基于数论模运算的深度学习效率提升

张弛

2025 年 6 月 27 日

摘要

近年来，随着图形处理器性能的飞速提升，深度神经网络在多个领域取得了显著的进展，尤其是在图像识别、语音识别和自然语言处理等方面。尽管深度神经网络在多个任务中表现出强大能力，但主流模型仍存在一些显著缺点。因此，如何在确保计算精度的前提下，降低训练时间、提高训练效率，已成为深度学习研究中的一个重要方向。数论理论因其在计算效率上的潜力，逐渐引起了学术界的广泛关注。本文主要探讨了数论理论在提升深度神经网络效率方面的潜力，提出了两种可能的实现方式。首先，研究了数论中的模运算是否能够通过加速卷积神经网络的计算过程，降低时间复杂度。在此过程中，数论的模运算可以减少卷积操运算中大量复杂的计算，从而提升计算效率并缩短训练时间；其次，本文还探讨了数论模运算是否能在数据增强方面发挥作用，通过有效的数据增强技术提升模型的泛化能力，进而提高训练效率。通过这些研究，旨在为深度神经网络的优化提供新的视角和有效的技术手段，以期在保证精度的同时显著提升深度学习模型的效率。

关键词： 数论、模运算、深度学习优化、卷积运算、数据增强、训练加速

Abstract

In recent years, with the rapid improvement of graphics processor performance, deep neural networks have made significant progress in multiple fields, especially in image recognition, speech recognition, and natural language processing. However, despite the powerful capabilities demonstrated by deep neural networks in multiple tasks, mainstream deep learning models still have some significant drawbacks. Therefore, how to reduce training time and improve training efficiency while ensuring computational accuracy has become an important direction in deep learning research. Number theory has gradually attracted widespread attention in the academic community due to its potential for computational efficiency. This article mainly explores the potential of number theory in improving the efficiency of deep neural networks and proposes two possible implementation methods. Firstly, we investigated whether modular operations in number theory can reduce time complexity by accelerating the computation process of convolutional neural networks. In this process, modular operations in number theory can reduce the large amount of complex calculations in convolution operations, thereby improving computational efficiency and shortening training time; Secondly, this article also explores whether number theoretic modular operations can play a role in data augmentation, by using effective data augmentation techniques to improve the model's generalization ability and thus enhance training efficiency. Through these studies, the aim is to provide new perspectives and effective technical means for optimizing deep neural networks, in order to significantly improve the efficiency of deep learning models while ensuring accuracy.

Keywords: Number theory, modular operations, deep learning optimization, convolution operations, data augmentation, training acceleration

1 引言

1.1 研究背景

随着深度学习技术的快速发展，神经网络模型在计算机视觉、自然语言处理和语音识别等领域取得了显著进展。然而，随着模型的规模日益增大和数据集的复杂度提高，深度学习面临的计算瓶颈愈加显著，尤其在计算效率、训练时间和模型精度等方面。

卷积神经网络在处理大量数据时，计算量呈现出指数级增长，尤其在卷积操作中，随着网络层数的增加，卷积层的计算复杂度也急剧增加。尽管现代 GPU 加速计算极大提升了训练速度，但随着模型架构和数据集规模的增加，训练时间依然成为一个重要瓶颈。此外，模型中大量的参数和计算需求导致了巨大的存储和计算资源消耗，尤其是在内存受限的环境下，如何优化这些模型成为了深度学习研究中的一个重要问题。其次，深度学习的训练过程通常依赖于大量的标注数据。为了提高模型的泛化能力，数据增强技术被广泛应用于训练过程中。传统的数据增强方法包括旋转、裁剪、翻转等变换，然而，这些变换虽然能够增加训练样本的多样性，但由于其随机性较强，可能导致增强后的数据分布与实际数据分布存在差异，进而影响模型的泛化能力。

近年来，数论模运算作为一种优化计算效率的有效工具，逐渐引起了学术界的关注。数论模运算方法通过对卷积神经网络中卷积结果进行模运算，能够有效减小计算量并提高计算速度。这种方法能够在不显著降低模型精度的前提下，大幅度加速深度神经网络的训练过程。此外，数论模运算的引入还为数据增强提供了新的思路。在传统的数据增强方法中，变换的过程通常是随机的，这可能导致训练数据的不稳定性。而数论方法通过利用模运算生成具有规律性的旋转角度、裁剪比例等变换参数，从而使数据增强更加稳定和高效。

因此，本研究旨在探索数论模运算在卷积运算和数据增强中的应用，希望通过这两种优化，能够显著减少训练时间，降低计算资源消耗，并提升模型在实际应用中的表现。

1.2 研究目的

本研究的主要目标是通过数论模运算优化深度学习模型的计算效率，并探索其在提升训练效率、加速卷积运算和优化数据增强中的应用潜力。具体来说，本文旨在研究以下几个方面：

1. 优化卷积运算：传统卷积运算的计算复杂度较高，尤其在卷积层数较多的深度神经网络中，卷积操作往往成为性能瓶颈。本研究提出将数论模运算应用于卷积神经网络（CNN）的卷积层中，通过对卷积输出结果进行模运算，期望在不显著损失精度的前提下，显著加速卷积运算，并降低模型的计算复杂度。

2. 加速训练过程：传统卷积神经网络的训练时间往往较长，特别是在处理大规模数据集时，本研究将探讨如何利用数论模运算降低计算量，提升训练效率，减少训练时间。通过在计算过程中应用数论模运算，本研究希望能在保证模型性能的前提下，优化计算过程，从而实现加速训练的目标。

3. 优化数据增强：数据增强是深度学习中的一种常见技术，能够有效提高模型的泛化能力。然而，传统的数据增强方法通常依赖于随机变换，容易导致增强结果的不确定性。本研究通过数论方法生成更加规律的变换方式，探索数论模运算在数据增强中的应用，期望通过数论模运算实现更加高效、稳定的增强过程。

本研究通过在卷积神经网络中引入数论模运算，提出了一种新的优化方案，旨在减少计算量、提高训练效率并降低存储需求，从而在深度学习的计算瓶颈问题上提供一种可能的解决路径。

