学习的整体框架, 学习创建一个自制数据集, 其次以基于关系嵌入网络的小样本图像类(ReNet)^[25]为基础模型进行算法探究。视觉变换器模型在小样本学习当中还处于萌芽阶段, 为实现前沿突破探究, 最后尝试以视觉变换器模型为骨干模型进行模型复现以及创新突破。

(2) 研究方案:

通过前面的分析, 本课题研究方案可确定如下:

- 学习计算机视觉基本知识, 了解小样本学习的研究现状;
- 利用爬虫从各大搜索引擎上爬取图像样本, 构造数据集;
- 研究和学习深度学习、小样本学习基本理论、方法, 了解他们的区别与联系;
- 基于当前先进的小样本学习方法进行改进, 并将所设计的方法在公开数据集和自建数据集上进行广泛的实验和分析, 包含图表和可视化分析;
- 撰写毕业论文, 准备毕业答辩。

(3) 预期目标:

- 构造一个大于 100 类垃圾分类训练数据集
- 进行模型性能提升算法设计探究, 主要包含一下切入点:
 - 在 ResNet 的基础上修改网络模型: 增加残差网络, 自相关模型, 互相关模型, 找到更适合小样本学习的改进模型, 目标至少在原有基础上有性能提升。

- 数据增强算法在小样本学习当中的设计，提升模型的鲁棒性，使得小样本学习模型分数的以提升 1 个百分点以上。
- 实现前沿突破探究，进行视觉变换器模型 ViT 以及衍生模型在小样本学习的研究，使得视觉变换器模型超过 ConvNet-4 模型的效果，效果接近甚至超越 ResNet 模型。

1.5 本章小节

在本章当中，首先介绍了课题来源及研究目的和意义，其次介绍了研究方法分类和历史及其现状，小样本学习可以分成基于模型微调、数据增强、迁移学习的方法，随后介绍了小样本学习主流数据集介绍，最后介绍了研究内容与方案及其预期目标。

2 自制垃圾分类数据集

2.1 构建目的以及设计思想

当人类生活生产质量的提升，各种各样的垃圾例如：生活垃圾，工业垃圾数量也在急速增长，垃圾如何正确有效的处理已成为一个非常重要并紧迫的全民社会和经济的大问题。我国各种各样的垃圾产生量迅速增长，生活当中的环境隐患问题逐渐突出，已经成为制约发展的关键因素。遵循资源化、减量化、无害化的原则，实施生活垃圾分类，可以有效改善社会环境，促进资源回收利用，打造美丽社会。近年来，我国高度重视生活垃圾分类处理工作，为了响应国家垃圾分类号召改善社会环境，以及为了加强对数据集的理解，以及通过数据集的构建过程来增加思考问题、解决问题以及工程实践的能力，采用 Python 爬虫构建图像分类数据集。预计将构建一个含 117 种垃圾类别、每类图片数量不均衡的数据库，爬虫结果示意图 2.1 所示。

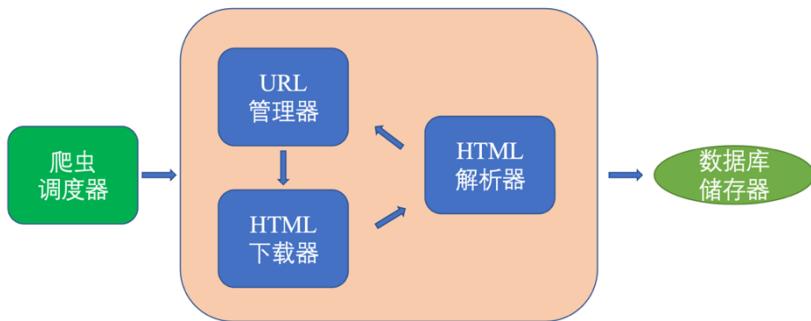


图 2.1 爬虫流程图

2.2 数据集构建介绍

本节当中，首先介绍所构建的垃圾种类，其次，说明图片的获取方式，随后介绍图片的清洗方法，清洗方法当中分成两种不同的清洗方法，在下面逐一介绍。

2.2.1 垃圾的种类选择

根据网络查询，垃圾分为许多类别，根据调研，在本次构建中，将垃圾的种类分成一下四大类，以下这四大类当中是垃圾分类其中的一种形式，可以包含大部分的人类产生的各行各业的垃圾，从当中随机爬取构建数据集的图片：

- 1) 可回收物：可回收物是指可以回收再利用的废品和垃圾。可回收物的主要类型包括：废塑料瓶、废金属、废纸、废包装、废玻璃、废纸塑料铝复合包装等；

- 2) 有害垃圾：通常是指在日常生活当中会对人体造成很严重危害的一些化学物理制品的废弃物，比如电磁，废弃药品，农药等等，这一些物品常常被归类为有害垃圾；
- 3) 厨余垃圾：在日常生活中，最常见的便是厨余垃圾，不仅仅是在家庭当中，还在餐饮服务业通常会产生，比如：剩菜剩饭、骨头、蛋壳等，其主要来自家庭厨房、公共食堂、餐厅等与食品加工相关的行业；
- 4) 其他垃圾：指除可回收物、危险废物和厨房废物外的其他生活垃圾，如无汞电池、卫生纸、烟头、瓷砖、陶瓷等其他难以回收的废物。

2.2.2 图片的获取方式

网络爬虫(Web Spider)是一种自动扫描网络的软件。它通常被设计用来收集资源(Word 文档、网页、视频、图像、PostScript 或 PDF 等)，以便搜索引擎对它们进行索引。通过 python 爬虫，结合 Request，Numpy 等库对以各大搜索引擎进行图片爬取。

在本文中，数据主要采用百度进行图片收集。利用百度的图片搜索引擎的原因在于百度作为国内的最大的搜索引擎，能爬去丰富的图片数据。又考虑到 Google 在国内无法访问，所以百度图片成为本论文爬虫的首选搜索引擎。在后续的爬虫当中，均以百度图片作为爬虫的对象。

值得注意的是在在图片爬取过程中，一定要注意爬取图片的速度。当速度过快时，会对服务器造成访问压力。应用爬虫就要合理适当的应用，遵守国家法律。通常在获取时，每次图片访问的速度控制在 0.1 秒每张，这样一方面既能保证对服务器访问压力较小，又能保证获取图片的速度。

2.2.3 图片的清洗

为了尽量提高数据集的质量，对图片进行两次清洗。第一次验证图片完整性，主要将错误空白图片进行删除；第二次清洗，采用深度神经网络删除错误图片。

验证图片完整性：由于爬虫有时候爬取的图片会存在错误，通常会将一些不是图片的文件当成图片而被爬取下来，为了验证图片完整性，可以用扩展名判断文件格式，jpg 文件有固定的文件头，其文件为：“\xff\xd9”。通过读取文件的内容，判断其真实的类型，即可将真实图片保存下来。

深度神经网络删除错误的图片：所有图片进入卷积神经网络训练，用得到的训练权重对所有图像预测，预测值与其真实标签比较，若判断其类别与图片类别不同，则删除此错误图片。

通过两次的数据清洗，能尽可能保证数据的可靠性。模型性能的好坏与数据集的好坏有着非常紧密的联系，数据的清洗为后续实验的稳定打下坚实基础。

2.3 数据集构建过程及其成果

2.3.1 图片爬取

在数据集构建当中，爬虫爬取可分为以下几个步骤：

(1) 获取图片的 URL 以及定义 Headers 头

图片获取的 URL 可由如下公式 (2.1) 所示

$$\text{url} = \text{'website'} + \text{name} + \text{'&pn='} + \text{str(page)} \quad (2.1)$$

其中，website 表示将百度图片的地址，name 表示图片的名字，page 表示从哪一页开始。

定义 Headers 头的目的在于让爬虫模拟浏览器，在本论文当中，将浏览器头设为公式 (2.2) 所示

$$\text{'User-Agent': 'Mozilla/5.0 AppleWebKit/537.36 Chrome/84.0.4147.125'} \quad (2.2)$$

(2) 发起 Requests 请求

有了 url 和 headers 后，用 Import 导入模块 Requests，并由如下公式发起 Get 请求，发起请求的公式如 (2.3) 所示

$$\text{res} = \text{requests.get(url, headers=headers)} \quad (2.3)$$

(3) 数据保存

当 Requests 请求回应 200 后表示接收成果，此时 For 循环找到图片位置，并用公式

(2.4) 提取出来

$$\text{img} = \text{requests.get(img_content)} \quad (2.4)$$

其中，img_content 表示图片在 Requests 请求的 Html 当中的位置。

通过百度查询四种垃圾分类的详细类别，选定了数据集分类详细情况，如图 2.2 所示展示了数据集当中部分垃圾分类的文件可视化展示图。

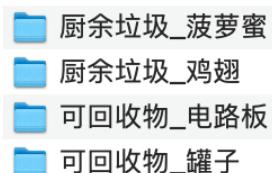


图 2.2 垃圾分类部分类别名称

2.3.2 数据集构建

(1) 中英转换

数据集构建当中，将清洗后的数据进行格式构建，构建过程中，需要将中文翻译成英文，在此应用了百度翻译 API 有效解决数据集翻译问题，整体流程如下图 2.3 所示。

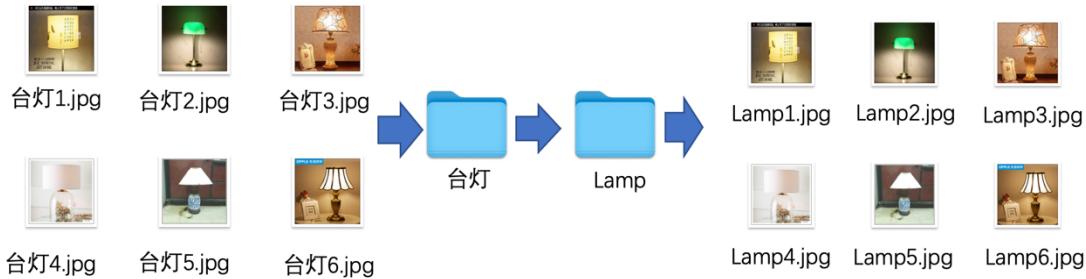


图 2.3 爬虫构建数据集中英翻译结构图

(2) 文件夹结构构造

通过上述构造，数据集一共包含 117 个小类和 4 个大类，4 个大类分别是可回收垃圾、厨余垃圾、有害垃圾和其他垃圾，图片分为 3 个数据集分别是 Meta-Train，Meta-Val，Meta-Test。与传统的图像分类数据集结构不同，小样本学习当中 Meta-Train，Meta-Val，Meta-Test 种类各不一样，3 个数据集当中的图片均为 jpg 格式。文件结构如下图 2.4 所示。

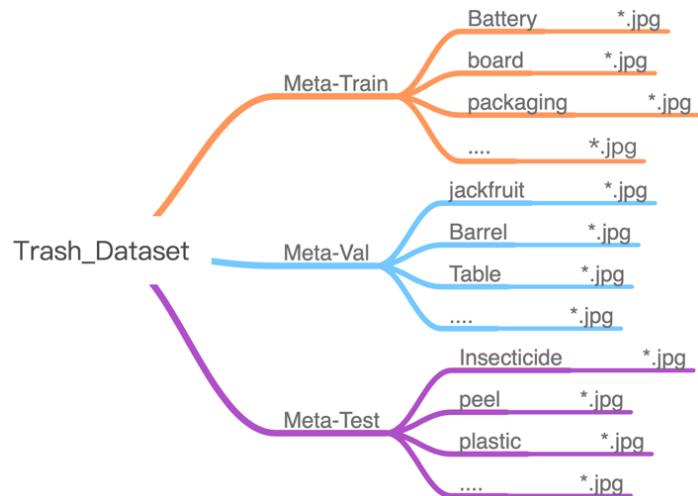


图 2.4 爬虫构建数据集可视化

2.3.3 数据清洗

验证图片完整性：在 2.3.1 节图片爬取过程中，通过将图片变成文本 base64 的格式形式，提取出文本类别的头，通过文件类别的头判断文件扩展名格式时就已经删除错误图片，如下图 2.5 所示：



图 2.5 左：正确图片 右：错误图片

深度神经网络删除错误的图片：通过神经网络，可以初步的判断图片的精确度。先将所有图片进入卷积神网络 ResNet-50 进行训练，获得训练权重后。用得到的训练权重加载并对所有图像进行预测，得到的预测值与其真实标签进行比较，若发现神经网络判断其类别与图片类别不同，则删除此错误图片，其流程图 2.6 所示：

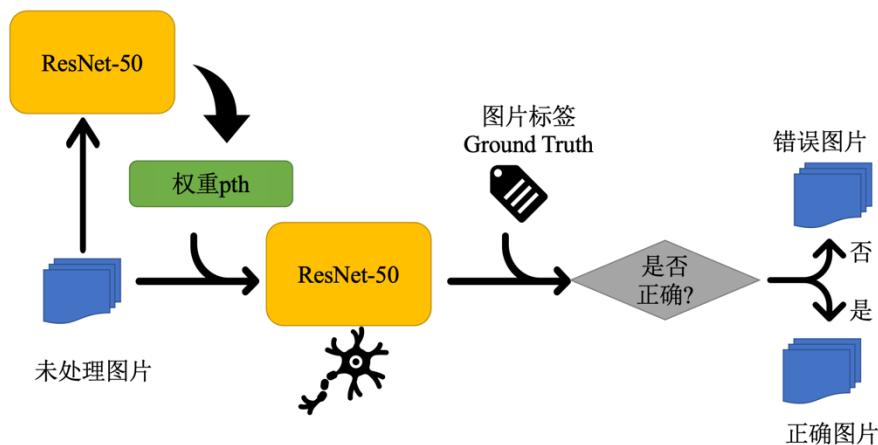


图 2.6 神经网络数据清洗流程图

一共训练数据集得到的损失函数收敛曲线与精确度曲线如下图 2.7 所示。选用的网络为 ResNet-50 模型，一共训练了 76 个 Epoch，选用的优化器为 Adam，学习率为 0.001。随后采用训练好的训练权重，删除掉在爬虫爬取时由于类别而错误的一些图片。但值得注意的是，采用网络进行数据清洗时，需要进行一次手动清洗，这样才能保证在深度神经网络删除错误的图片准确性。

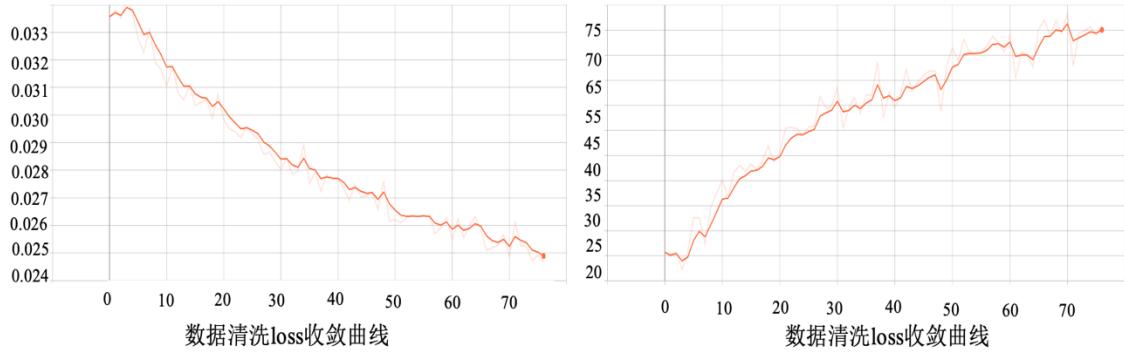


图 2.7 神经网络数据曲线

2.3.4 数据集类别统计

将其中部分的图像展示为如下图 2.8 所示。

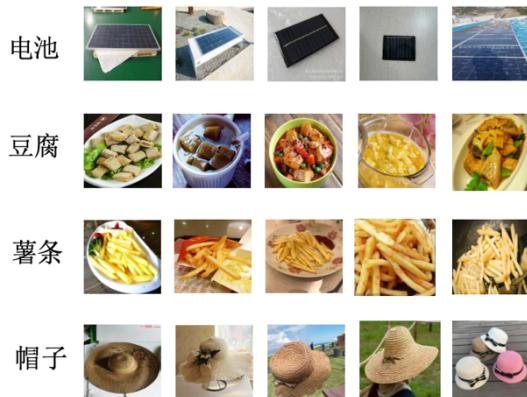


图 2.8 爬虫构建数据集可视化

如下表 2.1 为自建数据集的部分参数，包括有：Meta-Train，Meta-Val，Meta-Test。以及他们各自含有的类别数量，图片数量，所包含的范围，图片的大小。所有数据详细的参数可以在附录 A 自建数据库类别详细目录中查看，在附录 A 中包含种类的名称和种类的数量

表 2.1 自建数据集参数配置

Datasets	Class Num	Image Num	Range	Size
Meta-Train	75	30915	208~1550	
Meta-Val	18	7601	237~857	224×224
Meta-Test	24	9435	216~1021	

将所有数据集 Meta-Train, Meta-Val, Meta-Test 三者进行统计，可以得到如图 2.9 所示的数据类别统计图。由图可以看出，每一种数据都不均衡，目的就是为了模拟生活中各种各样的情况，以模拟生活中数据样本不均衡的问题。

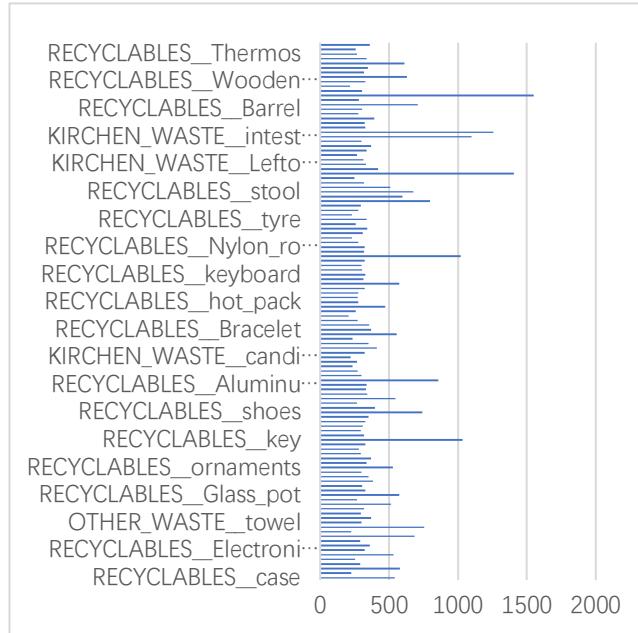


图 2.9 爬虫构建数据集类别统计图

2.4 本章小节

本章主要介绍了垃圾分类数据集的制作流程，首先介绍了制作目的以及设计思想，随后介绍数据集构建的思路及其方法，分成三个部分进行：垃圾的种类选择，图片的获取方式，图片的清洗。最后，介绍数据集构建过程及其成果，此部分分层三个模块：图片爬取，数据集构建(中英转换，文件夹结构构造)，数据清洗，数据集类别统计。

