北京科技大学

**硕士学位研究生**

**毕业设计(论文)选题报告**



题 目： 面向任务自适应的模型重参数化研究

学 院： 计算机与通信工程学院

专 业： 计算机技术

姓 名： 李冠辰

学 号： M202120812

指导教师 ： 何杰

2022年 12月 9日

目 录

[1 研究背景及意义 3](#_Toc121840155)

[1.1 研究背景 3](#_Toc121840156)

[1.2 研究意义 4](#_Toc121840157)

[2 文献综述 5](#_Toc121840158)

[2.1 卷积神经网络结构设计 5](#_Toc121840159)

[2.1.1 高性能网络结构设计 5](#_Toc121840160)

[2.1.2 轻量化网络结构设计 6](#_Toc121840161)

[2.2 结构重参数化 8](#_Toc121840162)

[2.2.1 结构重参数化原理 8](#_Toc121840163)

[2.2.2结构重参数化综述 11](#_Toc121840164)

[3 研究内容、预期目标及研究方法 13](#_Toc121840165)

[3.1研究内容 13](#_Toc121840166)

[3.2 预期目标 14](#_Toc121840167)

[3.3 拟采取的研究方法 14](#_Toc121840168)

[4 进度安排 15](#_Toc121840169)

[参考文献 16](#_Toc121840170)

1 研究背景及意义

1.1 研究背景

神经网络[1,2]是人工智能领域最热门的模型工具之一，其诞生可以追溯到20世纪40年代。美国心理学家Mcculloch和数学家Pitts在1943年提出了M-P模型，将神经元作为算法的功能逻辑装置，开创了神经网络模型的理论研究[3]。但受限于当时的计算水平，神经网络的发展并未收到广泛的关注。随着硬件计算水平的不断提升以及反向传播算法的诞生[4]，神经网络在2010年代终于大放异彩，成为了人工智能领域的重点研究子领域，并在计算机视觉、文本处理、语音识别、医疗卫生、交通安全等实际应用场景中取得了众多卓越的成果[5,6,7,8]。

卷积神经网络[9,10]是神经网络应用于计算机视觉领域重要工具之一，其利用卷积操作来模仿生物的视觉感知机制，有着参数共享、擅长提取局部空间特征等特点。卷积神经网络最早由法国科学家LeCun于1989年提出，其创新性地使用反向传播算法来训练网络，使卷积神经网络能够顺利胜任手写数字识别任务[11]。2012年，AlexNet首次使卷积神经网络成为了ImageNet竞赛的冠军，其使用的批归一化、Dropout、数据增强、ReLU激活等训练技巧或结构一直流行到现在[12]。2014年提出的GoogLeNet将多分支并联的思想用于卷积神经网络的设计，并获得了当年的ImageNet竞赛冠军[13]。2016年提出的ResNet在网络中使用了残差连接来避免梯度消失问题，促进了超深卷积神经网络的研究[14]。纵观卷积神经网络的发展，新兴模型一般有性能高、体量大、多分支等特点。

随着卷积神经网络体量的不断增大，其计算消耗和内存访问也不断增长，网络的推理速度因此受限，以至于很难将网络应用于需要实时推理的场景中，因此以轻量化设计[15,16]、剪枝[17,18]和量化[19,20]为代表的模型压缩方法被广泛研究，但这些方法普遍会牺牲一部分模型性能，使模型在一些同时需求高性能和高速度的场景中依然可用性较低，例如自动驾驶[21]。此外，模型一般会被设计成“迷你版”、“基础版”、“巨大版”等固定的体量大小，即无法个性化地适应用户任务，这也会导致不必要的性能或效率上的损失。综上所述，设计同时具备高性能和高速度，同时能适应用户任务的神经网络模型迫在眉睫。

1.2 研究意义

性能强大的深度神经网络往往需要更大的存储空间、更繁琐的内存访问、更高的能耗以及更强大的计算资源，因此目前神经网络落地于实际应用场景依然面临着重重挑战。

（1）推理速度和精度的权衡。深度神经网络模型的推理速度和精度被认为是难以兼顾的，高性能的模型往往伴随着更高的参数量和更多的计算量，其推理速度因此受限。为了加快模型的推理速度，轻量化设计、网络剪枝和网络量化等模型压缩方法被广泛研究，然而这些方法都会不可避免地牺牲一部分网络的性能。快速推理和高精度难以兼顾的现象阻碍了深度神经网络在同时需求高性能和高效率的场景中的应用，以自动驾驶为例，不及时的决策或不够准确的识别都可能造成灾难性的后果。深度神经网络模型想要在此类应用场景中落地，则必须要解决推理速度和精度的权衡问题。