1.3 论文结构

本文结构如下：第二章简要介绍了数论的基本概念和数论模运算的相关理论，并探讨了数论模运算在深度学习中的应用基础。第三章详细描述了基于数论模运算的优化卷积运算的方法，包括具体实现细节与实验分析；第四章探讨基于数论模运算的数据增强方法，并设计实验验证其效果与分析结果；第五章给出论文的结论，强调数论在深度学习模型优化与泛化中的关键作用。本研究通过两个实验，验证上述创新点在提升深度学习模型性能方面的有效性，为进一步探索数论在深度学习中的应用提供了参考。在本研究中，所有代码均已开源并托管于 GitHub：https://github.com/freastzc/shulun-CNN.git。

2 数论在深度学习领域中的理论基础

2.1 数论概述

数论，作为数学的一个分支，专注于研究整数及其相互关系的性质。它的研究内容涉及许多重要的数学概念，包括素数、同余、模运算、算术函数等，这些概念在许多应用领域，如密码学、计算机科学和图像处理等，都有着广泛的应用。[1]

1. 模运算 [3]

   模运算是数论中的核心概念之一，它用于计算一个整数除以另一个整数的余数，通常表示为 a mod m, 其中 a 是被除数，m 是除数。例如 17 mod 5 = 2，因为 17 除以 5 的余数为 2。模运算具有以下性质：

   (a) 交换律：$(a+b) \mod m = [(a \mod m) + (b \mod m)] \mod m$

   (b) 结合律：$(a \times b) \bmod m = [(a \bmod m) \times (b \bmod m)] \bmod m$

   (c) 分配律：$(a + b \times c) \bmod m = [(a \bmod m) + (b \bmod m) \times (c \bmod m)] \bmod m$

   在计算中，模运算常用于简化数值范围、减少计算量以及加速运算，尤其适合大数运算。

2. 同余 [5]

   两个整数 a 和 b 对模 m 同余，表示 $a \equiv b \mod m$，即 a 和 b 除以 m 后的余数相等。同余在数论中具有重要的作用，它与加法、乘法等操作密切相关。例如，如果 $a \equiv b \mod m$ 和 $c \equiv d \mod m$，则我们可以得到：

   - $a + c \equiv b + d \mod m$
   - $a \times c \equiv b \times d \mod m$

   同余关系广泛应用于加密算法、哈希函数以及各种计算中，特别是在需要大量重复计算时，利用同余关系可以有效减少计算开销。

3. 质数 [4]

   质数是只能被 1 和它自身整除的自然数，最小的质数是 2。质数在数论中具有重要地位，因为任何大于 1 的自然数都可以唯一地分解成质数的乘积，这个过程叫做质因数分解。质数在现代密码学中，尤其是在公钥加密算法中有着至关重要的作用。

这些数论概念提供了许多有效的工具，用于简化和加速数学计算，尤其在大规模数据处理和计算中，数论方法显示出了极大的潜力。

2.2 深度学习中的数论基础

在深度学习中，计算效率和资源优化是研究的核心问题之一。随着模型的不断发展，如何在保证模型性能的前提下提高计算效率成为了一个迫切需要解决的问题。数论方法，尤其是模运算和同余理论，已经被逐步引入到深度学习的多个方面，包括卷积操作和数据增强等，以提升计算效率、减少计算资源消耗，并改善模型的性能。

1. 数论在卷积运算中的应用

   卷积神经网络是深度学习中最常用的结构之一，卷积运算的计算复杂度较高，尤其是在处理大规模图像时。数论中的模运算为卷积运算提供了优化思路。传统的卷积运算依赖于大量的乘法和加法操作，这些操作需要大量的计算资源，而通过引入模运算，可以将卷积过程中的结果限制在一定的数值范围内，从而减少计算复杂度。例如：

   - 在卷积运算中，对中间结果进行模运算（例如模 17 或模 256）能够有效减少大数计算，并加速计算过程。
   - 模运算还可以限制结果的大小，避免溢出问题，提高数值稳定性，并且可以通过降低计算精度来减少资源消耗。

2. 数论在数据增强中的应用

   数据增强是深度学习中提高模型泛化能力的重要手段。传统的数据增强方法通过对图像进行旋转、翻转、裁剪等随机变换，增加训练数据的多样性。然而，这些方法可能会导致增强后的数据分布与实际数据分布存在一定差异，从而影响模型的泛化能力。

   数论方法为数据增强提供了新的思路，尤其是在旋转、裁剪等操作中。利用数论中的模运算，可以为旋转角度、裁剪比例等参数生成具有规律性的值，从而使得数据增强过程更加稳定且具有较强的规律性。举例如下：

   - 在图像旋转中，可以通过模运算生成旋转角度，而不是完全随机选择角度。例如，可以使用模 360 的方式生成一组固定的旋转角度，使得图像变换更有规律性，避免过度随机化导致的数据分布偏差。
   - 在图像裁剪中，使用数论中的序列生成裁剪比例，能够确保每次裁剪操作的比例符合一定的数学规律，这样既能增加数据的多样性，又能保持一定的结构一致性。

   通过结合数论的规律性与传统数据增强方法的随机性，可以在提高模型泛化能力的同时，减少过拟合的风险，从而使模型在测试集上的表现更为稳定。

综上所述，数论在深度学习中的应用，特别是在卷积运算和数据增强方面，展示了其强大的潜力。通过数论模运算的引入，可以优化深度学习中的计算效率、减少训练时间、提高内存使用效率，并增强模型的泛化能力。

3 优化卷积运算

3.1 传统卷积运算的局限性

卷积层作为 CNN 的核心组成部分，通过卷积运算提取图像中的特征。然而，传统的卷积运算在面对大规模数据集和深度神经网络时，暴露出了计算复杂度高的局限性：

传统卷积运算的高计算复杂度是 CNN 性能瓶颈的关键因素之一。[2] 假设输入图像的尺寸为 $H \times W$，卷积核的尺寸为 $K \times K$，且输入和输出的通道数分别为 C_{in} 和 C_{out}。那么，一个标准的卷积操作的计算量可以表示为：

$$\text{FLOPs} = C_{in} \times C_{out} \times K^2 \times H \times W$$

其中，FLOPs (Floating Point Operations) 表示浮点运算数。随着输入图像尺寸 $H \times W$ 和卷积核大小 $K \times K$ 的增加，卷积运算的计算量呈指数级增长。例如，假设在处理高分辨率图像（如 224x224 像素）时，标准的卷积层需要在每一层执行大量的乘法和加法运算。每个卷积核都需要对整个输入图像进行操作，这就导致了计算上的冗余，尤其是卷积核的大小和步长设置固定时，卷积操作往往无法自适应输入数据的特点，从而增加了不必要的计算工作量。如此庞大的计算量直接导致了训练时间的大幅增加，尤其在处理如 ImageNet、COCO 等大规模数据集时，每一次迭代的时间消耗往往达到数十秒甚至数分钟，训练时长可延续数天或数周。