3 基于关系嵌入网络的小样本图像分类复现及其改进

3.1 基于关系嵌入网络的小样本图像分类复现

本文将以基于关系嵌入网络的小样本图像分类(Relational Embedding for Few-Shot Classification, ReNet)作为主要模型，基于度量学习和元学习的方法进行小样本学习。从关系的角度出发，通过元学习来解决少样本分类问题。方法通过自相关表征 (SCR) 和交叉相关注意 (CCA) 来控制图像内部与之间的关系模式。在每幅图像中，SCR 模块将基本特征映射转换为自相关张量，并从张量中提取特征结构。在图像之间，CCA 模块计算图像之间的互相关特征，并学习在它们之间产生共同注意机制特征。关系嵌入网络 (ReNet)结合了两个关系模块，以端到端的方式学习关系嵌入。

3.1.1 体系结构概述

给定一对查询集(Query Set)和一张支持集(Support Set)图像 \mathbf{I}_q 和 \mathbf{I}_s ，骨干模型 ResNet-12 作为特征提取器提供基表示， \mathbf{Z}_q 和 $\mathbf{Z}_s \in \mathbb{R}^{H \times W \times C}$ 。SCR 模块通过卷积分析图像表示中的相关性特征，将基表示转换为自相关表示， \mathbf{F}_q 和 $\mathbf{F}_s \in \mathbb{R}^{H \times W \times C}$ 。然后 CCA 模块采用自相关表示生成共同注意映射 \mathbf{A}_q 和 $\mathbf{A}_s \in \mathbb{R}^{H \times W}$ ，将聚合 \mathbf{F}_q 和 \mathbf{F}_s 的空间注意权重赋予图像嵌入特征， q 和 $s \in \mathbb{R}^C$ ，CCA 模块并行应用于所有支持集图像 $\mathbf{I}_s \in \mathcal{S}$ ，然后将查询集(Query Set)分类为其距离最近的支持集(Support Set)嵌入类，将基表示特征 \mathbf{Z}_q 和 \mathbf{Z}_s 转换为自相关张量 \mathbf{R}_q 和 \mathbf{R}_s ，然后通过卷积块 g 更新为自相关表示 \mathbf{F}_q 和 \mathbf{F}_s 。对图像表示之间计算互相关 C ，然后通过卷积块 h 细化到 $\hat{\mathbf{C}}$ ，双向聚合生成共同注意映射 \mathbf{A}_q 和 \mathbf{A}_s 。将这些共注意映射应用于对应的图像表示形式 \mathbf{F}_q 和 \mathbf{F}_s ，并对其参与特征进行聚合，分别生成最终的关系嵌入 q 和 s 。如图 3.1 所示的过程展示了基于关系嵌入网络的小样本图像分类整体结构。

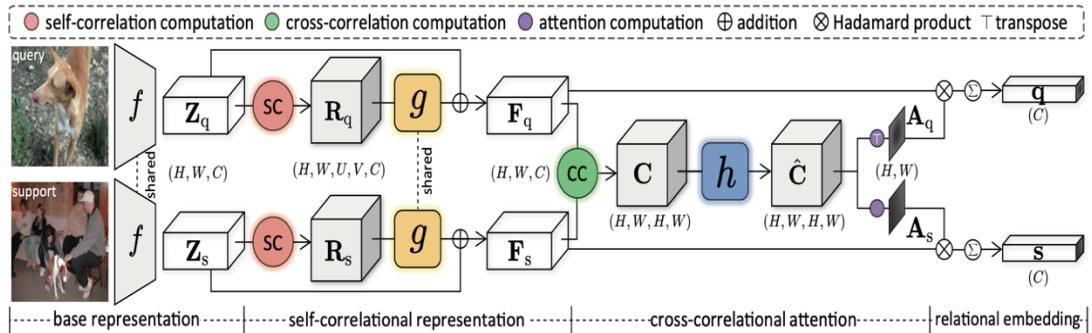


图 3.1 ReNet 的总体架构

3.1.2 自相关与互相关表示法

本节主要解释在 ReNet 模型当中采用的自相关 SCR 与互相关 CCA 表示法。图 3.2a 说明了 SCR 模块的基本结构架构。(a): SCR 模块通过将其在 $U \times V$ 维上卷积来捕获输入自相关 \mathbf{R} 中的特征，将结果 $g(\mathbf{R})$ 添加到基表示特征 \mathbf{Z} 中，与 \mathbf{Z} 相加后形成自相关表示 \mathbf{F} 。(b): CCA 模块将互相关特征归纳为共注意映射 \mathbf{A}_q 和 \mathbf{A}_s 。

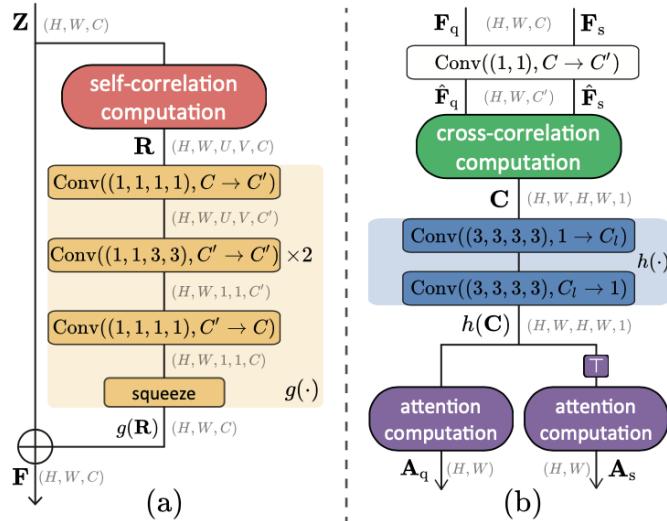


图 3.2 SCR 和 CCA 模块的架构

(1) 自相关表示法

SCR 模块采用基表示特征 \mathbf{Z} 并将特征 \mathbf{Z} 转换为含有自注意力机制的特征图 \mathbf{F} ，即 SCR 模块为 CCA 模块准备可靠的输入，CCA 模块分析一对查询集特征图和一张支持集特征图像之间的特征相关性。

自相关计算：

给定一个基表示特征 $\mathbf{Z} \in \mathbb{R}^{H \times W \times C}$ ，论文计算 c 维向量在 $x \in [1, H] \times [1, W]$ 及其邻域的每个位置的 Hadamard 矩阵乘积，并将它们变成一个自相关张量 $\mathbf{R} \in \mathbb{R}^H \times W \times U \times V \times C$ 。张量 \mathbf{R} 可以表示为一个带有 c 维向量输出的函数，如公式 (3.1) 所示：

$$\mathbf{R}(x, p) = \frac{\mathbf{z}(x)}{\|\mathbf{z}(x)\|} \odot \frac{\mathbf{z}(x+p)}{\|\mathbf{z}(x+p)\|} \quad (3.1)$$

其中 $p \in [-d_U, d_U] \times [-d_V, d_V]$ 对应于邻域窗口中 $2d_U + 1 = U$ 、 $2d_V + 1 = V$ 的相对位置，包括中心位置。在特征图的边缘卷积前采用零填充用于从边缘取样。在自相关后降维时，采用像素点与其周围点之间相乘，随后用卷积的方法，将维度降到同基特征 $\mathbf{Z} \in \mathbb{R}^{H \times W \times C}$

相同的维度，也即后续自相关表征学习所介绍。与先前论文的方法中将一对特征向量简化为标量相关值不同，ReNet 采用了通道方向的相关，保留了特征向量丰富的语义进行分类。

自相关表征学习：

为了分析 R 中的自相关模式，ReNet 应用了一系列沿着 $U \times V$ 维的二维卷积。如图 3.2a 所示，卷积块结构包括一个用于减小信道尺寸的点卷积层，两个用于变换的 3×3 卷积层，以及一个用于恢复信道尺寸的点卷积层。在卷积之间插入批处理归一化 BN 和 ReLU。这种卷积块 $g(\cdot)$ 使其空间维数从 $U \times V$ 降至 1×1 使输出 $g(R)$ 具有与 Z 相同的大小，即 $g: \mathbb{R}^{H \times W \times U \times V \times C} \rightarrow \mathbb{R}^{H \times W \times C}$ 。这种结构产生的特征作为基表示特征 Z 中出现模式的补充。因此，ReNet 将这两个表示结合起来，产生自相关表示 $F \in \mathbb{R}^{H \times W \times C}$ ，如公式 (3.2) 所示：

$$F = g(R) + Z \quad (3.2)$$

这加强了基特征与关系特征的联系，并帮助少样本学习更好地使得模型理解图像所需要关注的点。ReNet 实验表明，SCR 对类内变化是具有鲁棒性的，并有助于推广到看不见的目标类。

(2) 互相关表示法

CCA 模块输入为一对经过 SCR 的查询集与支持集，通过互相关计算以及 4D 卷积，生成相应的注意图 A_q 和 A_s ，每个空间注意力特征图将与基表示聚合为一个嵌入向量。图 3.2b 可可视化了 CCA 模块的结构。

互相关计算：

论文首先将查询集表示和支持集表示 F_q 和 $F_s \in \mathbb{R}^{H \times W \times C}$ 使用点卷积层转换为更紧凑的表示，将其通道维 C 减少到 C' 。由输出 \hat{F}_q 和 $\hat{F}_s \in \mathbb{R}^{H \times W \times C'}$ 构造一个四维相关张量 $C \in \mathbb{R}^{H \times W \times H \times W}$ ，如公式 (3.3) 所示：

$$C(x_q, x_s) = \text{sim}(\hat{F}_q(x_q), \hat{F}_s(x_s)), \quad (3.3)$$

式中， x 为特征图上的空间位置， $\text{sim}(\cdot, \cdot)$ 为两个特征间的余弦相似度。

卷积的匹配：

互相关张量 C 可能包含不可靠的相关，这是由于在少样本学习设置中有很大的外观变化。为了消除这些不可靠匹配的歧义，ReNet 采用了卷积匹配过程，通过匹配核进行 4D 卷积来细化张量；张量的四维卷积通过分析四维空间中相邻匹配的一致性，起到了几何匹配的作用。如图 3.2b 所示，卷积匹配块 $h(\cdot)$ 由两个 4D 卷积层组成；第一次卷积产生多个匹配核的多个相关张量，将通道大小增加到 C_1 ，第二次卷积将它们聚合为一个四维

相关张量，即它们为一个四维相关张量，即 $\hat{\mathbf{C}} = h(\mathbf{C}) \in \mathbb{R}^{H \times W \times H \times W}$ 。同时注意，在卷积层之间需要归一化 BN 和激活层 ReLU。

共同注意力计算：

从提炼后的张量 $\hat{\mathbf{C}}$ 开始，生成了共同注意映射特征 \mathbf{A}_q 和 \mathbf{A}_s 揭示了查询集和支持集之间的相关性。查询集 $\mathbf{A}_q \in \mathbb{R}^{H \times W}$ 的注意映射由公式 (3.4) 计算

$$\mathbf{A}_q(x_q) = \frac{1}{HW} \sum_{x_s} \frac{\exp(\hat{\mathbf{C}}(x_q, x_s)/\gamma)}{\sum_{x'_q} \exp(\hat{\mathbf{C}}(x'_q, x_s)/\gamma)} \quad (3.4)$$

其中 x 为特征图上的位置， γ 为温度因子。 $\hat{\mathbf{C}}(x_q, x_s)$ 是匹配分数 x_q 和 x_s 的计算余弦距离的函数，支持集度 \mathbf{A}_s 的注意映射类似 \mathbf{A}_q 地通过在式(3.4)中查询集和支持集来计算。

这些共同注意特征图通过元学习交叉相关模式来提高少样本分类的准确性。

3.1.3 学习关系嵌入

在本小节中，论文从 \mathbf{F}_q 和 \mathbf{F}_s , \mathbf{A}_q 和 \mathbf{A}_s . 推导关系型嵌入 $q \in \mathbb{R}^c$ 和 $s \in \mathbb{R}^c$ 。然后论文通过描述学习目标来总结 learning objective 方法。

注意池化：

为了得到查询集的注意嵌入 $q \in \mathbb{R}^c$, $\mathbf{F}_q \in \mathbb{R}^{H \times W \times c}$ 的每个位置乘以空间注意映射 $\mathbf{A}_q \in \mathbb{R}^{H \times W}$, 然后池化, 如公式 (3.5) 所示:

$$q = \sum_{x_q} \mathbf{A}_q(x_q) \mathbf{F}_q(x_q) \quad (3.5)$$

注意, A_q 的元素和为 1, 因此注意嵌入 q 是在支撑环境下参与的 F_q 的凸组合。关系嵌入特征 s 随是通过计算 $A_s(x_s) F_s(x_s)$ 相乘的方法计算出的, 即通过 A_s 加入支持集特征图 F_s , 然后加入池化, 如公式 (3.6) 所示:

$$s = \sum_{x_s} \mathbf{A}_s(x_s) \mathbf{F}_s(x_s) \quad (3.6)$$

在 N -way K -shot 分类设置中, 该共注意池化生成查询集的 NK 不同视图集合 $\{q^{(l)}\}_{l=1}^{NK}$, 以及在查询集上下文中参与的支持集嵌入集合 $\{s^{(l)}\}_{l=1}^{NK}$ 。

Learning objective:

ReNet 是端到端的模型。而最近的小样本学习分类方法大多采用二阶训练的方法，初始训练前和随后的情景训练的两阶段训练方案，ReNet 采用单阶段训练方案，结合两种损失函数对所提出的模块和骨干网络进行联合训练: 基于 anchor 的分类损失 $\mathcal{L}_{\text{anchor}}$ 和基于度量的分类损失 $\mathcal{L}_{\text{metric}}$ 。首先，在基表示特征 z_q 之上使用一个附加的全连接分类层计算 $\mathcal{L}_{\text{anchor}}$ ，如公式 (3.7) 所示。这种损失指导模型正确地对 $c \in \mathcal{C}_{\text{train}}$ 查询集进行分类。

$$\mathcal{L}_{\text{anchor}} = -\log \frac{\exp(w_c^\top z_q + b_c)}{\sum_{c'=1}^{|C_{\text{train}}|} \exp(w_{c'}^\top z_q + b_{c'})} \quad (3.7)$$

其中 $[w_1^\top, \dots, w_{|C_{\text{train}}|}^\top]$ 和 $[b_1, \dots, b_{|C_{\text{train}}|}]$ 分别是全连接层中的权重和偏差。接下来，通过查询集和支持集原型嵌入之间的余弦相似度计算基于度量的损失 $\mathcal{L}_{\text{metric}}$ 。在计算损失之前，论文对 K 个查询集结果进行平均。在计算损失之前，论文对 n^{th} 类的 k^{th} 支持集下的上下文嵌入向量中的 K 个查询集嵌入向量进行平均，计算 $\{\bar{q}^{(n)}\}_{n=1}^N$ 。类似地，ReNet 平均每个类的 K 个支持集嵌入得到一组原型嵌入： $\{\bar{s}^{(n)}\}_{n=1}^N$ 。基于度量的损失函数引导模型将查询集嵌入映射靠近同一类的原型嵌入，如公式 (3.8) 所示：

$$\mathcal{L}_{\text{metric}} = -\log \frac{\exp\left(\frac{\text{sim}(\bar{s}^{(n)}, \bar{q}^{(n)})}{\tau}\right)}{\sum_{n'=1}^N \exp\left(\frac{\text{sim}(\bar{s}^{(n')}, \bar{q}^{(n')})}{\tau}\right)} \quad (3.8)$$

式中， $\text{sim}(\cdot, \cdot)$ 为余弦相似度， τ 为标量温度因子。在推理时，查询集的类被预测为最近的原型的类。该目标结合了两项损失，如公式 (3.9) 所示：

$$\mathcal{L} = \mathcal{L}_{\text{anchor}} + \lambda \mathcal{L}_{\text{metric}} \quad (3.9)$$

λ 是一个超参数，平衡损失权重。注意，计算 $\mathcal{L}_{\text{anchor}}$ 所涉及的全连接层在推理过程中被丢弃。

3.1.4 复现过程及其结果评估

(1) 实验环境说明

在本节中主要展示 ReNet 模型的基本复现过程以及结果。由于 ReNet 论文是在 RTX 3090 上面进行实验，为保持结果尽可能与论文一致，本次复现在云服务器 3090 上面进行，选用的服务器为矩池云。复现参数可以列为如下表 3.1 所示，在本章实验当中，均采用此配置方法。

表 3.1 ReNet 复现参数

CONFIG	1-shot	5-shot
Epoch	80	60
Milestone	60	40
Optimizer	SGD	SGD
Learning Rate	0.1	0.1
Val Episode	200	200
Test Episode	2000	2000

(2) 复现结果评估

在复现过程中,需要的数据集有: MiniImageNet, CUB, CIFAR_FS, TieredImageNet,但是由于 TieredImageNet 数据集非常的大,需要耗费的时间和金钱成本太过于重,所以在本次实验当中,不会将 TieredImageNet 进行实验。实验结果如下表 3.2 所示, PAPER 表示原始论文的数据, OURS 表示本论文在矩池云服务器上面采用 RTX 3090 运行的结果。从结果可以有效的说明, ReNet 的论文复现效果一致, 效果正常。那么可以在这个模型的基础上进行各种实验探究

表 3.2 ReNet 基本复现结果

Method	CUB		CIFAR_FS		MINIIMAGENET	
	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
ReNet(PAPER)	79.49	91.11	74.51	86.60	67.60	82.58
ReNet(OURS)	79.20	91.10	74.64	86.67	67.33	82.38

3.2 基于 ReNet 的 ResNet-12 骨干模型更改探究

3.2.1 探究目的与思路

在小样本学习中,经常出现的模型一共有 ConvNet-4, ResNet-12, WRN-28-10, ResNet-18 等, 其中最常用的是 ResNet-12。在基于关系嵌入网络的小样本图像分类论文里面采用的是 ResNet-12 网络。

近十年来,在全球计算机视觉中卷积神经网络一直在研究领域保持着重要主导作用。Google 提出一种新方法,想要利用视觉变换器的功能赋予图像更深层的意义。一开始视觉变换器模型应用在自然语言处理当中,而非计算机视觉领域,在 2018 年 ViT 模型的提出使得视觉变换器的注意力机制在计算机视觉领域也被广泛应用。在深度学习当中,视觉变换器论文已经有很多,但是目前在小样本学习当中视觉变换器相关的论文却很少。由此在 ResNet-12 的基础上,将目前一些著名的深度学习模型视觉变换器应用到小样本学习当中。

值得注意的是: 在后续章节当中,也会对视觉变换器进行讨论,但是与本章节的讨论内容出发点并不同,在本章节当中,是基于 ResNet-12 骨干模型的视觉变换器探究。但是第 4 章当中,是以 ViT 模型为骨干进行探究。两种讨论看似相似,其出发点以及本质有着区别。