（2）模型体量。一般认为，神经网络性能的高低和网络参数量呈正相关关系，性能强大的模型一般拥有更大的参数量，这些参数可能在磁盘上占据百兆甚至千兆的空间。一方面，大量的参数对于嵌入式和移动设备来说是一个巨大的资源负担，模型在这类设备上的可用性因此受限。另一方面，面向不同的识别任务，同一模型的参数量可能呈现出不同程度的不足或冗余，以参数冗余为例，面对图像清晰、数据量有限、类别较少的识别任务，即便采用ResNet系列的小型模型ResNet-18，也难免产生“大材小用”的情况，造成大量的参数冗余。

结构重参数化[22]可以将一个神经网络模型解耦为训练时结构和推理时结构，这两个结构在推理时是线性等价的。训练时结构一般被设计得很复杂，以获得更高的性能。训练完成后，模型被重新参数化为一个相对简单的推理时结构，以避免增加其推理负担。本课题将以结构重参数化为主要方法，通过在训练阶段复杂化网络结构来获得更高精度，并在推理时将网络重参数化为参数量更少、访存更友好、硬件利用率更高的简单结构来提升推理速度，提升网络的可用性，寻找更优的推理速度和精度的权衡点。此外，本课题将探索对网络的扩张，以构建更能适应特定任务复杂程度且体量适当的网络。

2 文献综述

2.1 卷积神经网络结构设计

卷积神经网络的结构需要进行精心的设计以满足各异的需求，例如，追求更高的性能时，一般会考虑使用大卷积核、残差结构、瓶颈结构等易于提升网络性能的结构；追求更轻量化的网络体量时，一般会考虑使用深度可分离卷积、分组卷积、SE模块等低参数量的结构。本节将对高性能以及轻量化的网络结构设计进行分别介绍。

2.1.1 高性能网络结构设计

通过对卷积神经网络结构进行精心设计，可以训练并得到性能卓越的识别模型。2016年何凯明团队提出了ResNet，使用残差连接来使网络自主控制有效深度，有效避免训练超深卷积神经网络时伴随的梯度消失问题，促进了超深卷积神经网络的研究与发展。ResNet诞生之后，大批高性能卷积神经网络百花齐放，2017年，SENet[23]被提出，其使用一个轻量的瓶颈式的编码解码器结构来对通道间关系建模，巧妙地给网络注入了注意力机制，以微小的速度为代价实现了巨大的性能提升，赢得了当年的Imagenet竟赛冠军。同一年，DenseNet[24]被提出，相比ResNet通过残差连接将特征仅传递一次，DenseNet在局部不同层之间建立了密集的连接，使用拼接的方式实现特征的复用，这些特点让DenseNet在参数和计算成本更少的情形下实现比ResNet更优的性能。同年，Xception[25]将GoogleLeNet的Inception结构极端化成深度可分离卷积结构，获得了卓越的识别性能，并从一个全新的角度解释了深度可分离卷积。同年，ResNeXt[26]使用深度可分离卷积来代替ResNet中的普通卷积结构，并在不增加参数量的前提下增加网络的宽度，同样实现了超越ResNet的性能。2018年，SENet的注意力思想和ResNet系列模型被充分结合，因此诞生了性能卓越的SE-ResNet以及SE-ResNeXt模型。2020年，网络架构搜索（NAS）开始流行，以RegNet[27]为主的模型因此创造了更强大的性能，它依靠的是从ResNet系列模型的若干变体子结构搜索出一套性能强悍的模型结构的排列组合来形成最终的网络模型。

2021年是高性能卷积神经网络设计趋势的分水岭，因为基于自注意力机制[28]的Vision Transformers（ViTs）[29]在这一年诞生了。该系列中最典型的Swin Transformer[30]更是在各类计算机视觉任务书超过了纯卷积神经网络的性能，ViTs从此改变了高性能卷积神经网络的设计思路。结合卷积神经网络和ViTs的优势是最新的卷积神经网络设计趋势之一。CoAtNet[31]用卷积层取代了ViT的浅层，以提取通用特征[32]，在几乎没有先验知识的Transformer结构中加入了一定程度的平移不变性[33]，并表现出优异的性能。BoTNet[34]使用多头自注意（MHSA）来代替ResNet中的3×3 卷积层，大大改善了其性能。模仿ViTs的结构来设计卷积神经网络也能取得良好的性能。受ViTs的大感受野的启发，RepLKNet[35]探索了使用非常大的核来使卷积神经网络的性能优于ViTs。为了避免提取大核的纹理偏差的困难，使用小卷积核与之并行训练，并在推理时使用结构重参数化将两者合并。ConvNeXt[36]对高性能的卷积神经网络设计范式进行了重新思考，通过采用更多来自ViTs的设计思想，实现了超越Swin Transformer的性能。