此外，随着卷积层数的增加，深层网络的训练时间不仅仅依赖于卷积运算的计算量，还与每一层的梯度计算及反向传播过程中的计算复杂度密切相关。反向传播需要存储每一层的激活值和梯度信息，增加了内存占用，并在更新权重时要求进行大量的计算。更高的层数意味着需要更多的梯度更新，这进一步加重了计算负担，从而延长了训练时间。

3.2 数论模运算优化卷积方法

数论模运算作为一种基础的数论运算，其核心思想是对数值进行模操作，即将一个数除以一个给定的模数后得到一个余数。例如：给定一个整数 a，它对模数 m 进行模运算的结果是一个余数用 a mod m 来表示。数论模运算有一个重要的性质：数值在经过模运算后变得更小，但是尽管这个数值发生了变化，但是其运算的逻辑依然能够成立（无论是经过加法模运算、减法模运算、乘法模运算）。

在卷积神经网络中，卷积层通过滑动卷积核与输入特征图进行逐元素乘法和求和。引入模运算后，卷积运算中的每个加法和乘法步骤都会被替换为模运算，从而可以让卷积结果的数值变小，计算速度加快。原本标准的卷积运算：设输入特征图为 X，卷积核为 W，输出为 Y。可以表示为：

$$Y = \sum_i (X_i \times W_i)$$

在加入模运算后，卷积运算可以用如下来表示：

$$Y = \left(\sum_i (X_i \times W_i) \right) \mod m$$

利用数论模运算的重要性质：结合律，可以在 Xi 和 Wi 在计算乘法和加法之前进行模运算，让两者的数值变小，让其计算难度降低，加快运算速度：

$$Y = \left(\sum_{i=1}^{n} ((x_i \mod m) \times (w_i \mod m)) \right) \mod m$$

3.3 实验

3.3.1 实验设计

1. 实验目的: 本实验旨在验证数论模运算在卷积神经网络中对卷积计算的加速效果。具体过程是通过在深度网络 ResNet18 的卷积层中引入数论模运算（ModConv2D），与传统卷积方法进行对比，考察其在训练时间上的提升。

2. 实验数据与模型结构: 本实验数据采用的是 CIFAR-10 数据集。CIFAR-10 是一个自然生活物体的彩色图像数据集。一共包含 10 种类别的 RGB 彩色图片：飞机、汽车、鸟类、猫、鹿、狗、蛙类、马、船和卡车。每个图片的尺寸为 32×32 ，每个类别有 6000 个图像，数据集中一共有 50000 张训练图片和 10000 张测试图片。下面是数据集图片的示例图1:

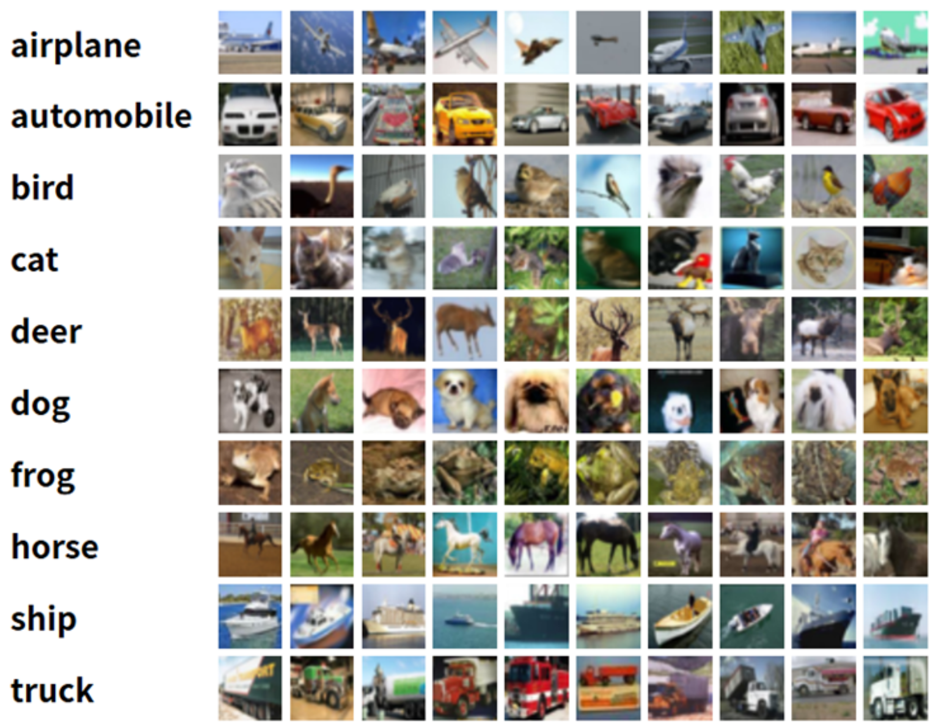

图 1: 数据集示意图

本实验中，我们采用了 ResNet-18 模型结构作为基础模型结构，并在此基础上进行改进以便研究模运算在卷积神经网络中的加速效果和时间复杂度优化。下面为模型结构设计:

(a) 传统卷积策略模型结构采用的是 ResNet-18 架构。

- 输入层:CIFAR-10 数据集的图像输入，尺寸为 32x32x3（32x32 的 RGB 图像）。
- 卷积层: 经过一层 7x7 卷积，输出 64 个特征图，步长为 2，随后进行最大池化（Max Pooling）操作。

- 残差块: 每个残差块由两个 3x3 卷积层和一个跳跃连接组成。ResNet-18 由 4 个阶段组成，阶段间的特征图数量逐步增加。第 1 阶段：64 个卷积核，2 个残差块。第 2 阶段：128 个卷积核，2 个残差块。第 3 阶段：256 个卷积核，2 个残差块。第 4 阶段：512 个卷积核，2 个残差块。
- 全连接层：最后，ResNet-18 通过一个全连接层将卷积后的特征映射转换为 10 个类别的输出。
- 输出层：通过 softmax 激活函数输出概率分布。