3.2.2 模型介绍

(1) Bottleneck-Transformer (BotNet) 介绍

比如 GoogLeNet^[26]、ResNet 等基于深层卷积的骨干网络结构在实例分割方面、目标检测、图像分类取得了重大进展，骨干网络大多数都是基于 3×3 的卷积核。由于关键点检测、实例分割、目标检测等任务需要对远程依赖关系数据建模，尽管 3×3 的卷积核运算有效地捕捉局部信息，对于一些全局信息难以捕捉。基于卷积的网络结构常要多层次目的是为了能够通过聚合局部来获得全局信息。更多层确实提升骨干网络的性能，但直接建立一个能够建模全局关系模型是更加合理的方便的，因为它不需要多层卷积的堆叠，只需要一个结构就能进行全局信息的建模。由下图 3.3 所示描述了自注意层的结构，左边为 ResNet 结构，右边为 BotNet^[27]结构。唯一的区别是用 Multihead Self-Attention (MHSA^[28])替换了空间 3×3 卷积层。

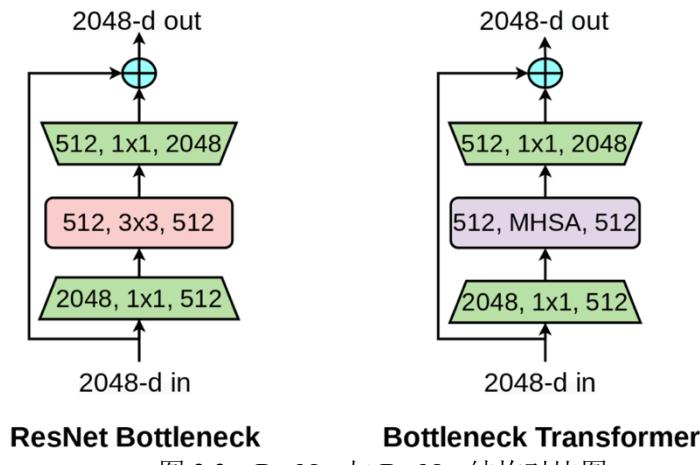


图 3.3 ResNet 与 BotNet 结构对比图

(2) 局部自注意力机制 (LSA) 介绍

卷积运算是视觉任务的基本组成部分，具有一定的局域局限性。在计算机视觉任务中，为了通过基于内容的交互计算来捕捉引入的长程依赖，注意力块通常作为卷积的一种增强操作引入。以前没有人试图把注意力作为一个基本的模块，LSA^[29]论文尝试构建一个全注意网络来代替卷积运算，并验证了自我注意可以作为网络中单独的一层。卷积中的归纳偏差是局部性的，局部性是指空间相似的网格特征之间的连接，而距离较远的网格之间不存在连接。在本文中，卷积运算被基于内容的自注意力机制所取代。实际上，LSA 当中的视觉变换器中的运算被限制在一个卷积核上， \mathbf{Q} 、 \mathbf{K} 、 \mathbf{V} 的三个映射是通过 3 个 1×1 卷积核的 2D 卷积得到。图 3.4 所示，为局部自注意力机制在注意力模块的 \mathbf{Q} 、 \mathbf{K} 、 \mathbf{V} 结构图。

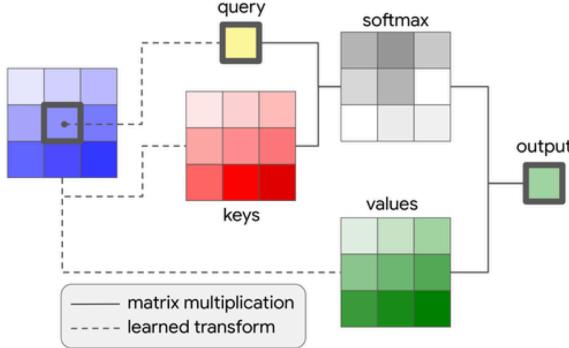


图 3.4 局部自注意力机制(LSA)

3.2.3 模型更改探究过程及其结果

在本节当中，采用的基本训练配置与前实验配置相同，具体详情，请看表 3.1

(1) BotNet 的引入

由此，Google 提出了一个名叫 Bottleneck-Transformer 的模型，只需要在 ResNet 的第四层当中加入了 MHSAs 模块将 ResNet 变成了 BotNet，其实验效果在 ResNet-50 与 ResNet-101 取得了不错效果，出于这个想法，将 ResNet-12 当中融合 BotNet 模型。在本论文当中，把此网络叫做 BotNet-12，其的流程图 3.5 所示。图片首先进入到正常的 ResNet-12 网络当中的前 3 层，网络的深度逐渐加深，分别为 [64, 160, 320]。在进入第四层经过一次卷积操作后，随后进入 MHSAs 模块，其余地方与 ResNet-12 一致。

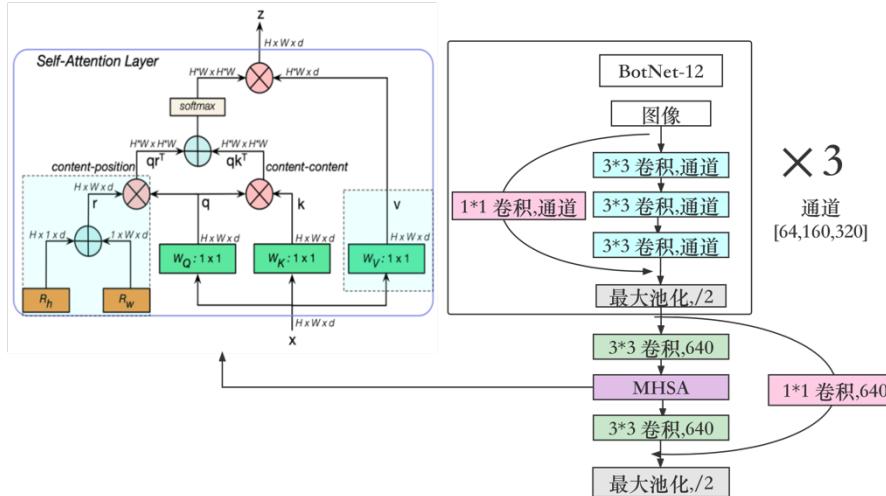


图 3.5 BotNet-12 结构图

(2) 局部自注意力机制的引入

与 BotNet-12 类似，并不需要更改太多的结构，只需要在 ResNet-12 的最后一层结尾处添加一个 LSA 模块。此 LSA 模块当中，不同于其他的局部自注意力机，它的 Q 、 K 、 V 矩阵由卷积产生。ResNet-12+LSA 的结构图 3.6 所示。

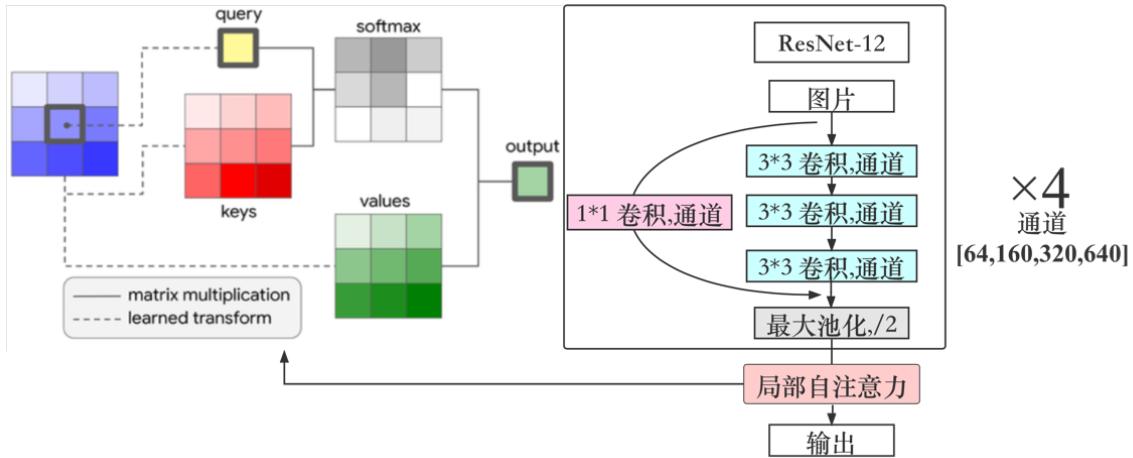


图 3.6 左：LSA 模块 右：ResNet-12+LSA 结构图

(3) 探究结果展示

在细粒度鸟类数据集 CUB 上，5-Way 1-Shot 的情况下，在基于关系嵌入网络的小样本图像分类(ReNet)模型中，一共进行了 5 次实验。ReNet 的基本结构为：ResNet-12+SCR+CCA。其中 SCR 表示的是自相关模块，CCA 表示互相关模块。图 3.7 中 Origin 代表的是 ReNet 原始参数情况下得出的结果，在表 3.3 中用 ResNet 表示。在 ReNet 原始参数的情况下，在自相关模块当中添加了 LSA 自注意力模块，在表 3.3 中用 ResNet+LSA+SCR 表示。在 ReNet 原始参数的情况下，将 LSA 模块替换掉 SCR 模块，在表 3.3 中用 ResNet+LSA 表示。在 ReNet 原始参数的情况下，改动 ResNet-12 模块，将 ResNet 的四层中各种添加 MHSA 模块，其 BotNet-L1 表示在 ResNet-12 第一层添加 MHSA；其 BotNet-L3 表示在 ResNet-12 第三层添加 MHSA 模块；其 BotNet-L4 表示在 ResNet-12 第四层添加 MHSA 模块。

表 3.3 在细粒度鸟类数据集 CUB 上对模型架构的改进实验的评估(基于 ReNet)

BackBone	ResNet	ResNet+ LSA+SCR	ResNet + LSA	BotNet- L1	BotNet- L3	BotNet- L4
5-way-1 shot	79.197	78.899	79.337	78.978	78.934	78.837

如下图 3.7 左图所示，展示的是在细粒度鸟类数据集 CUB 上对模型架构的改进实验的评估当中的验证集 Acc 曲线。除 BotNet-L4 外，其余 4 个实验结果，都在 Origin 上下波动。如下图 3.7 右图所示，展示的是在细粒度鸟类数据集 CUB 上对模型架构的改进实验的评估当中的验证集损失函数收敛曲线。BotNet-L4 波动较大，说明其相对较不稳定。未来还有很多可以改进的空间。

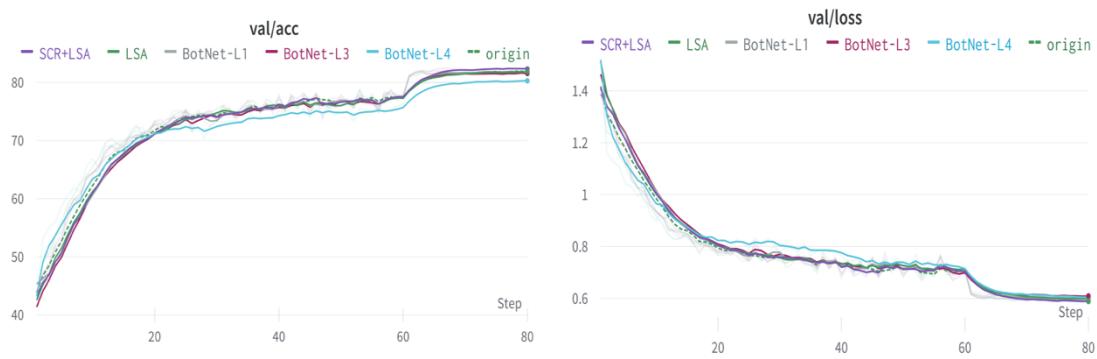


图 3.7 在细粒度鸟类数据集 CUB 上收敛曲线图(5w1s)

3.3 基于 ReNet 的主流数据增强与快照集成引入

训练时，一个算法模型的性能表现和泛化能力直接由和样本的质量和数据的规模决定，所以说深度学习的发展离不开大数据的支持。然而无法实现场景的全覆盖是在数据集的获取当中经常遇到的难题，数据增强这时就需要用到。扩充数据样本丰富程度与规模成为数据增强的主要用处之一。例如，不可控导致光照条件的数据样本相对单调，此时就可以在训练模型中加入数据增强使得光照变化方面的多样性增加；或是标注难度较大或者数据的获取难度大时，也可以使用数据增强的思路来做生成充足的训练数据。

样本多样性低是小样本学习的根本问题，可以通过数据增强在数据规模限制的情况下提高样本多样性。数据增强指借助辅助信息与数据，对原数据集进行特征增强或是数据扩充，可以是合成的带标签数据或者是无标签数据。特征增强增加特征多样性是在原特征空间中添加分类特征。

3.3.1 数据增强图像融合算法 MixUp

应用于计算机视觉中的一种数据增强方法 (beyond empirical risk minimization, MixUp)^[30]发表在 2018 的 ICLR 的文章是 MIT 的张弘毅以及 FAIR 合作的结果。为此，提出了一种训练集图像与标签的凸组合方法。实验表明，该方法可以减少误标记的影响，提高对抗样本的鲁棒性，稳定对抗网络的训练。MixUp 是一种用于计算机视觉的混合增

强算法。它可以混合不同类别的图像来扩展训练数据集。虽然深度模型是强大的，但它受到抗样本影响。为此，提出了一种训练集标签和训练集图像的凸组合方法。实验表明，该方法可以减少误标记对模型的效果的影响，提高模型网络的鲁棒性。

MixUp 原理为 $batch_{x1}$ 与 $batch_{x2}$ 分别是两个 batch 样本，他们对应的标签是 $batch_{y1}$ 与 $batch_{y2}$ ，贝塔分布的 α, β 可计算出来混合系数 λ ，由此论文可以得到 MixUp 公式如下所示：

$$\lambda = \text{Beta}(\alpha, \beta) \quad (3.10)$$

$$mixedbatch_y = (1 - \lambda) * batch_{y2} + \lambda * batch_{y1} \quad (3.11)$$

$$mixedbatch_x = (1 - \lambda) * batch_{x2} + \lambda * batch_{x1} \quad (3.12)$$

其中， $mixedbatch_x$ 与 $mixedbatch_y$ 分别是混合后的 batch 样本与对应标签，Beta 指的是 β 分布。

从作者的角度出发同时解释了这种方法合理性。在 CIFAR-10^[31], CIFAR-100^[31] 等数据集上，MixUp 方法都取得了 SOTA 的效果。图 3.8 为 MixUp 简单的猫狗可视化展示图。

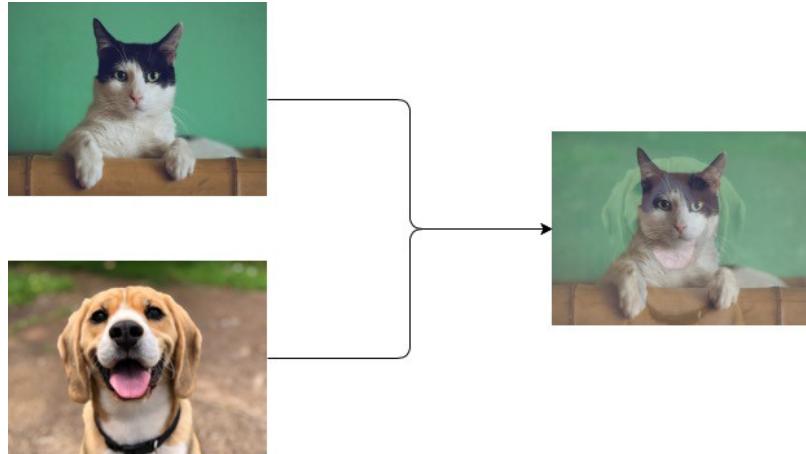


图 3.8 MixUp 简单可视化

3.3.2 数据增强图像拼接算法 CutMix

图像拼接算法 (regularization strategy to train strong classifiers with localizable features, CutMix)^[32], x_A 是一个训练 Batch 当中的训练样本，其对应的样本标签是 y_A 。同理， x_B 是一个训练 Batch 当中的训练样本，其对应的样本标签是 y_B ，CutMix 需要生成的是新的训练样本和对应标签 x_i 和 y_i ，公式如下：

$$x_i = (1 - \mathbf{M}) \odot x_B + \mathbf{M} \odot x_A \quad (3.13)$$

$$y_i = (1 - \lambda)y_B + \lambda y_A \quad (3.14)$$

其中, \odot 是逐像素相乘, λ 是 β 分布: $\lambda \sim \text{Beta}(\alpha, \alpha)$, $\mathbf{M} \in \{0,1\}^{W \times H}$ 是抹掉部分区域和进行填充掩码。

为了对 \mathbf{M} 采样, 要对采样剪裁域的边界 $\mathbf{B} = (r_x, r_y, r_w, r_h)$ 用来对样本 x_A 与样本 x_B 做裁剪区域的标定。在论文中对矩形掩码 \mathbf{M} 采样(样本与长宽大小比例), 剪裁区域的边界框采样公式如下:

$$r_y \sim \text{Unif}(0, H), r_h = H\sqrt{1 - \lambda} \quad (3.15)$$

$$r_x \sim \text{Unif}(0, W), r_w = W\sqrt{1 - \lambda} \quad (3.16)$$

要保证 $\frac{r_w r_h}{WH} = 1 - \lambda$ 为剪裁区域的比例, 确定剪裁区 \mathbf{B} 后, 将 \mathbf{M} 中的裁前其他区域置 1, 区域 \mathbf{B} 置 0, 即可完成掩码的采样。然后将样本 B 中的剪裁区域 \mathbf{B} 进行裁剪然后填充到样本 A 对应位置。如下图 3.9 所示 CutMix 猫狗简单可视化。

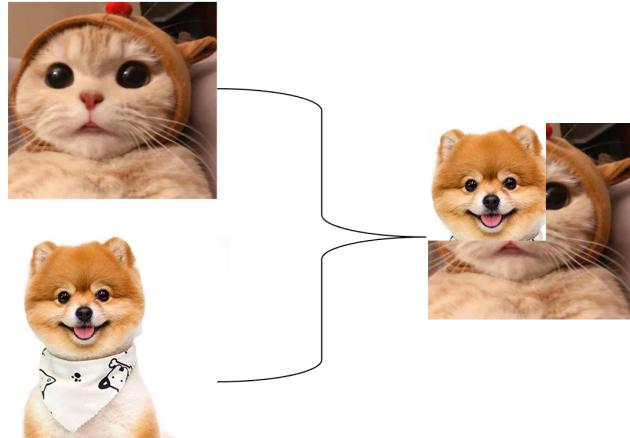


图 3.9 CutMix 简单可视化

3.3.3 数据增强图像擦除算法 Random Erasing

图像擦除算法(Random erasing data augmentation, Random Erasing)^[33], 在选定的图片矩形区域 I_e 中随机值擦除其像素, 随机初始化“擦除矩形”大小为 S_e , $H_e = \sqrt{S_e * r_e}$ 和 $W_e = \sqrt{\frac{S_e}{r_e}}$ 分别为矩形 I_e 的高和宽, 假设 $W * H$ 为训练图像大小, $S = W * H$ 为图像面积, 在图随机取一个点 $P = (x_e, y_e)$, 矩形区域以这个点展开 $I_e = (x_e, y_e, x_e + W_e, y_e + H_e)$,