纵观高性能卷积神经网络发展的历史，可以将高性能的卷积神经网络设计范式总结如下：使用ViT风格的Stem结构作为网络的浅层；使用更大的卷积核；使用层规范化或批规范化；使用深度可分离卷积层[37]；使用点式卷积层[38]；加宽神经网络[39]；使用瓶颈或倒瓶颈结构；使用快捷的分支；引入注意力机制。

2.1.2 轻量化网络结构设计

为了降低模型的参数量，提高模型在终端板卡、移动端等设备上的可用性，轻量化的卷积神经网络结构设计被广泛研究，该方法的主要思想是设计参数量更少、计算更为高效的网络结构来替代普通卷积神经网络中冗余性高、计算量大的特征提取结构。

2016年，SqueezeNet[40]提出使用fire module结构来构建网络，该结构中的squeeze层用来减少特征图的通道数，expand层用来以拼接的形式扩充特征图，该网络极大地减少了参数量，总参数量只有AlexNet的50分之一，却达到了和AlexNet相同水平的性能。2017年，谷歌提出了MobileNet-V1[41]，该网络在VGG的基础上，创新性地采用了深度可分离卷积和点式卷积结构来担任特征提取结构，并使用ReLU6来负责激活，MobileNet-V1不仅拥有更少的网络参数，还可以可以在低精度计算下具有更强的鲁棒性。次年，谷歌继续提出了MobileNet-V2[42]，其在MobileNet-V1的基础上增加了残差连接和倒瓶颈结构，以更少的参数实现了更高的性能。2019年，谷歌继续对MobileNet进行升级，提出了MobileNet-V3[43]，该网络是通过神经架构搜索得到的，其在MobileNet-V2的基础上，引入了轻量级注意力组件SE结构，使用了效率更高的h-swish激活函数，并扩大了网络的首尾的卷积核大小，实现了进一步的网络瘦身和性能提升。2018年，旷视科技提出了ShuffleNet-V1[44]，该网络采用分组卷积的方式减少参数量，采用通道打乱的方式实现通道间信息交互，相比MobileNet系列节省了点式卷积的步骤，进一步降低了网络的参数量。同年，旷视科技提出了ShuffleNet-V2[45]，该网络采用拼接而不是相加来实现特征复用，并将shuffle操作后移，提升了网络的推理速度；文章作者还认为FLOPs和参数量有时无法很好地衡量模型的推理速度，并总结了设计高速模型的指南，包括不要剧烈变动网络的宽度、分组卷积的存在会提升访存、碎片化的结构会降低网络的并行程度、逐元素操作是非常耗时的。2019年，谷歌提出了EfficientNet-V1[46]，网络重新考虑并定量评估了结构的宽度、深度以及特征图分辨率的大小对识别性能和效率的影响，通过神经架构搜索技术找到了以上三个指标的平衡点，以此构建出的EfficientNet-V1同时满足了参数少和性能高的要求。2021年，谷歌将MobileNet-V3使用的Fused-MBConv结构加入了EfficientNet的搜索空间中，此结构使用了密集卷积，更适合放在网络的浅层来提取普遍特征，以此构建出的EfficientNet-V2[47]展现出了更高的性能。2019年，谷歌提出MixNet[48]，认为既需要大卷积核来捕获高分辨率模式，也需要小卷积核来捕获低分辨率模式，因此该网络采用了混合的多尺度深度可分离卷积来提取特征，实现了更好的模型精度和效率。2020年，华为提出GhostNet[49]，为了减少网络计算量，作者将传统的卷积分成两步进行，首先利用传统的卷积压缩通道的数量，然后通过深度可分离卷积再扩张特征图，最后将两组特征图拼接到一起，得到最终的输出，这样的做法避免了使用点式卷积，大大减少了网络的参数量和计算量。