(b) 数论模运算优化卷积模型结构

在这个创新模型中,传统卷积神经网络中的每个卷积层被替换为自定义的 Modular Conv Layer（模运算卷积层）。这个自定义层在传统卷积层的基础上进行了数论优化。它的结构与传统卷积层几乎相同，但在卷积运算操作后的结果上加入了模运算。

取模数被设置为 17，从而保证卷积输出的所有的元素都被限制在 [0, 16] 的范围内，从而降低运算的复杂度。假设有一个卷积层计算得到的输出：

$$x = \begin{bmatrix} 18 & 23 & 4 \\ 19 & 30 & 5 \\ 7 & 6 & 16 \end{bmatrix}$$

执行 $x \mod 17$ 后，每个元素的取余操作会得到：

$$x \mod 17 = \begin{bmatrix} 1 & 6 & 4 \\ 2 & 13 & 5 \\ 7 & 6 & 16 \end{bmatrix}$$

所有的元素都被限制在 $[0, 16]$ 的范围内。

3. 评估指标为了对比传统卷积运算策略和基于数论模运算卷积策略的效果, 本实验定义了五个指标, 并通过这五个指标来帮助我们判断后者是否能够在减少细微精度的基础上减少训练所需时间。

   - 训练时间对比:
     这两种模型在相同训练任务下的总体训练所需时间。在相同 epoch 数下，较低的训练时长意味着策略是有效的。
   - 训练损失值对比：
     在训练集上每个 epoch 的平均损失值。在相同 epoch 数下，两种策略的训练损失值相差细微意味着策略是有效的。
   - 验证损失值对比：
     在验证集上每个 epoch 的平均损失值。在相同 epoch 数下，两种策略的验证损失值相差细微意味着策略是有效的。
   - 训练精度对比：
     在训练集上训练结束后的准确率。精度越高，意味着在训练过程中该策略提升了模型的泛化能力。

- 测试精度对比：
  在测试集上训练结束后的准确率。精度越高，意味着在测试过程中该策略提升了模型的泛化能力。

4. 实验结果展示

   (a) 首先是训练时长这一个评估指标的对比图2:

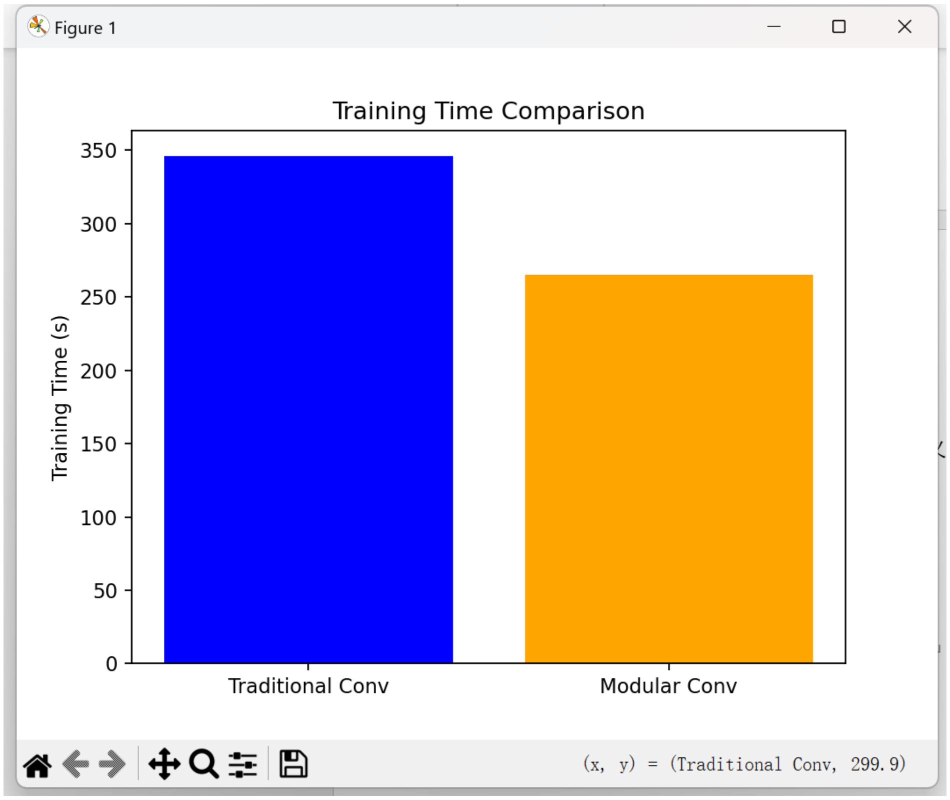

图 2: 训练时长对比图

   (b) 其次是训练损失值这一个评估指标的对比图3

   (c) 之后是验证损失值这一个评估指标的对比图4

   (d) 之后是训练精度这一个评估指标的对比图5

   (e) 最后是测试精度这一个评估指标的对比图6

3.3.2 实验结果分析

我们可以从图中清晰看出基于数论模运算卷积策略通过加速卷积运算操作，降低了训练过程中的计算复杂度，最终导致了训练时长的缩短。表明采用数论优化的方法的确可以有效提升

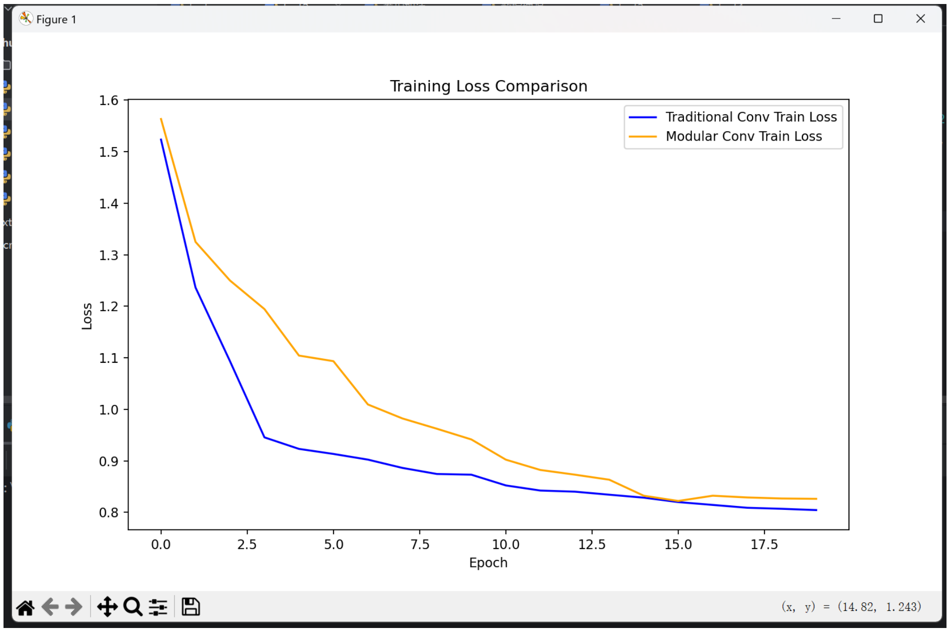

图 3：训练损失值对比图

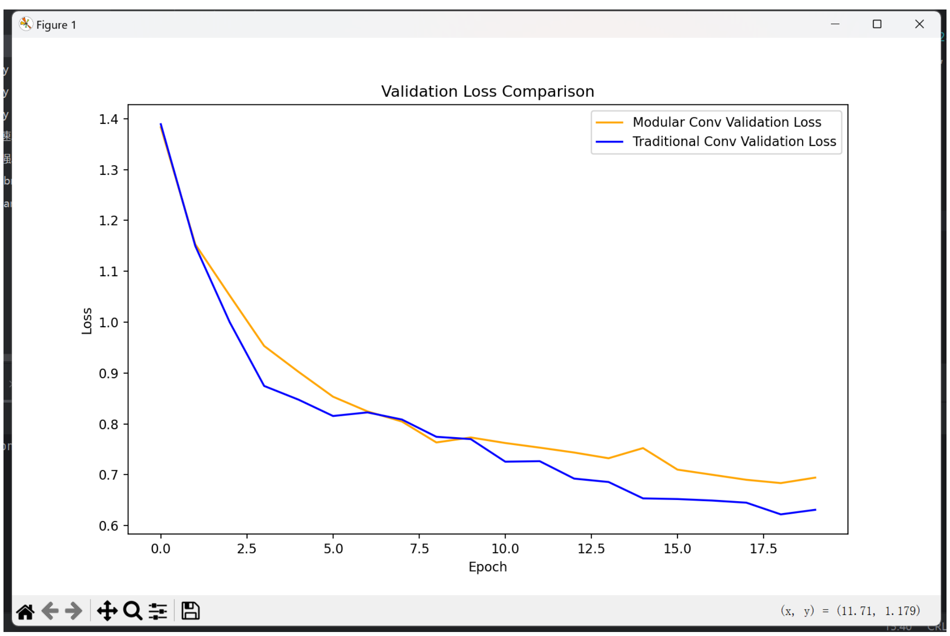

图 4：验证损失值对比图

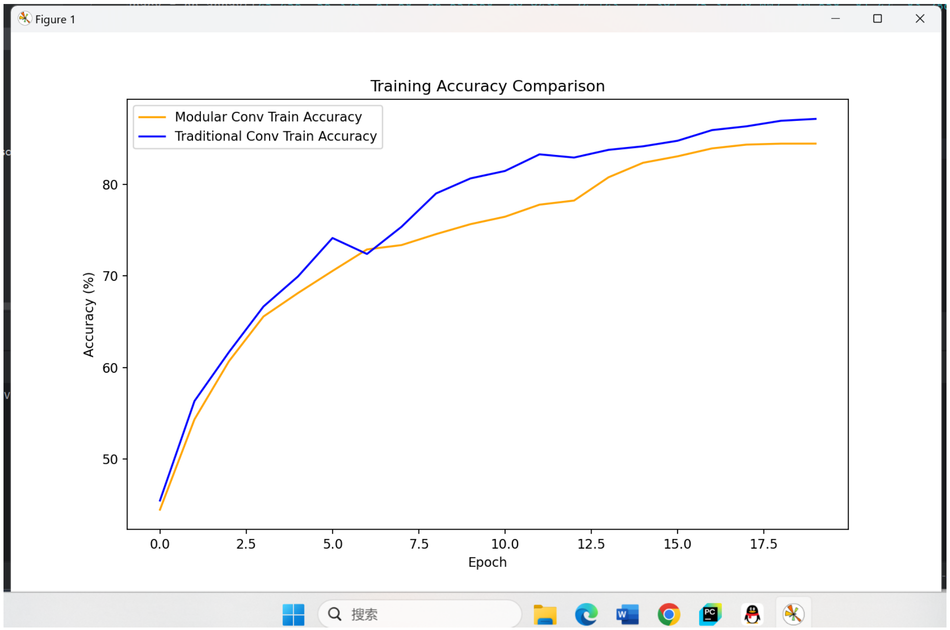

图 5: 训练精度对比图

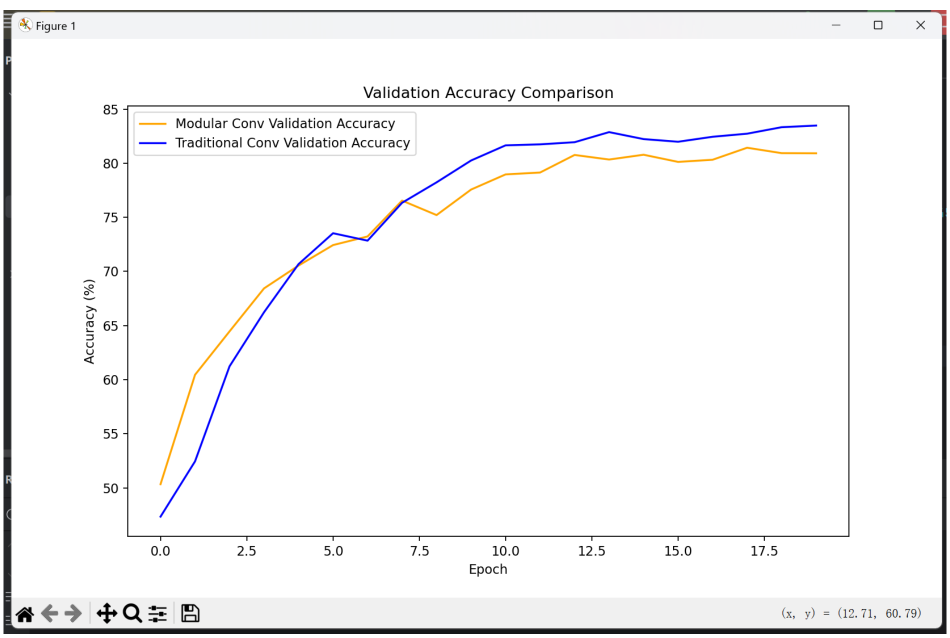

图 6: 测试精度对比图

效率，节省时间。而且该策略的训练损失和测试损失与传统策略相比，损失值虽然高出一定值，但差距非常小，几乎可以忽略不计，表明两种方法在优化模型和调整超参数方面基本相当。最后该策略的训练精度和测试精度稍微低于传统策略，但差距非常小，几乎可以忽略不计。表明数论模运算在减少训练时长的基础上提高训练效率，但对模型的最终精度（训练精度和测试精度）影响并不显著。