每个像素在选定的擦除区域内都被分别制定为 [0,255] 中一个随机值。如图 3.10 所示为随机擦除的简单例子。

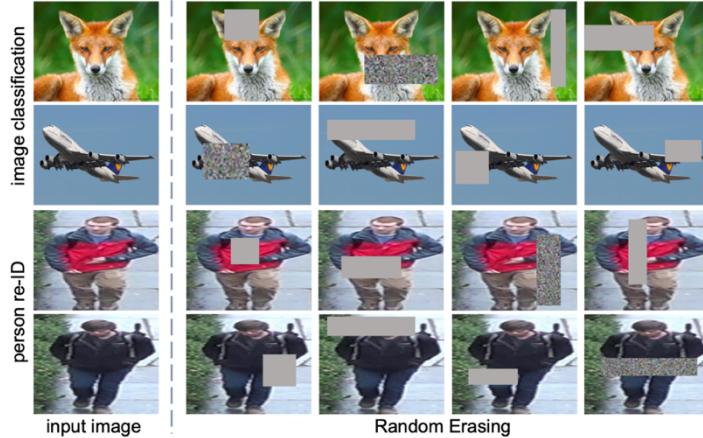


图 3.10 数据增强 Random Erasing

随机擦除有以下优点: (1) 一个轻量级方法, 容易集成 (2) 可对数据扩充进一步提高识别性能。 (3) 提高 CNN 网络鲁棒性等等优点

3.3.4 数据增强图像随机增强算法 TrivialAugment

(TrivialAugment:tuning-free yet state-of-the-art data augmentation, TA)^[34]是一种数据增强方法, 源于 NAS 且效果超越 NAS, 在 NAS 方法中, 虽然数据增强是自动搜索被认为是有效的, 但需要权衡性能和搜索效率是其局限性, 为了解决这个问题, TA 被提出, 相比于之前的数据增强策略, TA 不用将增强策略多种组合, 也不需要特定选择增强策略, 所以说它是一种简单有效的数据增强策略。TA 的最大特点相比于自动 AutoAugment(AA)^[35]乃至 RadomAugment(RA)^[36]是无参数的, 每张图片只使用一次数据增强方式, 搜索成本非常小, 而且在 ImageNet 当中取得了同类型 SOTA 的效果。

工作原理可表示为图 3.11 所示:

Algorithm 1 TrivialAugment Procedure

```

1: procedure TA( $x$ : image)
2:   Sample an augmentation  $a$  from  $\mathcal{A}$ 
3:   Sample a strength  $m$  from  $\{0, \dots, 30\}$ 
4:   Return  $a(x, m)$ 
5: end procedure

```

图 3.11 TA 算法原理

如上图所示，增强函数的集合 A 和输入图像 x 作为输入，A 中随机采样一个增强方法，从 $\{0,1,2,\dots,30\}$ 中均匀采样一个值作为强度 m ，然后对输入图像进行 TA，对于每幅图像，TA 统一地采样一个增强强度，并返回 TA 后的图像，其可视化图可以表示为如下图 3.12 所示。

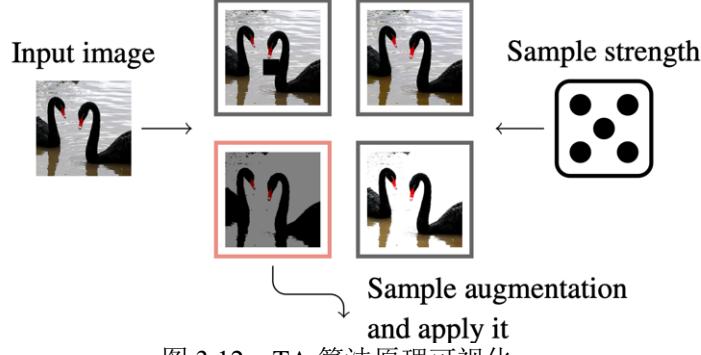


图 3.12 TA 算法原理可视化

3.3.5 数据增强 Horizon-Filp, CenterCrop, RandomCrop

如除以上较为复杂的数据增强外，本论文还用到了相对简单的有效的三种数据增强的方法，他们分别为水平翻转(Horizon-Filp)，中心裁剪(CenterCrop)，任意裁剪(RandomCrop)。以下将对他们进行一一说明。

水平翻转：以给定的概率水平随机翻转给定的图像，虽然水平翻转非常的简单，但是在图像分类以及目标检测等任务当中，有着非常显著的效果，其可以展示为如下图 3.13 所示。

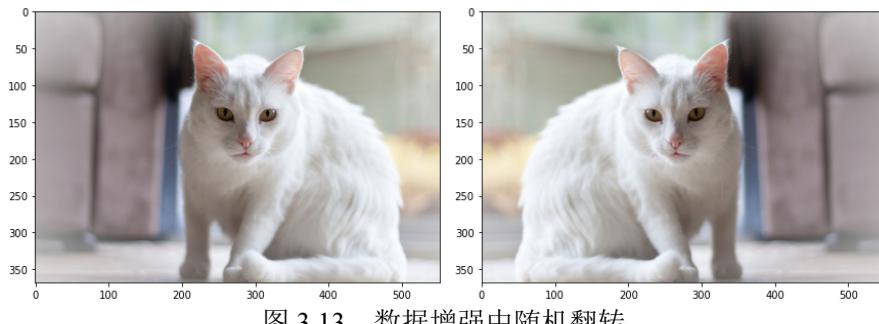


图 3.13 数据增强中随机翻转

中心裁剪：在一个中心的位置裁剪给定的图像，其效果可以展示为如下图 3.14 所示。中心裁剪与任意裁剪类似都是用在小样本学习数据加载部分的数据增强，在本论文

中，不仅将两者用在数据加载部分，还将他们用来增加小样本学习 Shot 的数量，这样可以在图片数量很少的情况下，增加模型精确度。

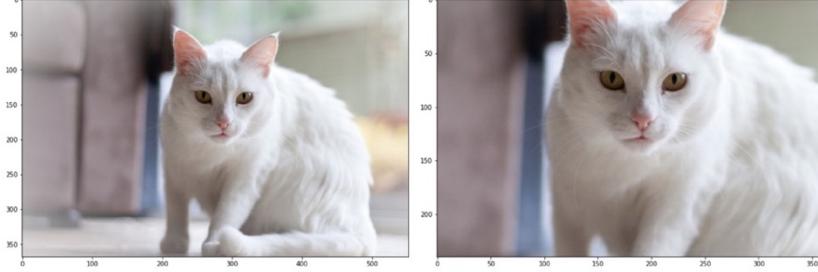


图 3.14 数据增强中 CenterCrop

任意裁剪：在一个随机的位置裁剪给定的图像，在小样本学习以及图像分类当中，几乎成为了必备品，其显著的效果让在各种数据增强任务当中脱颖而出，在深度学习的训练时将图片的随机剪裁已经成为很普遍的数据扩充，不但提高了模型精度其可以展示为如下图 3.15 所示。

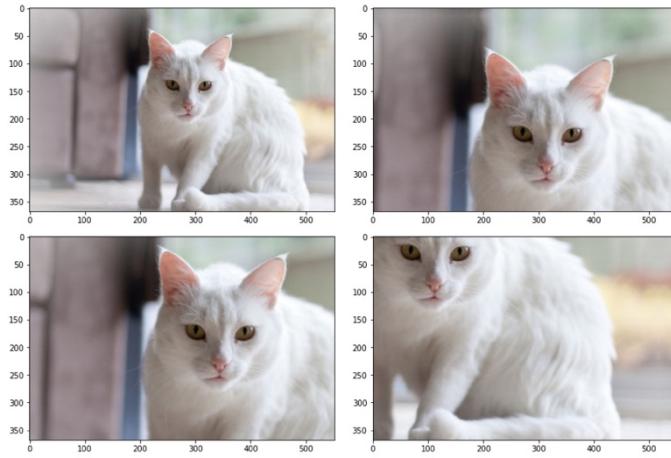


图 3.15 数据增强中 RandomCrop

3.3.6 快照集成算法 Snapshot Ensemble

快照集成算法(Snapshot ensembles: train 1, get m for free, Snapshot Ensembling^[37])。神经网络学习是非凸优化问题，一般只能收敛于局部最优解，且局部最优的个数随着参数的个数呈指数增长。在损失函数值的意义上，几乎所有的局部最优解都接近全局最优解。虽然这些局部最优具有相似的损失值，但它们具有不同的泛化性质。快照集成(Snapshot

Ensembling^[37])针对传统集成方法训练代价过大的缺点,受前面性能曲面研究的启发。快照集成作者认为性能曲面上的一个局部极小点其实就是每个弱模型对应的点,如果有一种方式不用每次从头开始训练就能找到多个局部极小点,即可提升性能。图 3.16 左为使用传统的 SGD 学习策略。图 3.16 右为使用快照集成,模型经历了多次学习速率退火循环逃逸多个局部极小值。小样本学习中,快照集成效果会优于明显优于传统方式训练的单个模型。

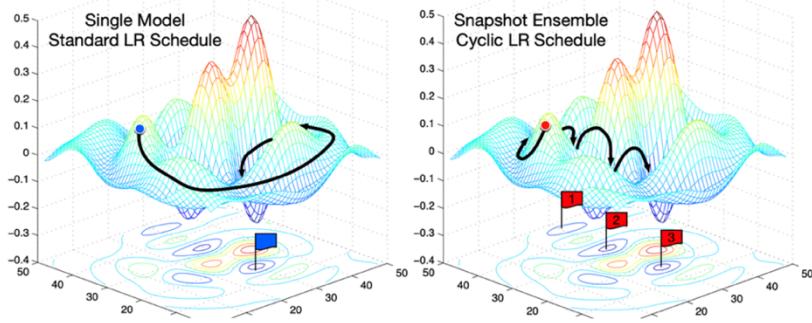


图 3.16 单个模型与快照集成

3.3.7 实验引入过程及其结果

(1) 基本实验设置

小样本学习当中最常用的方法通常采用 Episodic Training,即将训练任务分为 Meta-train && Meta-val && Meta-test。在 Meta-train 当中,是指将数据集分为多个小任务进行训练,从训练集中随机采样 N 个类别,每个类别含有 K 个样本记为 N -way K -shot,每一个小任务中参与训练的样本叫做 Support Set,测试的样本叫做 Query Set,最后将每一个小任务的损失函数和精确度取平均,从而更新模型。在本节当中,采用的基本训练配置与前实验配置相同,具体详情,请看表 3.1 小样本实验模型图如下图 3.17 所示。

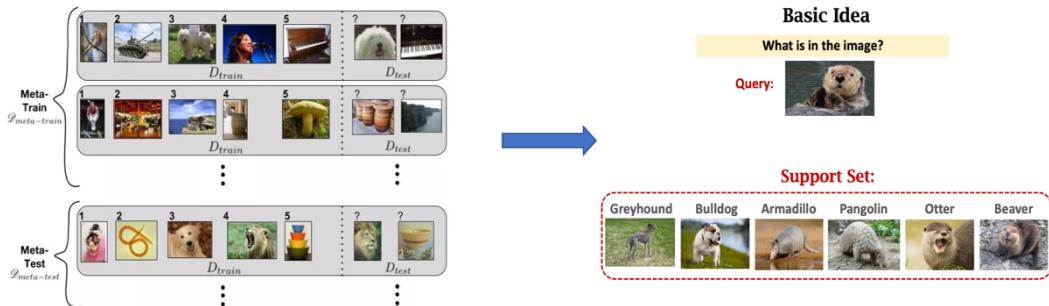


图 3.17 小样本实验模型

在本论文当中，训练过程中采取四种数据增强的方式，分别为：CutMix，MixUp 仅在 Query Set 采用；RandomErasing，TrivialAugment 在 Support Set 和 Query Set 同时采用。在测试过程中，采用 Horizon-Filp，RadomCrop 增加 Shot 的数，增加数量是一种有效并且实用的方法使得在很少样本的情况下，完美实现性能的提升，这种方法在小样本学习当中有了一些基本应用，所以在本论文当中进行实验，以达到复现效果。随后并将训练产生的 4 个数据增强权重采用 Snapshot Ensemble 平均模型。整体流程如下图 3.18 所示，采用数据增强训练产生的权重，在测试时将他们 Logits 进行平均求和，最后再计算交叉熵损失函数，以及求出精确度值 Acc。

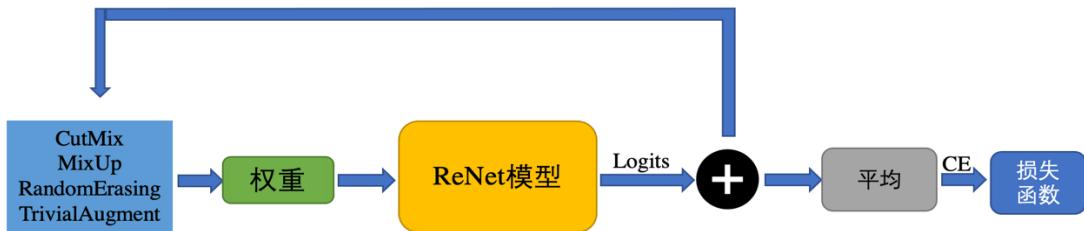


图 3.18 Snapshot Ensemble 模型流程图

(2) 实验结果展示

在此部分一共进行了 2 部分实验，分别在训练时和测试时分别进行数据增强实验以及 Snapshot Ensemble 实验。

训练实验结果：

不同于传统的数据增强，小样本学习可以在 Support Set 和 Query Set 同时或者分别进行数据增强(TASK 表示同时进行)，他们各自效果都不一样。文献[38]中提到 Support Set 上面数据增强的效果不明显甚至有时候反倒效果降低，但只在 Query Set 上面或者 TASK 进行数据增强，效果会得到显著提升。为追求模型效果提升，在 Query Set 和 TASK 数据增强。

训练结果如下表 3.4 所示可知，数据增强在训练过程中，都起到了一定的效果，数据增强通过提升模型的鲁棒性，减少模型的过拟合，从而提升模型效果。在小规模的细粒度数据集 CUB 上面提升尤为明显。其中，MixUp 和 TrivialAugment 增强上面的效果提升显著。MixUp 是公认的对计算机视觉当中有效的方法，TrivialAugment 则是在 AutoAugment 当中的改进版本。TrivialAugment 不需要额外的参数，完全是自适应模型的一种算法。

表 3.4 在 ReNet 上对数据增强的探究实验, QUERY 表示在 Query Set 进行实验, TASK 表示同时在 Query Set 和 Support Set 进行实验

Method	CUB		CIFAR_FS		TRASH	
	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
ReNet	79.20	91.10	74.64	86.67	63.28	78.59
+ CutMix0.5(QUERY)	79.87	91.23	75.48	87.31	63.79	79.27
+ MixUp0.5(QUERY)	80.55	91.78	75.56	86.96	63.40	79.04
+ RandomErasing(TASK)	79.90	91.77	74.67	87.11	63.43	79.47
+ TrivialAugment(TASK)	79.65	91.43	75.25	87.38	64.18	79.50

如下表 3.5 所示, 通过对比发现, 无论是概率 $P=0.5$, 还是概率 $P=1.0$, CutMix 都能在原有基础上得到非常大的提升。对比两个数据集的 1-shot 可知, CutMix1.0 的效果好于 CutMix0.5 的效果, 说明在少量的数据下, CutMix 的效果提升非常明显。对比两个数据集, CutMix 对于 CUB 的提升比 CIFAR_FS 更明显, 说明 CutMix 对于小规模细粒度数据集的效果提升更明显, 说明图像融合数据增强算法在小规模细粒度数据集当中的算法探究值得进一步研究。

表 3.5 在 ReNet 上对数据增强的 CutMix 参数探究实验, QUERY 表示在 Query Set 进行实验, TASK 表示同时在 Query Set 和 Support Set 进行实验

Method	CUB		CIFAR_FS
	1-shot	5-shot	1-shot
ReNet	79.20	91.10	63.28
+ CutMix0.5(QUERY)	79.87	91.23	63.79
+ CutMix1.0(QUERY)	80.45	90.69	63.95

图 3.19 所示展示的是在 ReNet 验证集上分别进行 5-way 1-shot 与 5-way 5-shot 数据增强的探究实验, 其精确度曲线收敛图如下所示。整体曲线较为稳定, 相比于原始模型, 有一定的提升。从当中可以发现, 1-shot 的波动比较于 5-shot 更大, 其原因是由于 5-shot 每一类的数量比起 1-shot 更多, 受到的随机波动更小。从图 3.19 所示可知, 每一种数据集都有各自的波动, 特别是自制数据集, 波动更明显, 这也是显然的, 因为在 CUB 和 CIFAR_FS 都是通过专业小样本学习而制作的数据集, 明显数据噪声会比本论文制作的噪声更少。由于在训练过程中的, Milestone 在 1-shot 是 60 Epoch, 在 5-shot 是 40 Epoch, 从图还可以发现, CUB 这样的小数据集的由于学习率的突然性改变性能提升比较不明显, 反倒是 CIFAR_FS 以及自制数据集提升非常明显。1-shot 在三个数据集当中的提升明显高于 5-shot。