2.2 结构重参数化

现有结构A对应一组参数X，结构B对应一组参数Y，如果我们能将X等价转换为Y，就能将结构A等价转换为B，通过将参数等价转换来实现结构等价转换的行为就是结构重参数化。针对模型压缩与加速领域，结构重参数化可以被认为是在传统思路（思路1）的基础上，提供了一种对偶的新思路（思路2），如图1所示。

QR 代码

描述已自动生成

图1 结构重参数化为模型压缩提供新思路

结构重参数化作为一种通过将模型解耦为训练时间结构和推理时间结构来提高性能的一种方法，最近受到了广泛的关注。结构重新参数化不仅能够提升特定网络的性能，也能专门用于高效率网络结构的设计。本节将详细介绍结构重参数化的原理和文献综述。

2.2.1 结构重参数化原理

在卷积神经网络的结构中，结构重参数一般用于复杂结构向单一卷积层的转换，包含了许多范式。如图2展示了常用的训练时复杂结构。

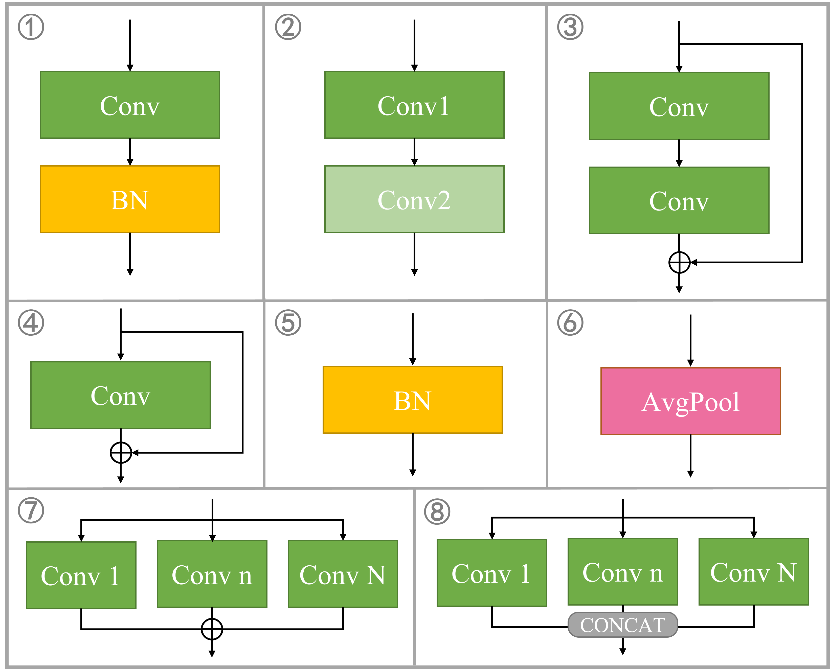


图2 结构重参数化前的复杂训练时结构

1、卷积层和批规范化层的融合。定义输入特征图为，输出特征图为。卷积层的权重为， 卷积层的偏置为（当卷积层在批规范化层前时。一般需要禁用卷积层的偏置，因为同时存在两个偏置是多余的）。将批规范化层的运行时平均、运行时标准差、权重和偏差定义为、、和。结构重新参数化之前的前向传播如公式（2-1）所示。

|  |  |
| --- | --- |
|  | （2-1） |

其中代表卷积运算。根据卷积运算的结合律，公式（2-1）和公式（2-2）等价。

|  |  |
| --- | --- |
|  | （2-2） |

因此，一个卷积层及其后续的规范化层可以合并成一个卷积层，其前向传播可以表示为，其中且。

2、两个相邻的卷积层可以合并。定义输入特征图为，输出特征图为，第一个卷积层的权重为。第一个卷积层的偏差为，第二卷积层的权重为，第二个卷积层的偏差为，根据卷积操作的结合律，结构重新参数化前后的前向传播如公式（2-3）和公式（2-4）所示。

|  |  |
| --- | --- |
|  | （2-3） |
|  | （2-4） |

我们可以使用和直接作为结构重新参数后的卷积层的权重和偏差。然而，合并两个（）的卷积核将产生一个更大的卷积核，这反而会增加推理成本。因此，更有适应价值的是将一个大小为的卷积核和一个大小为的卷积核合并成一个大小为的卷积核。

3、相邻卷积层外的残差连接可以被消除。消除残差连接的关键工具是Dirac初始化，如公式（2-5）所示。

|  |  |
| --- | --- |
|  | （2-5） |

当主干分支存在多个卷积层时，需要增加主分支卷积层的输入或输出维度。增加的输出维度将输入特征图以串联的形式保留在输出特征图中。增加的输入维度用于确保原始特征图可以正确和主分支逐元素相加以得到输出特征图。增加的维度对应的权重被Dirac初始化，如图3所示。