基于上述实验结果，我们可以总结出：数论模运算卷积策略在降低训练时长方面展示了显著的优势，但其代价是精度上的轻微损失。这种现象可以理解为在一定程度上牺牲计算精度来提高训练效率，从而减少训练所需的时间。因此，这种方法尤其适用于需要大规模训练且对精度要求不极端严格的应用场景，例如一些实时系统、大规模数据处理任务或需要频繁迭代调优的任务。

4 优化数据增强

4.1 传统数据增强的局限性

传统的数据增强方法通常通过简单的几何变换来增加训练数据的多样性，以提高模型的泛化能力。在本实验中，传统的增强方法包括随机水平翻转、随机旋转（最大角度 30°）、以及随机裁剪等操作。这些变换虽然能有效避免模型对训练数据的过拟合，并提升其在未知数据上的表现，但仍存在一些明显的局限性 [6]：

- 变换的规则性：例如，随机旋转方法仅选择从固定范围内（-30° 到 30°）的旋转角度，而这些角度的选择并非完全随机或多样化。因此，传统方法生成的样本变化有一定的局限性，可能无法充分模拟实际场景中的复杂数据分布。
- 计算效率：尽管传统的增广方法相对简单，但在大规模训练过程中，计算效率仍然可能成为瓶颈，尤其是当训练集非常庞大时。这些增广操作需要在每次训练时对每个批次的数据进行计算，这增加了训练的总时间。
- 增强效果的有限性：传统的数据增强方法通过对图像进行旋转、翻转和裁剪等基本变换来扩充数据集。虽然这些操作能够在一定程度上提升模型的表现，但它们的增效作用在数据集较为简单或数据多样性较低的情况下较为有限。在实际应用中，传统方法可能未能充分挖掘数据的潜在变换空间，导致训练效果不够理想。

4.2 数论模运算优化数据增强

本文提出基于数论的模运算优化数据增强的方法，通过数论中模运算和低差序列的应用，引入更加高效和具有理论依据的数据增强策略。具体来说，数论提供了可以应用于数据增强中的几个关键概念：旋转角度的非均匀选择、低差序列的裁剪比例生成，以及这些方法如何优化图像增广的计算过程。

首先，通过模运算生成旋转角度。传统的数据增强方法使用随机旋转角度来改变图像方向，然而这些角度通常是在 [0, 360) 范围内均匀随机选择的，这导致了旋转变换具有一定的规律性。为了解决这一问题，我们采用模运算生成旋转角度。具体而言，使用 (i * 7) mod 360 公

式生成旋转角度，其中 i 为整数，且 7 为模运算的乘数。通过这种方式，数据增强图片生成的角度序列将会更加多样，从而避免了常规旋转方法中角度分布的规则性，并且引入更多的数学约束，为模型提供了更广泛的训练样本。

4.3 实验

4.3.1 实验设计

- 实验目的

  比较基于数论的数据增强方法与传统数据增强方法在训练深度神经网络时的性能差异，特别是在 CIFAR-10 数据集上训练 ResNet-18 模型的效果。

- 实验数据与模型结构

  数据集：使用 CIFAR-10 数据集，与实验一的数据集一致，这里不过多赘述。

  传统数据增强方法：包括随机水平翻转、随机旋转（最大 30 度）、随机裁剪（大小调整到 32×32）。

  基于数论的数据增强方法：(i * 7) mod 360 生成的是 7 的倍数与 360 取余后的结果，即为我们得到一组旋转角度，其中 i 的取值范围为 0 到 9，从而可以生成 10 个不同的旋转角度。

  总体流程为：首先，图像通过随机水平翻转进行变换，然后进行旋转和裁剪，最后进行标准化处理。通过数论生成的旋转角度使得每次图像变换的结果都不完全相同。

  具体代码如图7。

```python
def number_theory_augmentation():  1 usage
    def mod_rotation(img):
        angles = [(i * 7) % 360 for i in range(10)]  # 数论生成旋转角度
        angle = angles[torch.randint(0, len(angles), (1,)).item()]
        return transforms.functional.rotate(img, angle)


    return transforms.Compose([
        transforms.RandomHorizontalFlip(),
        transforms.Lambda(mod_rotation),
        transforms.ToTensor(),
        transforms.Normalize( mean: (0.5, 0.5, 0.5),  std: (0.5, 0.5, 0.5))
    ])
```

图 7: 数论数据增强代码图

  该实验采用的 ResNet-18 模型与实验一的传统模型一致，这里不过多赘述。

- 评估指标: 为了对比传统数据增强策略和基于数论模运算数据增强策略的效果，本实验定义了四个指标，并通过这四个指标来帮助我们判断后者是否能够提高训练效率、模型的精确率以及泛化能力。

  (a) 训练以及测试损失值对比：

  在数据集上每个 epoch 的平均损失值。在相同 epoch 数下，基于数论模运算数据

增强策略的损失值较低且降低速度快，意味着该策略是有效的。

(b) 训练以及测试预测率对比：

在数据集上训练和测试结束后的准确率。在相同 epoch 数下，基于数论模运算数据增强策略的精度越高，意味着在整体过程中该策略提升了模型的泛化能力

(c) 召回率对比：

基于数论模运算数据增强策略若召回率高，意味着模型能捕获大部分正类样本，漏检率较低。

(d) F1 分数对比：

基于数论模运算数据增强策略若 F1 分数高，意味着模型在准确性和召回能力上都有更好的表现。

- 实验结果展示
  - 首先是训练以及测试损失值这一个评估指标的对比图8:

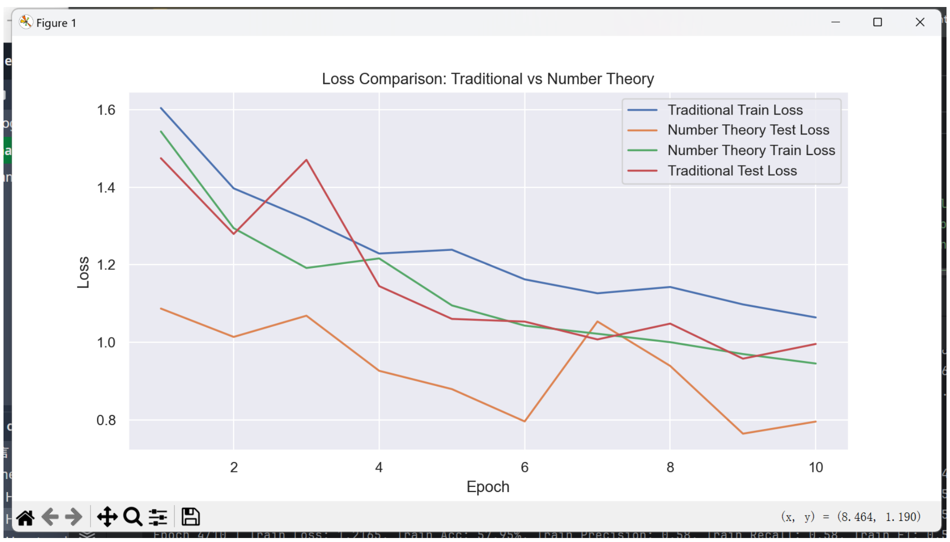

图 8: 训练以及测试损失值对比图

  - 其次是训练以及测试预测率这一个评估指标的对比图9:
  - 之后是召回率这一个评估指标的对比图10:
  - 最后是 F1 分数这一个评估指标的对比图11:

4.3.2 实验结果分析

在本次实验中，我们比较了传统数据增强策略与基于数论模运算的数据增强策略在训练和测试过程中的表现。实验结果表明，基于数论的创新增强策略无论在训练损失、测试损失、训练准确率、测试准确率、召回率，还是 F1 分数等指标上，都显著优于传统增强方法。