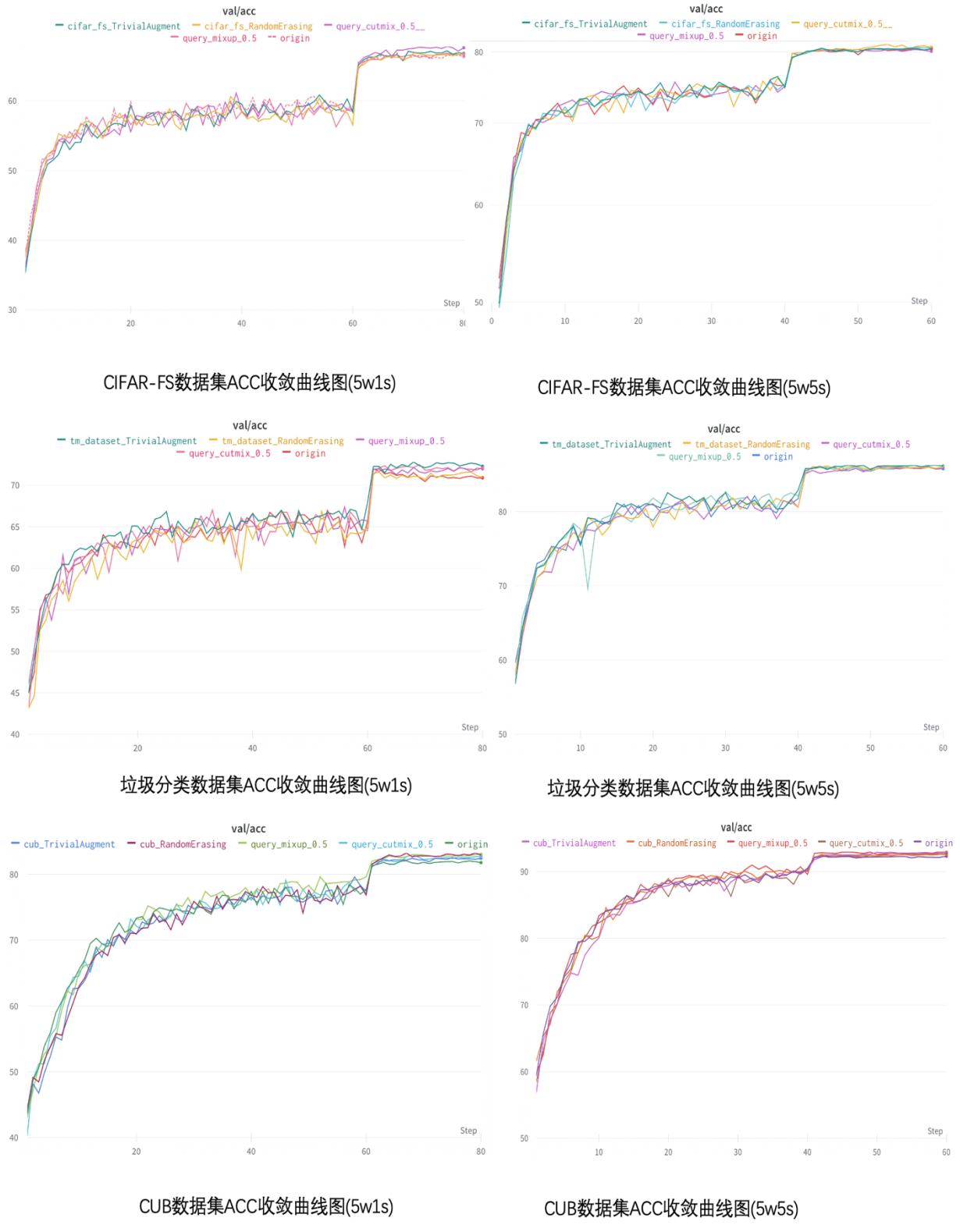


图 3.19 ReNet 数据增强的探究实验曲线图

测试实验结果：

首先对数据增强水平翻转(TASK)进行实验。由表 3.6 可知，水平翻转在 CUB 数据集上，在 1-shot 下有轻微的提升，然而对于 5-shot 性能有部分的下降，在 CIFAR_FS 上也有类似的效果，并不能同时提升 Acc，在垃圾分类数据集 TRASH 当中没有得到任何的改变，根据论文^[38]提到，在小样本数据增强当中，与传统的图像分类数据增强不同，小样本数据增强需要考虑 Support Set 与 Query Set 各种的影响，为此进行了改进。

表 3.6 在 ReNet 上水平翻转探究实验，TASK 表示同时在 Query Set 和 Support Set 进行实验

Method	CUB		CIFAR_FS		TRASH	
	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
ReNet	79.20	91.10	74.64	86.67	63.28	78.59
+ H-FILP(TASK)0.5	79.31	90.89	74.56	86.74	63.28	78.59

为了能在测试过程当中得到一定的提升，根据论文[38]，改进了方法，在 SHOT 数据上面进行水平翻转，其目的是增加 Shot 的数量，将 1-Shot 变成 2-Shot，这样可以在样本有限的基础上，使得精确度增加。由表 3.7 可知，水平翻转增加 Shot 的方式，在三个数据集上面，精确度提升明显。Randomcrop 尽管也有所提升，但是效果不如水平翻转效果好。去掉 ReNet 原有的 CenterCrop 后，有小部分比不上不去掉 CenterCrop 的效果，但是整体提升会比起原来更加明显。为此，为了进一步的效果提升，本文将采用 H-FILP(SHOT)*继续进实验，将采用训练过程中产生的训练权重，结合 H-FILP(SHOT)*进一步改进算法。

表 3.7 在 ReNet 上水平翻转探究实验，*表示去掉 ReNet 原有的 CenterCrop，(xxx)SHOT 表示采用 xxx 增加 Shot 的数量

Method	CUB		CIFAR_FS		TRASH	
	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
ReNet	79.20	91.10	74.64	86.67	63.28	78.59
+ H-FILP(SHOT)	79.58	91.19	74.94	86.81	63.49	78.75
+ H-FILP(SHOT)*	79.63	91.09	74.88	86.84	63.69	78.77
+ RANDOMCROP(SHOT)	78.69	90.68	74.75	86.36	63.37	78.97

由前面表格可知，相比于一开始的原始模型，以及有了非常明显的提升，但是为了能与当今的 SOTA 方法相匹敌，为此进一步进行算法探究。本实验中将训练效果比较好的权重与测试效果比较好的算法结合起来。将采用训练过程中产生的训练权重，结合 H-FILP(SHOT)*进一步改进算法。如下表 3.8 所示，采用数据增强后的权重，效果全部都

得以提升，其中训练过程当中提升明显的 MixUp 以及 TrivialAugment 仍就在测试过程中表现优秀。水平翻转提升 Shot 与训练过程中结合，效果是叠加的，不会因为一种方法导致另一种方法失效。

表 3.8 在 ReNet 上对多种权重数据增强探究实验，H*表示 H-FILP(SHOT)*，*表示去掉 ReNet 原有的 CenterCrop, QUERY 表示在 Query Set 进行实验，TASK 表示同时在 Query Set 和 Support Set 进行实验

Weight	Method	CUB		CIFAR_FS		TRASH	
		1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
ReNet	-	79.20	91.10	74.64	86.67	63.28	78.59
CutMix0.5(QUERY)	+ H*	80.30	91.48	75.74	87.37	63.98	79.12
MixUp0.5(QUERY)	+ H*	80.70	91.94	75.94	87.12	63.52	79.32
RandomErasing(TASK)	+ H*	80.06	91.70	74.82	87.15	63.70	79.56
TrivialAugment(TASK)	+ H*	80.06	91.42	75.23	87.47	64.60	79.63

虽然在 ReNet 上对多种权重数据增强探究实验已经有了进一步的效果提升，但是为了能与当今的 SOTA 方法相匹敌，需要进一步探究实验。通过阅读文献[38]以及[39]可以知道，他们用了 Snapshot Ensemble 平均模型，并取得了不错的效果。为此，我在 ReNet 模型当中，也采用了此方法。实验结果可以表示为如下表 3.9 所示，无论是哪一种模型融合，都使得效果得到了提升，其中 ALL 提升效果最明显。根据 Snapshot Ensemble 论文提出，每一种方式得到的值都是接近全局最优解的，那么通过平均可以使得效果得到巨大提升。

表 3.9 在 ReNet 上对多种权重数据 Snapshot Ensemble 平均模型探究实验，H*表示 H-FILP(SHOT)*，*表示去掉 ReNet 原有的 CenterCrop MAX 表示在表 3.4 当中，选取得分前两名的权重进行 Ensemble；ALL 表示将四种权重全部进行平均

Weight	Method	CUB		CIFAR_FS		TRASH	
		1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
ReNet	-	79.20	91.10	74.64	86.67	63.28	78.59
CutMix0.5(QUERY)	+ H*	81.93	92.72	77.09	88.19	65.50	80.62
+ TrivialAugment(TASK)							
MAX(1st and 2nd)	+ H*	82.01	92.57	76.83	88.29	65.82	80.75
ALL	+ H*	82.65	93.08	77.36	88.43	66.01	81.04

右下表 3.10 可以知道，在经过数据增强，以及快照集成后，效果得到了明显的提升，特别在 CUB 和自制数据集 1-shot 当中提升超过 4%，在 5-shot 当中均超过 2%，实验效

果说明了本论文采用的方法的有效性。也同时证明了小样本学习当中，数据是饥渴的，也严重面临着过拟合问题，通过数据增强的方法可以有效的解决数据饥渴和模型过拟合的问题。对表中数据进行分析可知，CUB 数据集作为一个小数据集，性能提升明显比较大的 CIFAR_FS 和 TRASH 数据集更高，说明了小数据集对于数据增强的需求是迫切的。1-shot 相比于 5-shot 来说，数据增强也是迫切需要的，因为 5-shot 使得模型可以从不同的角度获得更多的特征性能，使得模型收敛效果更好。总体上，可以总结为 5-shot 的提升比 1-shot 更好，大数据集的提升比小数据集更好。

表 3.10 基于 ReNet 的算法探究提升对比 SG 表示最终性能提升的百分点 PI 表示最终性能提升的百分比

Weight	CUB		CIFAR_FS		TRASH	
	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
ReNet	79.20	91.10	74.64	86.67	63.28	78.59
ReNet (OURS)	82.65	93.08	77.36	88.43	66.01	81.04
SG	3.45	1.98	2.72	1.76	2.72	2.45
PI	4.36%	2.17%	3.64%	2.02%	4.30%	3.11%

3.4 本章小节

本章主要介绍了：

- 1) 基于关系嵌入网络的小样本图像分类，首先概述了 ReNet 体系结构，随后阐述了在模型当中采用的自相关与互相关表示法，并说明了他如何关系嵌入的。最后，展示了基本复现过程及其结果评估。
- 2) 主要讲述了基于 ReNet 的 ResNet-12 骨干模型更改探究，首先介绍探究目的与思路；随后介绍了两种引入模型 BotNet 和局部自注意力机制 LSA；其次，阐述了模型更改探究过程及其结果，可视化了 BotNet 与局部自注意力机制 LSA 的结构图，最后进行探究结果展示。
- 3) 介绍了基于 ReNet 的主流数据增强与快照集成引入，分别展开讲解了训练中的数据增强的 MixUp, CutMix, Random Erasing, TrivialAugment, 测试中的水平翻转, 中心裁剪, 任意裁剪。随后介绍快照集成，以及实验引入设计过程，并讲解了基本实验设置以及展示了实验结果。

4 小样本学习 Vision Transformer 及衍生模型引入及改进

4.1 小样本学习 Vision Transformer 及衍生模型引入

4.1.1 探究目的与意义

深度学习当中，已经有非常多的视觉变换器 Vision Transformer (ViT) 模型，特别是在 ViT 横空问世后，越来越多的人开始效仿 ViT 模型，纷纷提出自己的模型。比较著名的有 SwinTransformer, T2T 等等。然而在当前小样本学习研究领域中，目前还没有一篇以视觉变换器为骨架的模型的学术论文，为了让视觉变换器在小样本学习当中能够蓬勃发展，在本章节当中将会对其进行讨论。

在以下章节当中，主要介绍在小样本学习中使用的各个视觉变换器模型。一共有 4 个模型，依次是经典的 an image is worth 16×16 words: transformers for image recognition at scale(ViT)^[40]，类似卷积思想的 tokens-to-token vit: training vision transformers from scratch on imagenet (T2T)^[41]，采用知识蒸馏的 Training data-efficient image transformers & distillation through attention (Distill-T2T)^[42]，Escaping the big data paradigm with compact transformers(CCT)^[43]。每个模型有着各自的特点，其中 ViT 为后续三个视觉变换器的基础模型。

4.1.2 基于纯 Transformer 的 ViT 模型

视觉变换器在自然语言处理领域取得突破性进展，最早提出视觉变换器也是针对自然语言处理领域的。Google 自家的 JFT 数据集预训练后在 ImageNet 上的性能达到了 88.55% 的准确率，说明视觉变换器在计算机视觉领域是可用高效的。

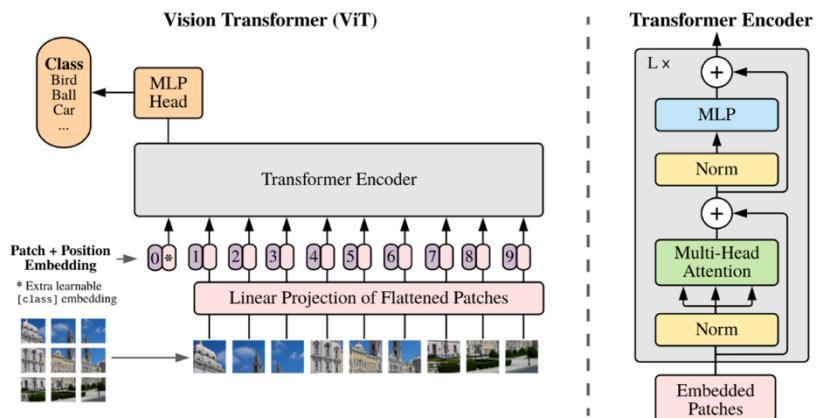


图 4.1 ViT 结构示意图

图 4.1 是关于视觉变换器模型框架结构示意图。简单而言，模型由三个模块组成：视觉变换器编码层，MLP 层，嵌入层。示意图用公式可以表示为：

$$\mathbf{z}_0 = [\mathbf{x}_{\text{class}}; \mathbf{x}_p^1 E; \mathbf{x}_p^2 E; \dots; \mathbf{x}_p^N E] + \mathbf{E}_{\text{pos}}, \mathbf{E} \in \mathbb{R}^{(P^2 \cdot C) \times D}, \mathbf{E}_{\text{pos}} \in \mathbb{R}^{(N+1) \times D} \quad (4.1)$$

$$\mathbf{z}'_\ell = \text{MSA}(\text{LN}(\mathbf{z}_{\ell-1})) + \mathbf{z}_{\ell-1}, \ell = 1 \dots L \quad (4.2)$$

$$\mathbf{z}'_\ell = \text{MSA}(\text{LN}(\mathbf{z}_\ell)) + \mathbf{z}_{\ell-1}, \ell = 1 \dots L \quad (4.3)$$

$$\mathbf{z}_\ell = \text{MLP}(\text{LN}(\mathbf{z}'_\ell)) + \mathbf{z}'_\ell, \ell = 1 \dots L \quad (4.4)$$

$$\mathbf{y} = \text{LN}(\mathbf{z}_L^0) \quad (4.5)$$

4.1.3 基于 Tokens 的 T2T 模型

为了以视觉变换器为骨干模型，寻找高效 ViT 模型，论文借鉴了卷积的一些设计来构建视觉变换器层。不同于 ViT 中使用的 Tokens，Tokens-to-Token(T2T)提出了渐进式 Tokens，将相邻 Tokens 聚合为一个 Tokens(Tokens-to-Token)。Tokens-to-Token 可以对周围 Tokens 的局部结构信息建模，Tokens 的长度迭代地减少。T2T 中，视觉变换器层输出的 Tokens 被图像重构，周围的 Tokens 通过展平分割的 patches 被聚集到一起。因此，来自周围 patches 的局部结构被嵌入要输入到下一视觉变换器层的 Tokens 中。迭代进行 Tokens-to-Token，将局部结构聚合成 Tokens 并聚集过程减少 Tokens 长度。最后，得到的 Tokens 在进入到 ViT 当中。T2T-ViT 由两个主要部分组成(图 4.2)：

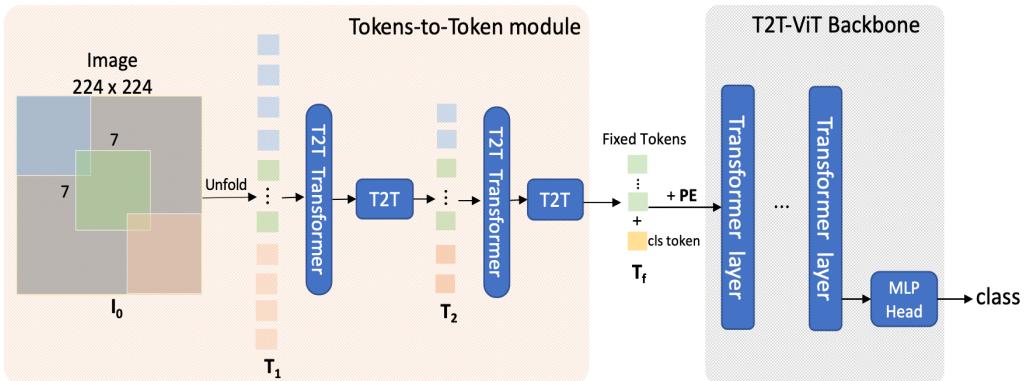


图 4.2 T2T-ViT 的整体网络架构

4.1.4 基于知识蒸馏的 Distill-T2T 模型

(1) 知识蒸馏：

知识蒸馏也叫做 Knowledge Distillation, 知识蒸馏常采用的是老师学生模型, ”知识”的接受者是学生, ”知识”的输出者是老师; 知识蒸馏就好像有一个有几十年经验的知识全面, 水平高的教师在知识蒸馏里面当作一个大模型; 小模型则是经验不丰富, 知识储备能力小的刚入学的学生。简化模型训练: 训练“学生模型”, 简称 NET-S, 是模型结构相对简单参数数量少的单一模型。同样, 对于输入 X , 可以输出 Y , Y 也可以输出 softmax 映射后对应类别的概率值。原始模型训练: “教师模型”, 简称 NET-T, 其特点是模型比较复杂, 也可以由多个单独训练的模型综合而成。对于“教师模型”, 论文没有对参数数量、集成和模型架构做任何限制。唯一的要求是输入 X 可以输出 Y , 其中 Y 由 SoftMax 映射, 输出值对应类别的概率值。有了对知识蒸馏基本的概念, 那么可以深度到 Distill-T2T 模型中来。

(2) 知识蒸馏视觉变换器模型 DeiT^[42]

ViT 使用 Tokens 进行分类, 即一个额外 Patch。Patch 学习自己和其他 Patch 的关系, 通过分类器计算交叉熵损失函数。在 DeiT 中, 对于蒸馏, 添加一个额外的蒸馏 Token 来学习蒸馏 Token 和其他 Token 之间的关系。然后结合教师模型计算 Kullback Leibler 散度损失函数。然后将交叉熵损失函数和 Kullback Leibler 散度损失函数组合成一个新的加权损失函数来指导学生模型训练(教师模型不更新梯度)。如图 4.3 所示, 展示了 DeiT 的基本框图

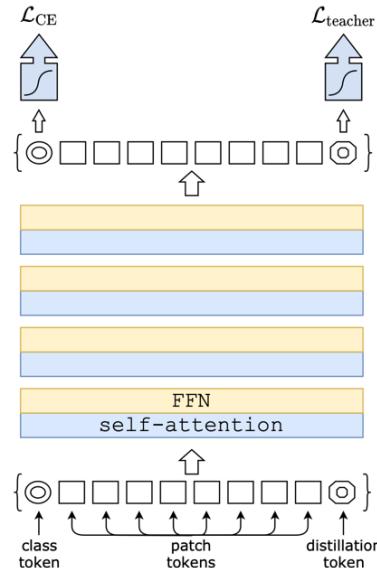


图 4.3 Distillation in Transformer 网络架构

(3) 知识蒸馏视觉变换器模型 Distill-T2T:

Distill-T2T 是在 Deit 的基础上，结合 Tokens-to-Token ViT 一起，使得 Distill-T2T 成为可能。

4.1.5 基于卷积的 CCT 模型

CCT 模型又叫做 Compact Convolutional Transformers，CCT 消除了视觉变换器“需要大量数据”的误解，消除了视觉变换器只处理大型数据集。通过适当的大小的 Tokens，视觉变换器可以在小数据集上与最新的卷积效果匹敌。该模型通过一种卷积的使用，消除了类标记和位置嵌入的需要。CCT 展示了紧凑 Compact Convolutional Transformers 比卷积参数和 MAC 更少，但达到了类似卷积的精度。如图 4.4 所示，展示了 CCT 的基本结构图。

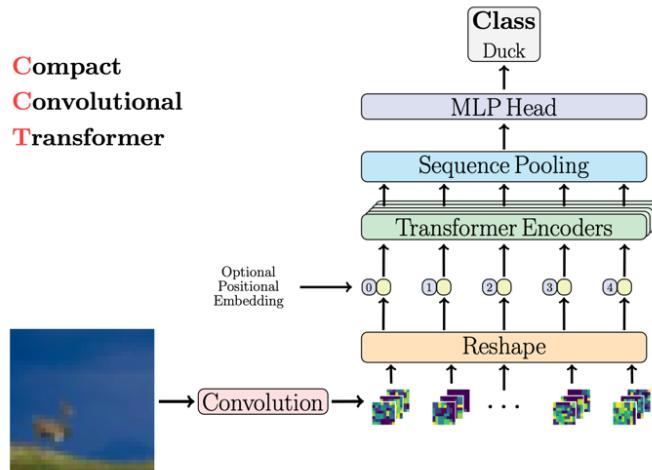


图 4.4 CCT 网络架构

4.1.6 模型复现过程及其结果

对 ViT 的小样本学习复现，采用 Gaussian prototypical networks for few-shot learning on omniglot(ProtoNet)^[20]作为基本模型，而不是采用 ReNet，原因是 ProtoNet 是整个小样本学习当中最原始的，最经典的模型，用其复现更有说服力。研究完成后，再会将 ViT 与 ReNet 模型统调。

(1) 实验设置说明

在本实验中，由于实验设备有限，以及考虑到经济问题，主要以实验室 1080ti 为实验 GPU，实验主要用 miniImageNet 为实验的数据集。在小样本学习当中，经常是多种方法融合到一起，在本章节当中，先用传统的图像类分的方法进行训练，目的是产生一个预训练权重。当得到预训练权重过后，再将得到的预训练权重进行小样本学习训练，

随后将得到的权重进行测试。如图 4.5 所示，是整个实验小样本学习过程的实验流程图。BackBone 表示使用的模型，Metric 则是 BackBone 结果后计算损失函数的方法。在本实验一共有两种，分别是：SQR 表示欧氏距离；COS 表示夹角余弦距离。在后续论文中 ResNet-12+SQR 表示模型通过 ResNet-12 网络后得到的 Logits 用欧氏距离进行计算损失函数后更新模型，ResNet-12+COS 模型通过 ResNet-12 网络后得到的 Logits 用夹角余弦距离进行计算损失函数后更新模型。

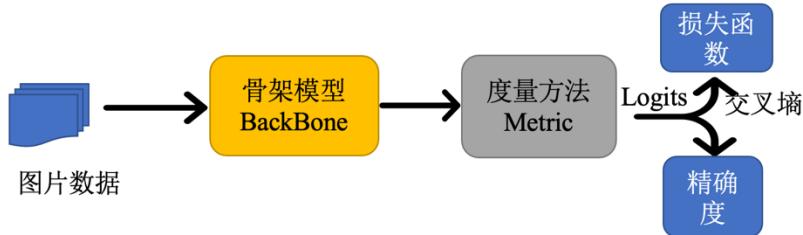


图 4.5 模型探究流程图

复现参数可以列为如下表 4.1 所示，在本章实验当中，均采用此配置方法。

表 4.1 ViT 实验参数说明

CONFIG	Pretrain	1-shot	5-shot
Epoch	100	10	10
Milestone	90	-	-
Optimizer	SGD	SGD	SGD
Learning Rate	0.1	0.001	0.001
Val Episode	-	200	200
Test Episode	-	2000	2000

(2) 实验过程及其结果

如下表 4.2 所示，展示的是以视觉变换器骨干模型在小样本学习上预训练结果展示，从下面的表格可知，去掉卷积网络直接以视觉变换器为骨干模型的效果非常差，并且比起 ConvNet-4 以及 ResNet-12 来说，还有很大的距离。从数据上看，ViT 模型比起 ConvNet-4 在 1-shot 上面差距在 4.7 个百分点，与 5-shot 差距大致在 3 个百分点左右。比起 ResNet-12+SQR 模型来说，在 1-shot 上面差距达到了 9 个百分点，在 5-shot 上面达到了 16 个百分点。这说明了 ViT 并不适合直接使用，在 ViT 原论文当中使用的是大型数据集，而不是 miniImageNet 这样的小数据集。从表格 4.2 当中，可以发现 T2T 与 CCT 以及知识蒸馏的 T2T 可知效果得到的明显的提升，效果基本超过了 ConvNet-4，通过分析，得知

之所以 CCT 与 T2T 的到了提升与采用了类似 CNN 的方法，提升数据的深度有关系，为了进一步验证视觉变换器骨干模型有效性，经过小样本学习训练后再看结果是否得到提升。

表 4.2 视觉变换器骨干模型在小样本学习上预训练结果展示 *表示不是预训练

PreTrain BackBone	Metric Method	MINIIMAGENET	
		1-shot	5-shot
ResNet-12*	SQR	53.81	75.68
ConvNet-4*	SQR	48.7	63.11
ResNet-12	COS	58.91	77.76
ViT	SQR	44	59.85
ViT	COS	46.57	62.74
T2T	COS	52.14	69.53
DISTILL-T2T(ResNet-12)	COS	51.55	66.77
DISTILL-T2T(ResNet-50)	COS	53.08	69.42
CCT	COS	53.04	70.29

如下表 4.3 所示，展示的是以视觉变换器骨干模型在小样本学习上训练结果展示，从下面的表格可知，比起直接用预训练权重测试小样本学习，所有模型效果基本上都在小样本训练后得到不同的提升，一般而言，都是提升 1 个百分点以上。从数据上看，ViT 模型比起 ResNet-12 和 ConvNet-4 在 1-shot 与 5-shot 上面差距变小。但是预训练带来的收益并不是天翻地覆的，如果还想进一步的提升，还需要进一步探索，比如还可以结合一些自监督的方法

表 4.3 视觉变换器骨干模型在小样本学习训练结果展示

BackBone	Metric Method	MINIIMAGENET	
		1-shot	5-shot
ResNet-12	SQR	53.81	75.68
ConvNet-4	SQR	48.7	63.11
ResNet-12	COS	63.17	79.26
ViT	SQR	44.6	62.46
ViT	COS	47.37	62.96
T2T	COS	53.71	70.16
DISTILL-T2T(ResNet-12)	COS	52.06	67.68
DISTILL-T2T(ResNet-50)	COS	53.89	70.27
CCT	COS	53.6	69.93

结合表 4.2 和 4.3 可知，采用预训练权重的方法，带来的收益是显著的，有效的，明显提升有 1 个百分点以上，但是这个收益对于 ResNet-12 和 ViT 的差距，提升却是杯水车薪。采用欧氏距离，夹角余弦距离相比，夹角余弦距离的方法在本论文实验中可以得出结论：欧氏距离来带的效果收益并不如夹角余弦距离的高，原始模型 ViT 的效果比起 ResNet-12 有很大的差距，但是效果与 ConvNet-4 相近。T2T 结合了卷积的思想以及甚至采用了卷积的 CCT 都取得了巨大的提升。分数提升差不多有 6-8 个百分点。知识蒸馏的方法让 ResNet-12 当老师模型，T2T 当成学生模型，反而效果变差，说明小模型无法胜任老师模型的职位，ResNet-50 老师模型却能让效果有大概 0.2 个百分点的提升，尽管这个 0.2 个百分点的分差对于赶上 ResNet-12 杯水车薪，但是也证明了知识蒸馏方法的有效性。

4.2 小样本学习 Vision Transformer 模型改进

4.2.1 探究目的与意义

根据第 4.1 节的探索，发现效果并不是非常的理想，ViT 模型以及衍生模型无法直接在小样本学习这种小数据集上面采用，但是由于许多论文都提出 ViT 模型是能作为主要的骨干模型替代掉卷积网络当中主干模型的。为了让性能进行突破，为此进行了创新，在本节中主要采用第 4.1 节效果最好的 CCT 与 T2T 的思路，结合卷积的思想，在 CNN+ViT 上面进行创新，创新一系列的模型，并分析其效果。根据 4.1 节探索，小样本学习视觉变换器模型目前最好的就是 CNN+ViT 的方法。

在本节中，下面将逐一介绍不同自创改进模型。首先介绍的是 Res9ViT，其次介绍的是 Dilated-Res9ViT，最后介绍 ResT9ViT。ResT9ViT 与 Dilated-Res9ViT 都为 Res9ViT 的衍生模型。图 4.6 所示，展示 Res9ViT 模型与 ResT9ViT 与 Dilated-Res9ViT 的关系。

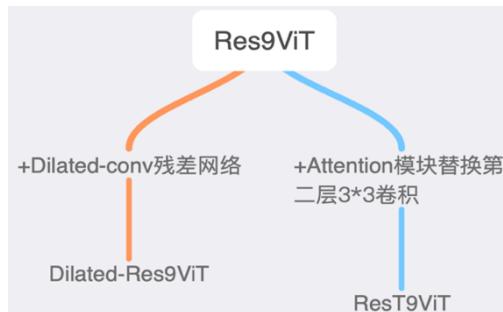


图 4.6 Res9ViT 模型与 ResT9ViT 与 Dilated-Res9ViT 关系图

4.2.2 结合卷积算法 Res9ViT 模型

Res9ViT 模型为自创模型，其核心为结合 ResNet 与 ViT 模型各种的特点。模型如下图 4.7 所示，卷积为 9 层，层数深度分别为 [64, 160, 320]。ViT 部分的维度 Dim 为 320，深度 Depth 为 6，多头注意力机制的头为 8 个，MLP 层的维度为 320×2 即 640。下文当中，会列出每一层的参数，以及有详细的文字说明。

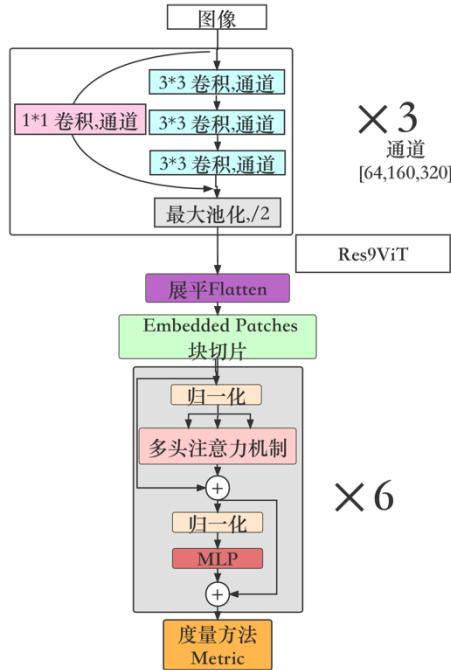


图 4.7 Res9ViT 模型

(1) 卷积部分：

首先， $B \times 3 \times 80 \times 80$ 大小的图片，经过归一化后，先后进入 3 层 ResNet 网络，各层网络的深度逐渐加深，每一层的的卷积核均为 3×3 的卷积。第一层： $B \times 3 \times 80 \times 80$ 经过 3 次卷积后变成 $B \times 64 \times 80 \times 80$ ，出第一层前先和上一层的经过 1×1 卷积后的特征图相加，最后经过最大池化，变成 $B \times 64 \times 40 \times 40$ 。同理，第二层： $B \times 64 \times 40 \times 40$ 经过 3 次卷积后变成 $B \times 160 \times 40 \times 40$ ，出第一层前先和上一层的经过 1×1 卷积后的特征图相加，最后经过最大池化，变成 $B \times 160 \times 20 \times 20$ 。最后，第三层： $B \times 160 \times 20 \times 20$ 经过 3 次卷积后变成 $B \times 320 \times 20 \times 20$ ，出第一层前先和上一层的经过 1×1 卷积后的特征图相加，最后经过最大池化，变成 $B \times 320 \times 10 \times 10$ 。

(2) ViT 部分：

$B \times 320 \times 10 \times 10$ 大小的特征图在进入视觉变换器之前，需要展平成 $B \times 320 \times (10 \times 10)$ 即 $B \times 320 \times 100$ 的特征图，之后先进入 Embedded Patches 目的是为了将特征图切片成 Tokens。在 Res9ViT 当中，Patch Size 为 10，将图片切片成 10×10 的 Tokens，随后将分类 Tokens 与特征图的 Tokens 结合到一起，形成一个 $100+1$ 大小的 Tokens 序列，此时的特征图的大小为 $B \times 101 \times 320$ ，切片完成后。先进行一次层归一化，随后在经过一层多头注意力感知机制，在这一层当中，先将 $B \times 101 \times 320$ 的特征图映射到 \mathbf{Q} 、 \mathbf{K} 、 \mathbf{V} 。由于多头注意力机制的头为 8 个($320/8=40$)， \mathbf{Q} 、 \mathbf{K} 、 \mathbf{V} 的维度为 $B \times 101 \times 40$ ，一共有 8 组 \mathbf{Q} 、 \mathbf{K} 、 \mathbf{V} ，最后再将 8 组 \mathbf{Q} 、 \mathbf{K} 、 \mathbf{V} 的输出拼接起来，输出维度是 $B \times 101 \times 320$ 。随后，加上进入多头注意力机制之前的特征图，经过一个 MLP 层，也即是全连接层将维度放大再缩小回去， $B \times 101 \times 320$ 放大为 $B \times 101 \times 640$ ，再缩小变为 $B \times 101 \times 320$ 。反复进行 6 次相同的操作。

其中，如表 4.4 所示，卷积表示 ResNet 残差网络部分，Transformer 表示 ViT 部分。

表 4.4 Res9ViT 参数表

Res9ViT		
	Layer-1	$\begin{bmatrix} 3 \times 3, 3 \rightarrow 64 \\ 3 \times 3, 64 \rightarrow 64 \\ 3 \times 3, 64 \rightarrow 64 \end{bmatrix} \times 1$
CNN	Layer-2	$B \times 3 \times 80 \times 80 \rightarrow B \times 64 \times 40 \times 40$
		Maxpool
	Layer-3	$\begin{bmatrix} 3 \times 3, 64 \rightarrow 160 \\ 3 \times 3, 160 \rightarrow 160 \\ 3 \times 3, 160 \rightarrow 160 \end{bmatrix} \times 1$
		$B \times 3 \times 40 \times 40 \rightarrow B \times 64 \times 20 \times 20$
		Maxpool
		$\begin{bmatrix} 3 \times 3, 160 \rightarrow 320 \\ 3 \times 3, 320 \rightarrow 320 \\ 3 \times 3, 320 \rightarrow 320 \end{bmatrix} \times 1$
		$B \times 3 \times 20 \times 20 \rightarrow B \times 64 \times 10 \times 10$
		Maxpool
	Flatten	$B \times 320 \times 10 \times 10 \rightarrow B \times 100 \times 320$
	Patch Embedding	$B \times 100 \times 320 \rightarrow B \times 101 \times 320$
Transformer	Layer $\times 6$	Norm $B \times 101 \times 320 \rightarrow$ Multi-head Attention $[B \times 101 \times 40] \times 8 \rightarrow$ $B \times 101 \times 320$
		Add $B \times 101 \times 320$
		Norm $B \times 101 \times 320$
		$B \times 101 \times 320 \rightarrow$
		MLP $B \times 101 \times 640 \rightarrow$ $B \times 101 \times 320$

Metric:

经过 ResNet 层与 ViT 层后，得到的是 $B \times 100 \times 320$ 直接经过全连接层，结合夹角余弦距离求出损失函数更新网络模型。

4.2.3 结合空洞卷积算法 Dilated-Res9ViT 模型

本模型在原来的模型的基础上，增加了空洞卷积的成分。根据调查，原本的 ViT 模型相当于是在一整张图片的基础上直接进行的一个全图卷积，这种方法使得 ViT 模型对于整张图片有了非常大的感受野。但是 Res9ViT 模型，相当于是卷积的基础上进行的注意，感受野受到卷积核的影响，感受到的部分缺乏全局性。为了在 Res9ViT 的基础上，增加感受野，引入了空洞卷积。(又称空洞卷积或膨胀卷积)从字面上看是很容易理解的，它是在标准的卷积图中注入孔洞来增加感受野。与普通卷积相比，膨胀卷积多了一个超参数称为膨胀率，膨胀率是指核区间的数量(如普通卷积为膨胀率 1)。如图 4.8 所示，展示的是卷积核为 3，膨胀率为 1 的空洞卷积示意图。

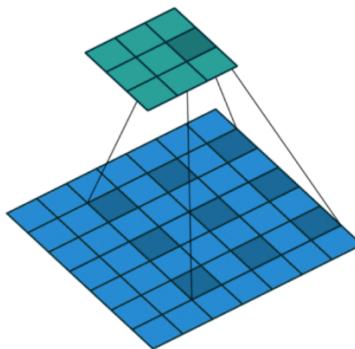


图 4.8 空洞卷积示意图

Dilated-Res9ViT 模型为自创模型，其核心为在 Res9ViT 的基础上增加空洞卷积。模型如下图 4.9 所示，卷积为 9 层，层数深度分别为 [64, 160, 320]。ViT 部分的维度 Dim 为 320，深度 Depth 为 6，多头注意力机制的头为 8 个，MLP 层的维度为 320×2 即 640 层。由于在小样本学习当中，卷积核大小通常为 3×3 ，在层数不深的情况下会导致感受野相对较小，为了解决感受野在层数较浅时特征缺乏全局性，卷积部分在 Res9ViT 的基础上增加了 1×1 空洞卷积的模块，目的就是为了能够增加感受野的大小，空洞卷积的模块卷积核大小设置为 5×5 ，膨胀率设置为 1，经过一次卷积后再通过一次最大池化层将图片从 $B \times 3 \times 80 \times 80$ 变成 $B \times 320 \times 10 \times 10$ 与卷积的出来的模块直接相乘。ViT 部分保持不变，依旧保持与 Res9ViT 相同的参数配置。