卡通人物

低可信度描述已自动生成

图3 消除相邻卷积层外的残差连接

4、单个卷积层外的残差连接可以被消除。一个残差连接可以被视为是一个Dirac初始化后的卷积层，根据卷积操作的结合律，消除单个卷积层外的残差连接的过程可以被转化为将主干卷积层参数和Dirac初始化的同型卷积层参数相加的过程，如图4所示。

图片包含 徽标

描述已自动生成

图4 消除单个卷积层外的残差连接

5、批规范化层可以转换为卷积层。为了满足转换的需要，可以设想批规范化层之前存在一个卷积层，其权重是被Dirac初始化的，这样，把一个批规范化层转换为卷积层的过程，就转换为了卷积层和批规范化层合并的过程。有趣的是，假想卷积层不受卷积核大小的限制，它还可以是一个分组卷积层。

6、平均池化层可以转化为一个卷积层。应用于个通道、池化核大小为、跨度为的平均池化层转化而成的卷积层同样是个通道、卷积核大小为且跨度为的。其重新参数化过程如图5所示。

QR 代码

低可信度描述已自动生成

图5 平均池化层向卷积层的转化

通过重新参数化得到的卷积层的输入和输出通道数是相同的，其参数初始化如公式（2-6）所示。

|  |  |
| --- | --- |
|  | （2-6） |

7、并联待相加的卷积层可以合并为一个卷积层。定义输入特征图为，输出特征图为。第个卷积层的权重为，第个卷积层的偏差为。对于并联待相加的卷积层，根据卷积操作的结合律，结构重参数化前后的前向传播过程分别如公式（2-7）和公式（2-8）所示。

|  |  |
| --- | --- |
|  | （2-7） |
|  | （2-8） |

因此，并联待相加的个卷积层可以合并为一个卷积层，其前向传播可以表示为，其中，。

8、并联待拼接的卷积层可以合并为一个卷积层。定义输入特征图为，输出特征图为。第个卷积层的权重为，第个卷积层的偏差为。对于并联待拼接的卷积层，根据卷积操作的结合律，其结构重参数化前后的前向传播过程分别如公式（2-9）和公式（2-10）所示。

|  |  |
| --- | --- |
|  | （2-9） |
|  | （2-10） |

因此，并联待拼接的个卷积层可以合并为一个卷积层，其前向传播可以表示为，其中，。

2.2.2结构重参数化综述

2019年，清华大学提出ACNet[50]网络，该网络使用并联待相加的3×3大小、3×1大小和1×3大小的卷积核来替代原始网络中3×3大小的卷积核来进行特征提取，并进行了卷积层和批规范化层的融合，在不降低原始网络的推理效率的情况下，大大提升了其识别的性能。2021年，清华大学继续提出DBB[22]，该方法在AcNet的基础上，采用了结构更为复杂的可重参数化模块来替代原始网络中的3×3卷积层，因此取得了更强大的性能提升。2021年，RepVGG[51]的诞生引发了学界的热议，该网络使用了一个包含有批规范化层的快捷分支的可重新参数化的结构，使得模型可以像ResNet一样被训练，并像VGG一样进行推断。2021年，RMNet[52]创新性地移除了ResNet中的捷径分支，使ResNet成为一个单分支结构，这给网络带来了极大的加速，并且这样重参数化后的模型还对修剪更加友好。2022年，清华大学继续提出RepKLNet[35]，旨在研究巨大的感受野是否拥有大幅提升识别性能的能力，该网络使用最大高达31×31的卷积核来提取特征，为了避免大卷积核带来的细节特征提取不足的问题，还使用了普通小型卷积核与其并联训练，在推理阶段，二者将被重参数化为单分支结构，减轻对推理速度的影响。2022年，浙江大学提出Online-Rep方法[53]，作者指出，虽然结构重参数化能够在不增加推理负担的条件下提升网络的性能，但复杂的网络训练时结构会大大增加训练的成本，在许多情况下是不可接受的，因此作者重新考虑结构重参数化块的设计，将训练时非线性结构替换为线性结构，使在训练阶段就开始结构重参数化成为可能，大大降低了网络的训练成本，还轻微提升了网络的性能。2022年，商汤科技提出了动态结构重参数化DyRep[54]，同样从降低网络的训练成本出发，提出在训练过程中实时对被认为是不重要的可重参数化分支进行修剪，不仅大幅降低了网络的训练成本，还带来了一定程度的性能提升。2022年，华为提出RepGhostNet[55]，认为网络结构中存在的拼接操作虽然不占任何参数量和计算量，但该操作本身较慢的执行速度是不可忽略的，因此通过结构重参数化技术，使逐元素相加操作替代了GhostNet中的拼接操作，同时提升了网络的推理速度和精度。2022年，清华大学提出RepMLP模型[56]，旨在探索同时采用对长远信息建模能力强的全连接层和对局部信息建模能力强的卷积层对模型性能提升的影响，该网络采用卷积层和全连接层并联训练，训练完成后，使用创新的与卷积底层实现方案无关的重参数化方法，将卷积层重参数化为全连接层以和原始的全连接分支合并，该网络在性能方面超越了同等体量的纯全连接网络和纯卷积网络。2022年，腾讯提出OptRep方法[57]，该方法不再区分网络的训练时结构和推理时结构，而是对网络的优化器进行重参数化，以此来模拟结构重参数化的训练过程，而不是真的需要构建复杂的训练时结构重参数化网络，该方法巨大地降低了训练成本，并证明了没有经过精心设计的网络同样能达到和精心设计的网络同等水平的性能。