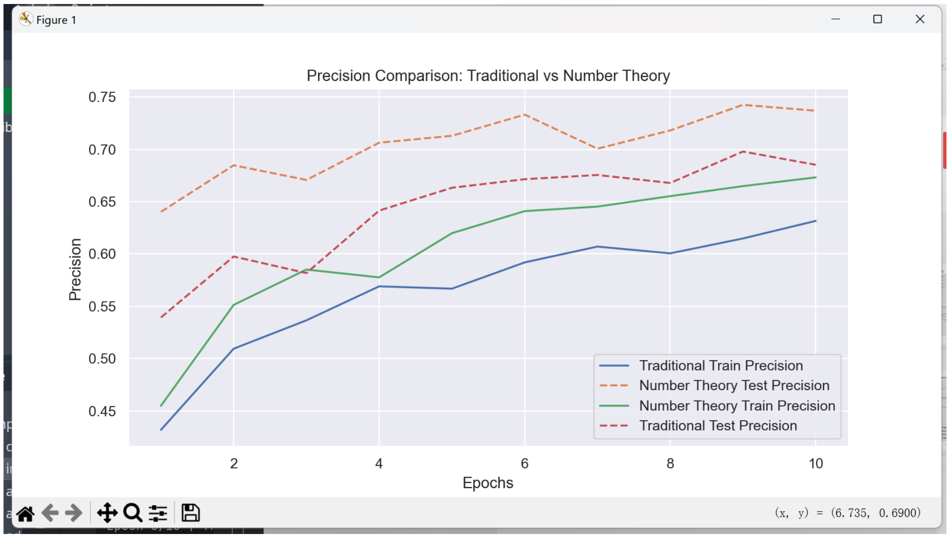

图 9: 训练以及测试预测率对比图

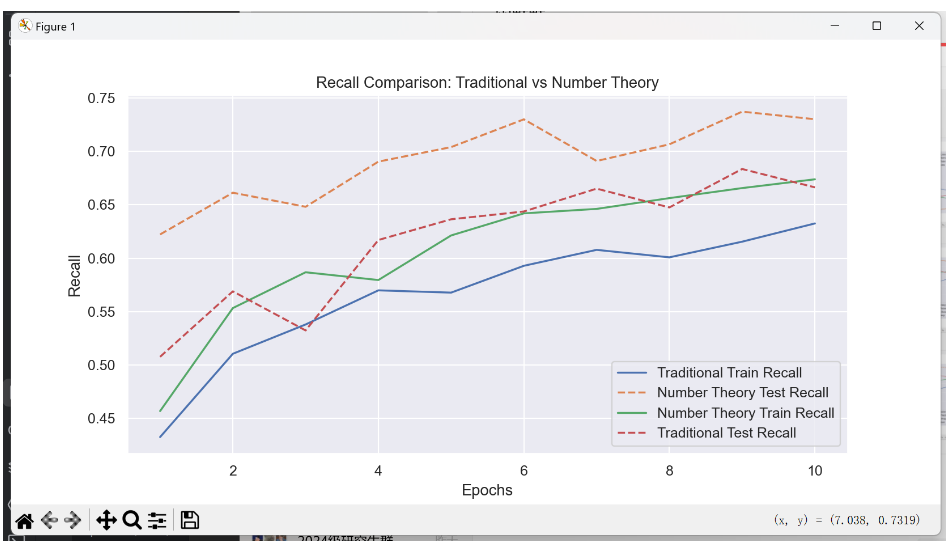

图 10: 召回率对比图

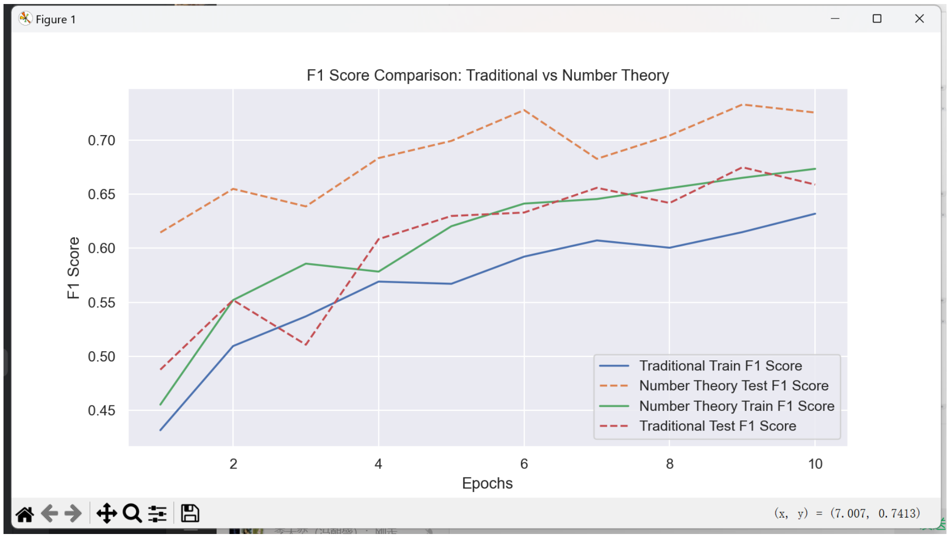

图 11: F1 分数对比图

(a) 训练和测试损失分析

在训练过程中，基于数论模运算的数据增强策略表现出了更快的收敛速度，训练损失下降较为平稳且迅速。这表明，数论增强策略能够更有效地为模型提供多样化的训练数据，帮助模型更快地学习到有效的特征。相比之下，传统数据增强策略的训练损失下降速度较慢，可能由于变换方式较为简单，导致模型在复杂数据分布上的学习效果受到限制。在测试阶段，前者的测试损失也明显低于传统方法。这表明，该策略不仅在训练阶段提升了模型的学习效率，还有效提升了模型的泛化能力，帮助其在未见过的数据上表现更好。

(b) 训练和测试准确率分析

在训练和测试准确率方面，基于数论模运算的数据增强策略都显著优于传统策略。特别是在测试准确率方面，前者的提升更为明显，表明其生成的多样化数据有效地提高了模型对测试集的预测能力。这一结果表明，该策略成功地增强了模型的泛化能力，降低了过拟合风险。

(c) 召回率和 F1 分数分析

除了准确率，召回率和 F1 分数也是评价分类模型性能的重要指标。基于数论模运算的数据增强策略在这两个指标上也有显著优势。召回率的提高意味着模型能够更好地识别正类样本，减少漏检。而 F1 分数的提升则表明，在平衡精确率和召回率方面，该策略能够提供更加均衡的性能。相比之下，传统策略的召回率和 F1 分数较低，可能因为其变换样本的多样性不足，导致模型对不同类别的识别能力较弱。

5 总结和展望

本文主要探讨了数论模运算在优化深度学习效率中的应用，特别是在卷积运算和数据增强方面的创新性应用。通过引入数论中的模运算，我们提出了一种新的思路来提高深度神经网络的计算效率与训练效果。

首先，我们将数论模运算应用于卷积神经网络的卷积过程，通过优化卷积运算的计算路径，有效地降低了卷积操作的时间复杂度。实验结果表明，数论优化后的卷积运算不仅提高了计算效率，还在保证模型精度的前提下，减少了训练时间和内存消耗，展现出了巨大的潜力。

其次，在数据增强方面，我们创新性地结合了利用数论模运算来生成旋转角度的策略。使得其更具随机性和多样性，避免了传统数据增强方法中角度选择的单一性；实验结果显示，基于数论的数据增强方法在训练和测试准确率、召回率、F1 分数等指标上均优于传统方法，进一步证明了其在提升模型性能方面的优势。

虽然本文的研究成果表明，数论模运算在卷积运算和数据增强方面具有显著的优化效果，但这仅仅是数论在深度学习中的一种应用。在未来的研究中，数论的方法有潜力在其他深度学习任务中发挥更大的作用，具体方向包括但不限于：

数论优化的网络架构设计：除了卷积运算，数论可以应用于其他网络层的优化，如全连接层、池化层等。通过数论方法降低计算复杂度、提升计算效率，可能为构建轻量级神经网络提供新的思路，特别是在边缘计算和资源受限设备上应用时，具有重要意义。

数论在强化学习中的应用：在强化学习中，特别是 Q-learning 和深度强化学习中，数论方法可以用于优化动作选择的策略空间，利用数论生成的低差序列来改进探索策略，从而提升学习效率与最终的决策质量。

高效的数值优化方法：数论中已有的优化算法，如基于模运算的快速傅里叶变换（FFT）等，可以应用于深度学习中的训练过程优化，尤其是加速梯度计算和网络参数的更新，减少计算时间并提高训练效率。

多任务学习中的数论应用：数论还可以为多任务学习提供新的优化策略。在多任务学习中，不同任务的学习过程可能有相互影响，而数论方法通过优化网络中的共享参数与任务间的关系，有望提高多任务学习的效率和效果。

总之，数论在深度学习中的应用远未达到极限，随着研究的深入，数论的更多方法和思想有望为深度学习领域带来革命性的突破。我们相信，数论的独特性质将在未来的深度学习任务中扮演更加重要的角色，推动深度学习技术向更高效、更智能的方向发展。

参考文献

[1] A. Gioia and 陈宏基. 数论导引. *惠阳师专学报 (自然科学版)*, S1:55–65, 1985.

[2] 刘志鹏, 瞿哲, 于聪, and 等. 基于深度图卷积神经网络的 gis 设备故障诊断方法. *电气传动*, 54(12):86–93, 2024.

[3] 李越, 唐聃, 孙敏钧, and 等. 基于二维模运算的高效率可逆信息隐藏方案. *计算机应用*, 44(06):1880–1888, 2024.

[4] 杜珊妮. 4102 万位! 已知最大质数诞生. *中国科学报*, (002), 10 2024.

[5] 赵军勇, 胡双年, and 尹秋雨. 一类模 n 四次对角同余方程的单位解的个数. *Wuhan University Journal of Natural Sciences*, 29(05):391–396, 2024.

[6] 陈舸 and 王中卿. 结合预训练模型和数据增强的跨领域属性级情感分析研究. 计算机科学, pages 1–13, 2024. Online.