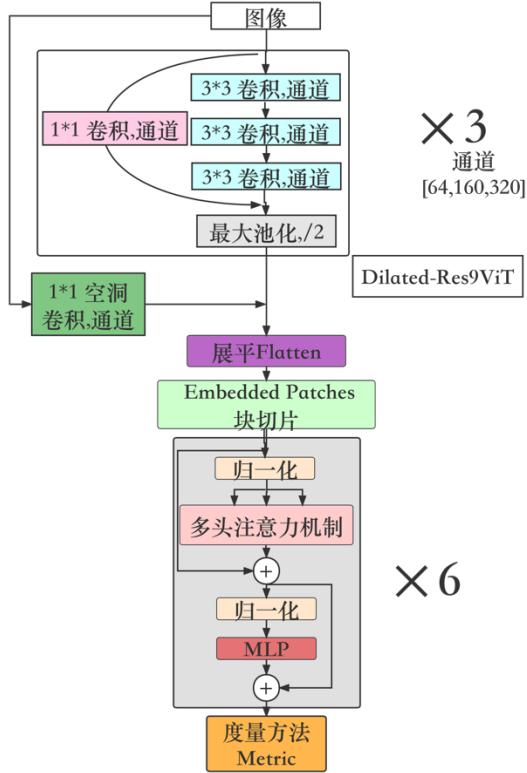


图 4.9 Dilated-Res9ViT 模型

4.2.4 结合 Attention 算法 ResT9ViT 模型

根据前面 Dilated-Res9ViT 和 Res9ViT 的探索，以及第 4.1 节的 ViT 以及衍生模型的复现。从 T2T 当中得到灵感，可在 ResNet 当中添加上 Attention 模块。因为从 ViT 的论文可知，对整张图的注意力，会产生很好的效果。由此，为了使得全局特征更丰富，在 ResNet 的每一层中间将 3×3 卷积替换视觉变换器形成新的模型 ResT9ViT，模型如图 4.10 所示。

ResT9ViT 模型与 Dilated-Res9ViT 的目的相同，都是为了尽可能的增加特征的全局性。Dilated-Res9ViT 是从卷积层面出发，通过空洞卷积的方式增加感受野的大小。不同的是 ResT9ViT 在 Res9ViT 的基础上，用注意力模块替代掉 Res9ViT 的卷积层的每层中间的 3×3 卷积。若替代掉第一层卷积，那么 ResT9ViT 称为 ResT9ViT-L1，同理替代掉第二层卷积，那么 ResT9ViT 称为 ResT9ViT-L2。在论文中，由于 ResT9ViT-L1 要求的 GPU 内存大，所以在实验当中，主要以 ResT9ViT-L2、ResT9ViT-L3 为主。

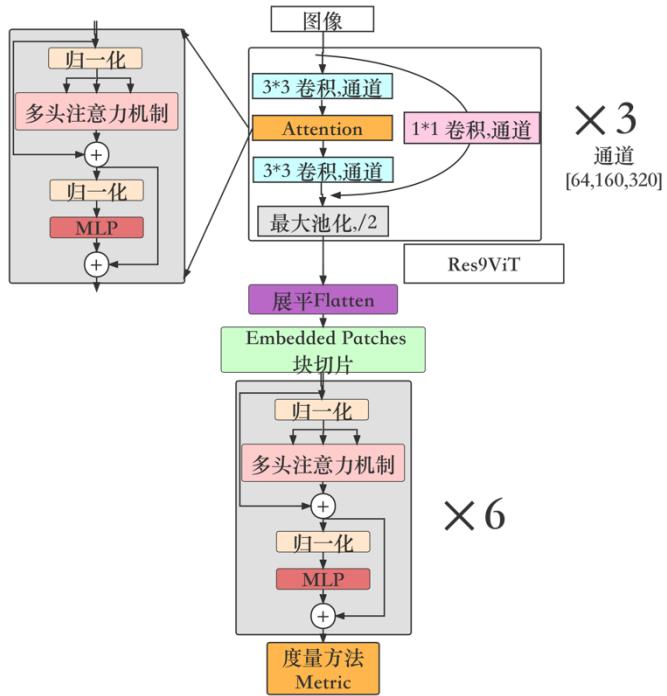


图 4.10 ResT9ViT 模型

4.2.5 模型改进过程及其结果

(1) 实验基本设置

以实验室 1080ti 为实验 GPU，实验主要用 miniImageNet 为实验的数据集。在小样本学习当中，在本章节当中，先用传统的图像类分的方法进行训练，目的是产生一个预训练权重。当得到预训练权重过后，再将得到的预训练权重进行小样本学习训练，随后将得到的权重进行测试。如图 4.11 所示，是整个实验小样本学习过程的实验流程图。BackBone 是本论文使用的模型，Metric 则是 BackBone 结果后计算损失函数的方法。在本实验一共有两种，分别是：SQR 表示欧氏距离；COS 表示夹角余弦距离。

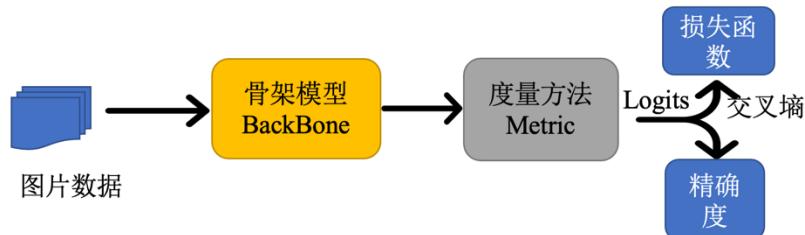


图 4.11 模型探究流程图

(2) 实验过程及其结果

如下表 4.5 所示，展示的是以视觉变换器骨干模型自主改进模型预训练结果展示，从下面的表格可知，结合卷积后，视觉变换器模型的效果得到了显著的提升。

表 4.5 视觉变换器自主改进模型预训练结果展示，*表示不是预训练，384 表示 ViT 层的维度为 384，Mix 表示卷积结果与 ViT 模型结果相加，L2 L23 分别表示在第二层与第二三层使用单头注意力机制模块

PreTrain BackBone	Metric Method	MINIIMAGENET	
		1-shot	5-shot
ResNet-12*	SQR	53.81	75.68
ConvNet-4*	SQR	48.7	63.11
ResNet-12	COS	58.91	77.76
ViT	COS	46.57	62.74
Res9ViT	COS	59.19	76.35
Res9ViT-384	COS	60.18	77.04
Dilated-Res9ViT	COS	59.38	76.34
Res9ViT-Mix	COS	60.79	77.15
ResT9ViT-L2	COS	60.38	76.91
ResT9ViT-L23	COS	59.38	76.01
Res12ViT	COS	59.8	76.78
Res12ViT-Mix	COS	60.97	76.63

根据第 4.1 节可知，所有预训练的结果在最后小样本训练都会得到不同程度的提升。在预训练阶段 Res9ViT 已经超过了 ConvNet-4 的性能。Res9ViT 效果也超过了 ResNet-12+ SQR 的性能效果。对比不同的 ViT 模型，发现采用空洞卷积的 Dilated-Res9ViT 在预训练结果，在 1-shot 比 Res9ViT 的效果更好，提升了大概 0.19 百分点，5-shot 则基本保持一致。说明了，在 1-shot 缺乏数据的情况下，效果会提升得更加明显。相比起 Res9ViT，ResT9ViT-L2 的效果得到了明显的提升，在 1-shot 的情况下，提升了 1.19 百分点，5-shot 的情况下，提升了 0.56 百分点。但是 ResT9ViT-L23 相比起 ResT9ViT-L2，效果反而下降，说明了并不是越多注意力机制越好。

如下表 4.6 所示，展示的是以视觉变换器自主改进模型小样本训练结果展示，从下面的表格可知，比起直接用预训练权重测试小样本学习，基本上所有效果都在小样本训练后得到不同的提升，同第 4.1 节一样，一般而言，都是提升 1 个百分点以上。经过小样本训练，得出效果最好的依旧还是 ResNet-12+COS，尽管采用视觉变换器骨干模型，在大数据集上面的效果得到了验证，但是在小数据集上的效果依旧不如 ResNet。通过下

表发现，ResT9ViT-L2 的 1-shot 的效果与 ResNet-12+COS 的效果差距只有 1.5 个百分点左右，在 Res9ViT-Mix 的 5-shot 的性能与 ResNet-12 的性能差距也在 1.5 个百分点左右，1-shot 与 5-shot 都已经非常接近 ResNet-12 的效果。

表 4.6 视觉变换器自主改进模型小样本训练结果展示，*表示不是预训练，384 表示 ViT 模型的维度为 384，Mix 表示卷积结果与 ViT 模型结果相加，L2 L23 分别表示在第二层与第二三层使用单头

注意力机制模块

BackBone	Metric Method	MINIIMAGENET	
		1-shot	5-shot
ResNet-12*	SQR	53.81	75.68
ConvNet-4*	SQR	48.7	63.11
ResNet-12	COS	63.17	79.26
ViT	COS	47.37	62.96
Res9ViT	COS	60.23	76.52
Res9ViT-384	COS	61.24	77.24
Dilated-Res9ViT	COS	60.6	76.8
Res9ViT-Mix	COS	61.2	77.58
ResT9ViT-L2	COS	61.4	77.22
ResT9ViT-L23	COS	60.45	76.26
Res12ViT	COS	61.21	77.33
Res12ViT-Mix	COS	61.02	77.14

采用预训练权重的方法，带来的收益是显著的，有效的，明显提升有 1 个百分点以上。由于在第 4.1 节里，本论文实验当中 SQR 的性能不如 COS 夹角余弦距离。Res9ViT+COS 性能上虽然已经超过了 ConvNet-4+ SQR 以及 ResNet-12+ SQR，但依旧比不过 ResNet-12+COS。

从 ViT 的 1-shot 的 44.6 个百分点以及 5-shot 的 62.46 个百分点，提升到现在的 ResT9ViT-L2 当中 1-shot 的 61.4 个百分点以及 Res9ViT-Mix 的 77.58 个百分点，分别在 1-shot 和 5-shot 上提升了 37.6% 和 24.2%，横向对比 ViT 来说，已经有了巨大的提升，Res9ViT 比起普通的 ViT 而言，不仅效果提升效果提升了，并且可以训练速度也有很大提升但是模型也有很大的局限性，以下依次列出：

- 模型效果局限性：Res9ViT 衍生出来了许许多多的模型，也有了不错的效果。但是这些的结果仍然需要依靠 CNN 卷积网络的帮助，Res9ViT 并不能摆脱掉 CNN 卷积，且效果比不过 ResNet-12。

- 模型时间局限性：Res9ViT 普遍的训练时间比起 ResNet-12 高 20-30%，有时，若 ViT 的 Tokens 太多，训练时间将会暴涨比起 ResNet-12 高出 20%-30%。
- 模型空间局限性：Res9ViT 虽然比起 ViT 来说，占用内存已经好了很多。但是在一些模型当中，需要占用的内存非常大，ResNet-12 的内存占用大概在 12G 内而 Res9ViT 经常内存占用达到 20G，ResT9ViT-L2 更是达到了 40G。

本论文得出本章结论：ViT 在小样本学习或者小数据集上面要想达到好的效果，还需要继续探究，这是需要花费大量时间和计算成本。

4.3 基于 ReNet 的小样本学习 Vision Transformer 模型

4.3.1 探究目的与意义

经过以上 4.1 和 4.2 节的探究，基本可以确定出 ViT 模型在小样本学习当中的性能。在 4.1 节可以看出尽管 ViT 在整个深度学习中有着巨大的性能提升，但是其并不能直接应用到小样本学习当中来，ViT 无法与 ResNet-12 的性能相匹敌。在 4.2 节的探究中可以知道，结合上卷积网络后，ViT 的性能能够从分的发挥出来，CNN+ViT 的组合使得其性能超越 ConvNet-4，性能接近 ResNet-12。但是但是 Res9ViT 及其衍生模型效果依旧超越不了 ResNet-12。总之，ViT 还有着非常多值得探究的地方，目前在小样本学习中仍然有着全新的模型待等着科研人员的发现。

为了能照应论文题目，在本节当中将会用改善模型应用到 ReNet 当中来。在为了进一步多方位的确定 ViT 模型在 2021 年 SOTA 模型 ReNet 上面的有效性，在本章节当中将替换 ReNet 当中的 ResNet-12 模型，改用 ViT 模型，以及本论文提出的几种衍生模型，由于时间有限，仅对部分模型进行实验。

4.3.2 ReNet 替换模型结构图

如下图 4.12 所示，展示的是 ReNet 模型与 ViT 以及 Res9ViT 等模型的替换效果图。



图 4.12 模型探究流程图

从图中可以知道，Query Image 与 Support Image 进入到 BackBone Model 后，再各自进行一次 SCR 自相关，随后将自相关的结果再进入到 CCA 互相关当中，最终得出的结果分别是特征 Q 与特征 S 。在本实验当中一共在 5 个数据集当中，分别在 1-shot 和 5-shot 的情况下进行实验。一共进行实验的模型分别是 ViT，Res9ViT，Res9ViT-Mix。

4.3.3 结果展示与分析

如下表 4.7 所示，展示的是在 5 个数据集当中，分别采用 ViT，Res9ViT，Res9ViT-Mix，进行实验。

表 4.7 基于 ReNet 的小样本学习视觉变换器模型结果展示，Mix 表示卷积结果与 ViT 模型结果相加

BackBone	CUB		CIFAR_FS		MINIIMAGENET		TRASH	
	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
ResNet	79.20	91.10	74.64	86.67	67.33	82.38	63.28	78.59
ViT	70.70	82.87	66.18	80.94	54.35	70.75	55.51	70.51
Res9ViT	79.88	91.59	73.42	86.25	67.05	82.21	63.24	78.97
Res9ViT-Mix	80.66	92.23	73.95	86.23	67.09	82.57	63.29	78.90

从表 4.7 可以知道，ViT 模型的结果基本实验结果低于 ResNet-12，说明 ViT 并不能直接用于小样本学习当中，从中可以发现 CUB 1-shot 和 5-shot 下降了 9 个百分点左右，CIFAR-FS 1-shot 下降了 8 个百分点，5-shot 下降了 6 个百分点，特别是 miniImageNet 1-shot 下降了 13 个百分点和 5-shot 下降了 12 个百分点。然而，经过改进后的 Res9ViT 效果比较明显，在 ViT 的基础上提升明显，平均提升在 8 个百分点左右。Res9ViT-Mix 与 Res9ViT 相比，并没有很高的提升，甚至在 CIFAR-FS 和 miniImageNet 上面有了一部分的下降。对比 ResNet-12 与 Res9ViT-Mix 而言，在 CUB 1-shot 上面提升了 1.46 个百分点，5-shot 提升了 1.13 个百分点，在垃圾分类数据集 TRASH 当中提升不明显，但是也有了一定的提升。

对比前面所做的工作，在以 ReNet 为基础，在部分数据集当中得到了明显的提升，由此可以得出结论，ViT 模型在小的 CUB 数据集当中效果是明显的，是可行的，但是不能单纯的使用 ViT 模型，ViT 模型在小的数据集当中，还需要有很长的路要走。并且在 GPU 内存的占用方面，依然是很大的问题，相比于 ResNet-12 而言，内存占用有很大的提升，这对于一些小的内存的显卡是一个很大的挑战。倘若想要在小样本学习当中使用 ViT 模型并且取得不错的效果还需要继续努力提升性能。

在下面以 miniImageNet 为例子，展示其数据集当中的性能曲线图，如图 4.13 所示。

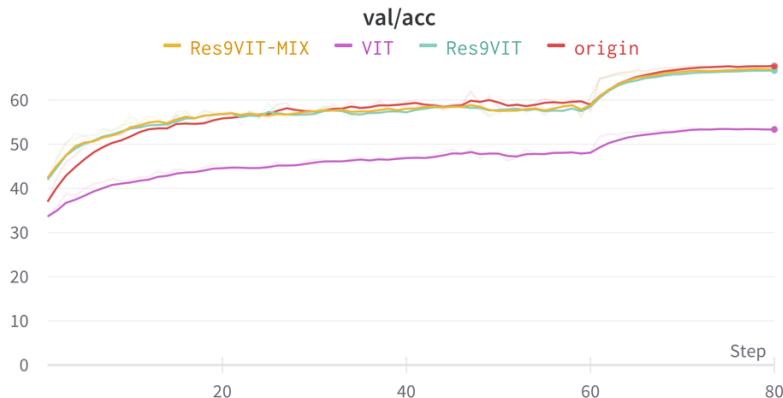


图 4.13 miniImageNet Acc 1-shot 曲线图

图中可知，miniImageNet 的效果在验证集当中，无论是 Res9ViT 还是 Res9ViT-Mix 都不如原始的 ResNet-12 的分数高，从而可以知道，在测试时结果很可能也不如原始的 ResNet-12。ViT 模型的曲线明显低于其他三种，也侧面证明了效果不如 ResNet-12。

在下面以 CUB 为例子，展示其数据集当中的性能曲线图，如图 4.14 所示。

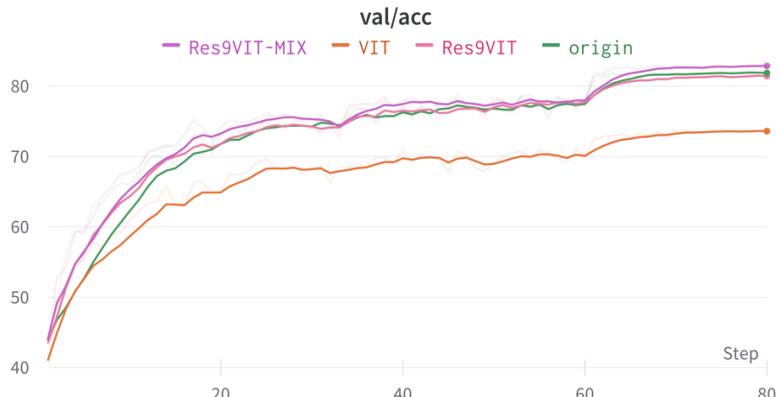


图 4.14 CUB Acc 1-shot 曲线图

从中可以得出，CUB 数据集当中，Res9ViT 的 Acc 曲线与 ResNet-12 曲线而言，在 Epoch 20 之前，明显高于 ResNet-12，然而到了最后，基本上与 ResNet-12 相近，Res9ViT-Mix 的曲线明显高于 ResNet-12 和 Res9ViT。在最终测试结果可以知道，Res9ViT-Mix 的效果好于 Res9ViT 的效果好于 ResNet-12 的效果。

4.4 本章小节

本章主要讲述了

- 1) 小样本学习视觉变换器及衍生模型引入，首先讲述了探究目的与意义，随后讲述了复现模型介绍，一共介绍了四个模型分别是 ViT, T2T, Distill-T2T, CCT。其次，进行了模型复现及其结果记录，在这复现过程中，首先说明实验设置，然后展示实验过程及其结果。
- 2) 主要在第 4.1 章的基础上进行改进，讲述了小样本学习视觉变换器模型改进，首先描述了探究目的与意义，介绍了几种改进模型 Res9ViT, Dilated-Res9ViT, ResT9ViT，最后阐述了模型改进过程及其结果。
- 3) 在 4.3 节当中，主要验证了 ViT 模型以及 ViT 的改进模型在 ReNet 模型的应用，首先介绍探究目的与意义，随后介绍 ReNet 替换模型结构图，最后介绍结果展示与分析。