3 研究内容、预期目标及研究方法

* 1. 研究内容

本课题将探索能对性能提升和效率提升都做出巨大贡献的可重参数化网络结构，探索网络体量如何适应特定任务。尝试构建性能强大、推理速度快、对特定任务适应度高的神经网络解决方案。具体内容包括：

1. 高性能高效率的可重参数化网络结构

结构重参数化在提升特定模型性能方面已经产生了一些优秀成果，但常见的多分支可重参数化结构的各个分支目的和可解释性不明确，并且结构重参数化在提升模型效率方面的潜力还未被完全开发。首先，本课题将探究可重参数化结构中各分支的相似性、重要性等指标，寻找合适的分支组合来构成可重参数化结构，使其各分支分工明确、分支总量合适、整体性能强大。其次，本研究将探索推理效率高的可重参数化结构，将推理时结构设计与轻量化高速推理结构的设计理念相结合，提高网络的推理效率。

1. 网络体量自适应特定任务

目前学界提出的模型一般被设计成“迷你版”、“基础版”、“巨大版”等固定的体量大小，即不能个性化地适应用户的任务，这会导致不必要的性能或速度上的损失。例如，面对图像清晰、数据量有限、类别较少的识别任务，即便采用ResNet系列的小型模型ResNet-18，也难免产生“大材小用”的情况，造成大量的参数冗余。为此，本研究将针对特定任务，探索动态的模型构建方法，使其能够适应特定任务。

结构重参数化一般用于提升固定体量模型的性能，但为了进一步提升模型效率，仍需要通过网络剪枝等手段对模型进行压缩，这可能使得结构重参数化带来的性能提升失去作用。本研究将同时考虑结构重参数化和模型体量对特定任务的适应，从一个极小的模型出发，动态扩张模型的宽度和深度直到达到理想的性能。对宽度的扩张一部分由宽度参数本身承担，另一部分由结构重参数化承担，模型将有潜力以更小的体量，取得更卓越的识别性能。

3.2 预期目标

根据现有的研究，构建同时满足高性能和高效率的可重参数化网络结构，并尝试使网络的体量自适应特定的任务。将本研究的成果在不同的数据集（任务）上进行应用，考察网络的体量、性能和效率等指标，并在相同条件下，与现有方法进行对比，实现更高的任务适应程度、更卓越的性能和更高效率的推理。

3.3 拟采取的研究方法

针对研究内容，拟定研究思路如下：

1. 查阅相关文献，了解结构重参数化目前研究的发展趋势；复现重要的文献方法，理解结构重参数化过程及其原理。
2. 对已有的结构重参数化方法进行分析，总结其优缺点。
3. 综合（1）和（2）的调研结果，实现高性能高效率的可重参数化网络结构的设计：通过推理时间构成分析、内存分析、并行程度分析、硬件利用率分析等方法，评估网络组件的执行效率，构建高效的推理时网络结构；通过对可重参数化结构的多分支进行重要性、相似性、可递归扩张等分析手段，增强可重参数化结构提升识别性能的能力，构建高性能的训练时网络结构。
4. 实现针对特定任务的自适应模型构建与训练方法：从一个极小的模型出发，建模并分析模型深度、宽度等因素对性能和效率的影响，动态扩张模型的宽度和深度直到达到理想的性能。
5. 将（3）和（4）的成果相互结合、相互促进，给出对特定任务适应度高、性能强大、推理速度快的神经网络解决方案，并进行框架化。
6. 整理总结成果，评估本研究提出方法的合理性、可用性，并将其与现有方法对比。
7. 进度安排