结 论

首先，本文描述了小样本学习国内外研究现状。其次，为响应国家垃圾分类号召制作了垃圾分类数据集。随后，基于关系嵌入网络的小样本图像分类为基础模型，进行了论文复现。在复现基础上，进行了两种类型的探究：1.ResNet-12 残差网络探究，在 CUB 鸟类数据集中最高提升了 0.17%。2. 数据增强与快照集成探究，在 CUB 和 CIFAR_FS 公开数据集中达到了 SOTA。然后，以视觉变换器为小样本学习骨干模型，对 ViT、T2T、Distill-T2T 以及 CCT 模型进行论文复现。在复现基础上，提出 Res9ViT 以及其衍生模型。最后，将本论文提出的 Res9ViT 模型以及 ViT 模型应用到 ReNet 当中。本文工作主要的目的在于提升小样本学习网络的性能。以下为本文创新与贡献之处总结：

(1) 本文构建含有可回收垃圾，有害垃圾，厨余垃圾，其他垃圾四大类的垃圾分类数据集，其含有 117 个种类，每类图片数量范围在 208~1150 张之间，共计 47951 张样本图片。

(2) 基于关系嵌入网络的小样本图像分类模型，本论文对其进行论文复现，实验结果与原论文基本一致。在复现基础上，进行了两种类型的探究：1.ResNet-12 残差网络探究，在 CUB 鸟类数据集实验设置 5-way 1-shot 下最高提升了 0.17%。2. 数据增强与快照集成探究，引入了 MixUp、CutMix、Radom Erasing、Trivial Augment 等算法。垃圾分类数据集和在 CUB 和 CIFAR_FS 公开数据集上，两种实验设置 5-way 1-shot 和 5-way 5-shot 下分别提升了 4.36%，2.17%，3.64%，2.02%，4.30%，3.11%，实验在 CUB 和 CIFAR_FS 公开数据集达到了 SOTA。

(3) 本文最后以视觉变换器为小样本学习骨干模型，实现视觉变换器及衍生模型引入，几种模型性能基本超过 ConvNet-4 性能，在实验设置 1-shot 与 5-shot 下，模型性能提升了 10% 左右。在复现基础上，提出 Res9ViT 模型以及衍生模型，ResT9ViT-L2 模型以 61.4 个百分点高于 ViT 模型 44.6 个百分点，在实验设置 1-shot 下提升了 37.6%。Res9ViT-Mix 以 77.58 个百分点在实验设置 5-shot 下高于 ViT 模型 62.46 分提升了 24.2%。最后，论文提出的 Res9ViT 模型以及 ViT 模型应用到 ReNet 当中，在 CUB 与垃圾分类数据集当中得到明显的提升。

在本论文当中有许多不足之处还有很多可以改进的空间，在构建数据集当中，可以采用更好的方法比如从各个公开数据集当中爬虫获取，数据清洗时也可以用更好的深度学习模型。在 ResNet-12 探究中，尝试不同深度的 ResNet 模型结合注意力机制是一个值得探究的领域。数据增强算法应用过程中，训练中只尝试了 4 种数据增强方法，测试中只尝试了 3 种数据增强方法。在未来研究中，数据增强算法可以进一步多元化实验。快

照集成的方法效果最为明显，但是同样带来的问题是训练的时间变长，是一个值得继续探究的问题。在视觉变换器骨干模型的探究中，由于视觉变换器占用的 GPU 内存空间经常呈现指数型增长，且视觉变换器常常用于大的数据集当中，小的数据集还有很多值得去探究的地方。

参 考 文 献

- [1] HOWARD J, RUDER S. Universal language model fine-tuning for text classification[C]. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Melbourne, Australia, 2018: 328-339.
- [2] NAKAMURA A, HARADA T. Revisiting fine-tuning for few-shot learning[J]. (2019,10,03) [2022,06,01]. <https://arxiv.org/abs/1910.00216>
- [3] WANG Y X, HEBERT M. Learning from small sample sets by combining unsupervised meta-training with CNNs[J]. Advances in Neural Information Processing Systems, 2016, 29:244-252.
- [4] BONEY R, ILIN A. Semi-supervised few-shot learning with MAML[C]. International Conference on Learning Representations, Vancouver CANADA, 2018:1-13.
- [5] REN M, LIAO R, FETAYA E, et al. Incremental few-shot learning with attention attractor networks[J]. Advances in Neural Information Processing Systems, 2019, 32:5275-5285.
- [6] LIU Y, LEE J, PARK M, et al. Learning to propagate labels: transductive propagation network for few-shot learning[C]. 7th International Conference on Learning Representations, New Orleans, USA, 2019:1-13.
- [7] CAI W, WANG Y, MA J, et al. Can: effective cross features by global attention mechanism and neural network for ad click prediction[J]. Tsinghua Science and Technology, 2021, 27(1): 186-195.
- [8] MEHROTRA A, DUKKIPATI A. Generative adversarial residual pairwise networks for one shot learning[J]. (2017,03,23) [2022,06,01]. <https://arxiv.org/abs/1703.08033>
- [9] WANG Y X, GIRSHICK R, HEBERT M, et al. Low-shot learning from imaginary data[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 2018: 7278-7286.
- [10] DIXIT M, KWITT R, NIETHAMMER M, et al. Aga: attribute-guided augmentation[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 2017: 7455-7463.
- [11] SCHWARTZ E, KARLINSKY L, SHTOK J, et al. Delta-encoder: an effective sample synthesis method for few-shot object recognition[J]. Advances in Neural Information Processing Systems, 2018, 31:2845-2855.
- [12] CHEN Z, FU Y, ZHANG Y, et al. Multi-level semantic feature augmentation for one-shot learning[J]. IEEE Transactions on Image Processing, 2019, 28(9): 4594-4605.
- [13] KOCH G, ZEMEL R, SALAKHUTDINOV R. Siamese neural networks for one-shot image recognition[C]. ICML Deep Learning Workshop, Lille Grande Palais, France, 2015:234-264.
- [14] VINYALS O, BLUNDELL C, LILLICRAP T, et al. Matching networks for one shot learning[J]. Advances in Neural Information Processing Systems, 2016, 29:1-4.
- [15] JIANG L B, ZHOU X L, JIANG F W, et al. One-shot learning based on improved matching network[J]. Systems Engineering and Electronics, 2019, 41(06): 1210-1217.
- [16] SNELL J, SWERSKY K, ZEMEL R. Prototypical networks for few-shot learning[J]. Advances in Neural Information Processing Systems, 2017, 30:4077-4087.

- [17] SANTORO A, BARTUNOV S, BOTVINICK M, et al. Meta-learning with memory-augmented neural networks[C]. International Conference on Machine Learning, New York, New York, USA, 2016: 1842-1850.
- [18] FINN C, ABBEEL P, LEVINE S. Model-agnostic meta-learning for fast adaptation of deep networks[C]. International Conference on Machine Learning, Sydney, Australia, 2017: 1126-1135.
- [19] ZHOU J, CUI G, HU S, et al. Graph neural networks: a review of methods and applications [J]. AI Open, 2020, 1: 57-81.
- [20] FORT S. Gaussian prototypical networks for few-shot learning on omniglot[J]. (2017,08,09) [2022,06,01]. <https://arxiv.org/abs/1708.02735>.
- [21] MALALUR P, JAAKKOLA T. Alignment based matching networks for one-shot classification and open-set recognition[J]. (2019,03,11) [2022,06,01]. <https://arxiv.org/abs/1903.06538>.
- [22] DENG J, DONG W, SOCHER R, et al. Imagenet: a large-scale hierarchical image database[C]. 2009 IEEE Conference on Computer Vision and Pattern Recognition. Miami, FL, USA, 2009: 248-255.
- [23] CUI Y, ZHOU F, LIN Y, et al. Fine-grained categorization and dataset bootstrapping using deep metric learning with humans in the loop[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 2016: 1153-1162.
- [24] REN M, TRIANTAFILLOU E, RAVI S, et al. Meta-learning for semi-supervised few-shot classification[C]. International Conference on Learning Representations, Vancouver CANADA, 2018:235-243.
- [25] KANG D, KWON H, MIN J, et al. Relational embedding for few-shot classification[C]. Proceedings of the IEEE/CVF International Conference on Computer Vision, Monterey, Canada, 2021: 8822-8833.
- [26] SZEGEDY C, LIU W, JIA Y, et al. Going deeper with convolutions[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 2015: 1-9.
- [27] SRINIVAS A, LIN T Y, PARMAR N, et al. Bottleneck transformers for visual recognition[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 2021: 16519-16529.
- [28] VASWANI A, SHAZEE N, PARMAR N, et al. Attention is all you need[J]. Advances in Neural Information Processing Systems, 2017, 30:1-8.
- [29] RAMACHANDRAN P, PARMAR N, VASWANI A, et al. Stand-alone self-attention in vision models[J]. Advances in Neural Information Processing Systems, 2019, 32:68-80.
- [30] Z ZHANG H, CISSE M, DAUPHIN Y N, et al. Mixup: beyond empirical risk minimization[C]. International Conference on Learning Representations, Vancouver, Canada, 2018:1-13.
- [31] KRIZHEVSKY A, HINTON G. Learning multiple layers of features from tiny images[J]. Citeseer, 2009, 29:1-8.
- [32] YUN S, HAN D, OH S J, et al. Cutmix: regularization strategy to train strong classifiers with localizable features[C]. Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 2019: 6023-6032.

- [33] ZHONG Z, ZHENG L, KANG G, et al. Random erasing data augmentation[C]. Proceedings of the AAAI Conference on Artificial Intelligence, Hilton New York Midtown, New York, New York, 2020, 34(07): 13001-13008.
- [34] MÜLLER S G, HUTTER F. Trivialaugment: tuning-free yet state-of-the-art data augmentation[C]. Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 2021: 774-782.
- [35] CUBUK E D, ZOPH B, MANE D, et al. Autoaugment: learning augmentation policies from data[J]. (2019,04,11) [2022,06,01]. <https://arxiv.org/abs/1803.00676>.
- [36] CUBUK E D, ZOPH B, SHLENS J, et al. Randaugment: practical automated data augmentation with a reduced search space[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Snowmass Colorado, USA, 2020: 702-703.
- [37] HUANG G, LI Y, PLEISS G, et al. Snapshot ensembles: train 1, get m for free[J]. (2017,04,01) [2022,06,01]. <https://arxiv.org/abs/1704.00109>.
- [38] NI R, GOLDBLUM M, SHARAF A, et al. Data augmentation for meta-learning[C]. International Conference on Machine Learning, PMLR, Virtual Event, 2021: 8152-8161.
- [39] LIU J, CHAO F, LIN C M. Task augmentation by rotating for meta-learning[J]. (2020,02,08) [2022,06,01]. <https://arxiv.org/abs/2003.00804>.
- [40] DOSOVITSKIY A, BEYER L, KOLESNIKOV A, et al. An image is worth 16x16 words: transformers for image recognition at scale[J]. (2021,06,03) [2022,06,01]. <https://arxiv.org/abs/2010.11929>.
- [41] YUAN L, CHEN Y, WANG T, et al. Tokens-to-token vit: training vision transformers from scratch on imagenet[C]. Proceedings of the IEEE/CVF International Conference on Computer Vision, Monterey, Canada, 2021: 558-567.
- [42] TOUVRON H, CORD M, DOUZE M, et al. Training data-efficient image transformers & distillation through attention[C]. International Conference on Machine Learning, Virtual Event, 2021: 10347-10357.
- [43] HASSANI A, WALTON S, SHAH N, et al. Escaping the big data paradigm with compact transformers[J]. (2021,08,13) [2022,06,01]. <https://arxiv.org/abs/2104.05704>.

附录 A 自建数据库类别详细目录

由于第 2 章，自建数据库中 Meta-train, Meta-val, Meta-test 的每个类别名称、图像数量内容过多，所以在此附录中，将每一类别给展示出来。

Meta-val 一共有 18 类，Meta-test 一共有 24 类具体类别名称、图像数量如下表 1 所示：

表 1 Meta-test 与 Meta-val 类别名称、图像数量统计

Meta-test		Meta-val	
Class Name	Number	Class Name	Number
RECYCLABLES_case	541	KIRCHEN_WASTE_ice_cream	531
RECYCLABLES_Steel_Products	579	RECYCLABLES_Table	359
RECYCLABLES_Milk_machine	290	RECYCLABLES_ruler	513
HAZARDOUS_WASTE_Insecticide	353	RECYCLABLES_Wooden_comb	383
OTHER_WASTE_Feather_duster	296	RECYCLABLES_pillow	298
KIRCHEN_WASTE_Egg_Tart	343	RECYCLABLES_ornaments	529
RECYCLABLES_book	233	RECYCLABLES_Circuit_board	320
KIRCHEN_WASTE_CherryTomatoes	269	RECYCLABLES_shoes	743
KIRCHEN_WASTE_radish	412	RECYCLABLES_coat_hanger	397
RECYCLABLES_Lampshade	259	RECYCLABLES_Measuring_cup	266
RECYCLABLES_skirt	576	RECYCLABLES_jar	857
RECYCLABLES_magnetic_furnace	298	KIRCHEN_WASTE_jackfruit	237
RECYCLABLES_plastic	1021	RECYCLABLES_hot_pack	275
RECYCLABLES_Nylon_rope	325	RECYCLABLES_tyre	336
RECYCLABLES_Glass_products	342	RECYCLABLES_mobile_phone	278
KIRCHEN_WASTE_Fruits	278	RECYCLABLES_Barrel	305
KIRCHEN_WASTE_biscuit	799	RECYCLABLES_Water_cup	628
KIRCHEN_WASTE_Straw_cup	317	KIRCHEN_WASTE_Chickenwings	346
RECYCLABLES_Remote_control	421	-	-
RECYCLABLES_card	283	-	-
RECYCLABLES_vacuum_cup	306	-	-
KIRCHEN_WASTE_peel	216	-	-
RECYCLABLES_Electric_shaver	318	-	-
RECYCLABLES_Draw_bar_box	360	-	-

Meta-train 一共有 75 类，具体类别名称、图像数量如下表 2 所示：

表 2 Meta-train 类别名称、图像数量统计

Class Name	Number	Class Name	Number
KIRCHEN_WASTE_pickled_cabbage	226	RECYCLABLES_network_card	271
RECYCLABLES_Wood_carving	291	RECYCLABLES_Fire_Extinguisher	277
OTHER_WASTE_cotton_swab	255	KIRCHEN_WASTE_shell	323
RECYCLABLES_Electronic_scale	322	RECYCLABLES_lid	313
HAZARDOUS_WASTE_packaging	685	RECYCLABLES_keyboard	329
KIRCHEN_WASTE_melon_seed	224	RECYCLABLES_Gas_stove	306
RECYCLABLES_Metalware	753	RECYCLABLES_cage	325
OTHER_WASTE_towel	300	KIRCHEN_WASTE_French_fries	318
RECYCLABLES_kettle	369	RECYCLABLES_boarding_pass	276
KIRCHEN_WASTE_nut	295	HAZARDOUS_WASTE_Battery	228
KIRCHEN_WASTE.Bean_curd	317	KIRCHEN_WASTE_Pepper	309
OTHER_WASTE_Straw_hat	268	RECYCLABLES_globe	256
RECYCLABLES_Glass_pot	573	RECYCLABLES_Subway_ticket	228
RECYCLABLES_rice_cooker	328	RECYCLABLES_Wrist_watch	295
KIRCHEN_WASTE_Mushroom	306	KIRCHEN_WASTE_egg	599
KIRCHEN_WASTE_Potato_chips	336	RECYCLABLES_stool	674
RECYCLABLES_skin_care_products	371	OTHER_WASTE_pen	508
RECYCLABLES_Alarm	297	KIRCHEN_WASTE_Coffee	248
KIRCHEN_WASTE_bean	283	HAZARDOUS_WASTE_bottle	1404
RECYCLABLES_Socks	326	KIRCHEN_WASTE_Leftovers	331
RECYCLABLES_key	1032	RECYCLABLES_Electric_iron	316
KIRCHEN_WASTE_Roast_Chicken	308	RECYCLABLES_Tag	268
RECYCLABLES_fish_tank	327	RECYCLABLES_Desk_lamp	339
RECYCLABLES_Wooden_spatula	350	RECYCLABLES_slipper	370
RECYCLABLES_bowl	544	OTHER_WASTE_Flyswatter	298
RECYCLABLES_Wire_ball	331	KIRCHEN_WASTE_intestines	1098
RECYCLABLES_Aluminum_products	337	RECYCLABLES_Glassware	1256
RECYCLABLES_table_tennis_bat	299	HAZARDOUS_WASTE_lamp	329
KIRCHEN_WASTE_pitaya	272	RECYCLABLES_envelope	323
KIRCHEN_WASTE_candied_stick	222	RECYCLABLES_tableware	391
RECYCLABLES_ashtray	321	HAZARDOUS_WASTE_Battery	707
RECYCLABLES_blanket	349	RECYCLABLES_Bag	1550
RECYCLABLES_Charging_line	556	RECYCLABLES_Wooden_board	326
RECYCLABLES_Bracelet	369	RECYCLABLES_Foam_board	612
KIRCHEN_WASTE_Walnut	355	RECYCLABLES_electric_fan	335
RECYCLABLES_Calculator	274	RECYCLABLES_Thermos	268
RECYCLABLES_solar_heater	208	KIRCHEN_WASTE_strawberry	259
KIRCHEN_WASTE_jelly	473	-	-

修改记录

第一次修改记录:

第2章当中，指导教师建议绘制一个表格，把自建数据库中每个类别名称、图像数量等都给出来，同时以表格的方式给出 meta training/ val/ test 的划分。

修改后:

由于每个类别名称多，在正文部分无法充分展开，展示内容过于冗长。所以导师建议在附录A中体现。

第二次修改记录:

导师建议下一步关注写作规范(图、表、参考文献、引用等)。

修改后:

对图、表、参考文献、引用进行多次修订，核对正确性

毕业论文正式检测重复比:

第一次检测重复率:1.9%

第二次检测重复率:1.8%

第三次检测重复率:1.7%

记录人(签字):

杨颖鸣

指导教师(签字):

李晓华

致 谢

有幸能考上大连理工，不仅丰富了我的生活，培养了我的科研能力，科研能力的提升离不开从大一以来帮助过我的几位老师。因为科研能力的提升，使得我有机会能在毕业论文实验上有充分的能力去完成一个又一个的尝试与创新，使得我能够有机会借助毕业论文的契机能够用代码复现自己的思考。

实验以及论文写作期间，离不开李培华老师给予的科研环境，离不开龙飞学长的耐心指点，给足了实验设备与资源，使得我能够安心的完成实验以及论文。我将继续努力，继续提升能力让我能有机会站在更高的舞台上，在研究生期间努力学习，科研。希望有朝一日能够站在世界的高点，为祖国做贡献。