依据第3章中所述的研究方法，大致将整个研究分为5个阶段。

第一阶段， 2022.12~2023.02，查阅相关文献，并对查阅到资料文献进行分类总结，理解结构重参数化过程及其原理，确定具体研究内容并细化工作安排。

第二阶段，2023.03-2023.05，深入研究重参数化方法，重点关注对性能和效率提升都起到较大作用的可重参数化结构设计原则及思路，进行实验和探究。

第三阶段，2023.05~2023.06，深入研究模型体量自适应特定任务的相关方法，进行实验和探究。

第四阶段，2023.07~2023.09，将第二阶段和第三阶段的工作结合起来，在特定任务背景下，发挥结构重参数化的优势，训练得到对特定任务适应度高、性能强大、推理速度快的神经网络解决方案。

第五阶段，2023.10~2023.12，总结之前的工作，进行论文的撰写，准备答辩。

|  |  |  |
| --- | --- | --- |
| 阶段 | 阶段名称 | 时间跨度 |
| 1 | 查阅、总结相关文献资料 | 2022.12~2023.2 |
| 2 | 探究结构重参数化方法 | 2023.03~2023.05 |
| 3 | 探究模型的任务自适应方法 | 2023.05~2023.06 |
| 4 | 汇总阶段3和4的方法，设计任务适应度高、性能强大、推理速度快的神经网络解决方案 | 2023.07~2023.09 |
| 5 | 总结、撰写论文 | 2023.10~2023.12 |

参考文献

1. 孙志军, 薛磊, 许阳明, 等. 深度学习研究综述[J]. 计算机应用研究, 2012, 29(8): 2806-2810.
2. LeCun Y, Bengio Y, Hinton G. Deep learning[J]. nature, 2015, 521(7553): 436-444.
3. McCulloch W S, Pitts W. A logical calculus of the ideas immanent in nervous activity[J]. The bulletin of mathematical biophysics, 1943, 5(4): 115-133.
4. Rumelhart D E, Hinton G E, Williams R J. Learning representations by back-propagating errors[J]. nature, 1986, 323(6088): 533-536.
5. Szeliski R. Computer vision: algorithms and applications[M]. Springer Nature, 2022.
6. Eronen A J, Peltonen V T, Tuomi J T, et al. Audio-based context recognition[J]. IEEE Transactions on Audio, Speech, and Language Processing, 2005, 14(1): 321-329.
7. Shen D, Wu G, Suk H I. Deep learning in medical image analysis[J]. Annual review of biomedical engineering, 2017, 19: 221.
8. Lv Y, Duan Y, Kang W, et al. Traffic flow prediction with big data: a deep learning approach[J]. IEEE Transactions on Intelligent Transportation Systems, 2014, 16(2): 865-873.
9. 周飞燕, 金林鹏, 董军. 卷积神经网络研究综述[J]. 计算机学报, 2017, 40(6): 1229-1251.
10. 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.
11. LeCun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.
12. Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[J]. Communications of the ACM, 2012, 60(6): 84-90.
13. 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.
14. 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.
15. 赵永强, 饶元, 董世鹏, 等. 深度学习目标检测方法综述[J]. 2020.
16. Zhou Y, Chen S, Wang Y, et al. Review of research on lightweight convolutional neural networks[C]//2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC). IEEE, 2020: 1713-1720.
17. Liu Z, Sun M, Zhou T, et al. Rethinking the Value of Network Pruning[C]//International Conference on Learning Representations. 2018.
18. Blalock D, Gonzalez Ortiz J J, Frankle J, et al. What is the state of neural network pruning?[J]. Proceedings of machine learning and systems, 2020, 2: 129-146.
19. Yang J, Shen X, Xing J, et al. Quantization networks[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 7308-7316.
20. Zhou A, Yao A, Guo Y, et al. Incremental Network Quantization: Towards Lossless CNNs with Low-precision Weights[C]//International Conference on Learning Representations. 2016.
21. 张新钰, 高洪波, 赵建辉, 等. 基于深度学习的自动驾驶技术综述[J]. 2018.
22. Ding X, Zhang X, Han J, et al. Diverse branch block: Building a convolution as an inception-like unit[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 10886-10895.
23. 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.
24. 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.
25. Chollet F. Xception: Deep learning with depthwise separable convolutions[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 1251-1258.
26. 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.
27. Radosavovic I, Kosaraju R P, Girshick R, et al. Designing network design spaces[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 10428-10436.
28. Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.
29. Dosovitskiy A, Beyer L, Kolesnikov A, et al. An image is worth 16x16 words: Transformers for image recognition at scale[J]. arXiv preprint arXiv:2010.11929, 2020.
30. Liu Z, Lin Y, Cao Y, et al. Swin transformer: Hierarchical vision transformer using shifted windows[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 10012-10022.
31. Dai Z, Liu H, Le Q V, et al. Coatnet: Marrying convolution and attention for all data sizes[J]. Advances in Neural Information Processing Systems, 2021, 34: 3965-3977．
32. Aragon-Calvo M A. Classifying the large-scale structure of the universe with deep neural networks[J]. Monthly Notices of the Royal Astronomical Society, 2019, 484(4): 5771-5784．
33. Kayhan O S, Gemert J C. On translation invariance in cnns: Convolutional layers can exploit absolute spatial location[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 14274-14285．
34. 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. 2021: 16519-16529．
35. Ding X, Zhang X, Han J, et al. Scaling up your kernels to 31x31: Revisiting large kernel design in cnns[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 11963-11975.
36. Liu Z, Mao H, Wu C Y, et al. A convnet for the 2020s[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 11976-11986.
37. Chollet F. Xception: Deep learning with depthwise separable convolutions[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 1251-1258.
38. Hua B S, Tran M K, Yeung S K. Pointwise convolutional neural networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 984-993.
39. Radosavovic I, Kosaraju R P, Girshick R, et al. Designing network design spaces[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 10428-10436.
40. Iandola F N, Han S, Moskewicz M W, et al. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size[C]//International Conference on Learning Representations. 2016．
41. Howard A G, Zhu M, Chen B, et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications[J]. arXiv preprint arXiv:1704.04861, 2017．
42. Sandler M, Howard A, Zhu M, et al. Mobilenetv2: Inverted residuals and linear bottlenecks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 4510-4520．
43. Howard A, Sandler M, Chu G, et al. Searching for mobilenetv3[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2019: 1314-1324.
44. Zhang X, Zhou X, Lin M, et al. Shufflenet: An extremely efficient convolutional neural network for mobile devices[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 6848-6856．
45. Ma N, Zhang X, Zheng H T, et al. Shufflenet v2: Practical guidelines for efficient cnn architecture design[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 116-131．
46. Tan M, Le Q. Efficientnet: Rethinking model scaling for convolutional neural networks[C]//International conference on machine learning. PMLR, 2019: 6105-6114．
47. Tan M, Le Q. Efficientnetv2: Smaller models and faster training[C]//International Conference on Machine Learning. PMLR, 2021: 10096-10106.
48. Tan M, Le Q V. Mixconv: Mixed depthwise convolutional kernels[J]. arXiv preprint arXiv:1907.09595, 2019．
49. Han K, Wang Y, Tian Q, et al. Ghostnet: More features from cheap operations[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 1580-1589．
50. Ding X, Guo Y, Ding G, et al. Acnet: Strengthening the kernel skeletons for powerful cnn via asymmetric convolution blocks[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2019: 1911-1920.
51. Ding X, Zhang X, Ma N, et al. Repvgg: Making vgg-style convnets great again[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 13733-13742.
52. Meng F, Cheng H, Zhuang J, et al. RMNet: Equivalently Removing Residual Connection from Networks[J]. arXiv preprint arXiv:2111.00687, 2021.
53. Hu M, Feng J, Hua J, et al. Online Convolutional Re-parameterization[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 568-577.
54. Huang T, You S, Zhang B, et al. DyRep: Bootstrapping Training with Dynamic Re-parameterization[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 588-597.
55. Chen C, Guo Z, Zeng H, et al. RepGhost: A Hardware-Efficient Ghost Module via Re-parameterization[J]. arXiv preprint arXiv:2211.06088, 2022.
56. Ding X, Xia C, Zhang X, et al. Repmlp: Re-parameterizing convolutions into fully-connected layers for image recognition[J]. arXiv preprint arXiv:2105.01883, 2021.
57. Ding X, Chen H, Zhang X, et al. Re-parameterizing Your Optimizers rather than Architectures[J]. arXiv preprint arXiv:2205.15242, 2022.