# 基于高血压眼病诊断问题的二分类任务综合研究

李昊　　汪毅恒　　赵泓博

## I. 摘要

随着深度学习跳跃式的发展，图像二分类问题似乎迎来了最终的答案。而医疗领域，这一集合了大量二分类任务的领域，对高性能、高性价比的二分类模型需求日益增加。同时，随着深度学习技术在社会各个领域的进一步发展，成熟的深度学习模型面临的将不仅仅是性能的要求，更多的是需要平衡性能、成本、隐私性等各方面的综合考量。本研究以基于小规模眼底图像数据集的高血压眼病诊断二分类问题为落脚点，通过模型参数微调实验、多模型训练综合对比实验、数据增强实验和模型聚合实验，最终给出综合考量后的最优解决方案。我们的模型在官方网站上达到了最高 0.646 的评分，也是截至报告编写时间的最高分。我们的工程代码已上传至 https://github.com/Anko-Official/Hypertensive-Retinopathy-Diagnosis-Challenge

## II. 研究介绍

随着技术的发展，机器学习在各个领域都得到了广泛的应用。在医疗保健、金融，零售和电子商务、智能交通以及生产制造等领域，机器学习发挥着越来越重要的作用。人工智能对社会的影响是深远且多方面的。经济方面，人工智能的发展极大地推动了经济的增长和创新；生活方面，AI 技术为人们提供了更便捷、更高效的服务，提高，也为医疗、教育等领域带来了革命性的变化。机器学习在提取图像特征方面展现出了优异的性能。机器学习算法能够自动地从数据中学习到更加抽象和高级的特征，从而提高了特征提取的准确性和泛化能力。此外，机器学习算法还具有强大的优化和自适应能力，以及较高的灵活性和通用性。机器学习拥有众多优势，使得其在图像处理和计算机视觉领域具有广泛的应用前景。

二分类图像分类任务在现实生活中具有广泛的应用场景。在医疗诊断、在安防监控、自动驾驶等方面，都有众多任务任务都可以转化为二分类图像分类问题，通过训练模型来自动进行识别和判断。从成本效益的角度来看，二分类图像分类技术具有较高的性价比。随着机器学习算法和计算资源的不断发展，图像分类任务的准确性和效率得到了显著提升。同时，通过合理地设计模型结构和参数，可以在保证性能的前提下降低计算资源的需求，进一步降低成本。基于高血压眼病诊断问题的二分类任务综合研究有助于及早发现和诊断高血压眼病，并提高诊断的准确性和效率。重要的是，我们的工作是具有很高的可扩展性的，可以给同类型的问题提供经验。

Alexnet [14] 的问世标志着 CV 领域正式进入了深度学习时代。此后各种更加复杂和高效的 CNN 架构相继出现，如 VGG [12]、GoogLeNet [13]、ResNet [4], 这些网络通过更深的层次、残差连接和注意力机制等创新，大幅提高了图像分类的准确率。而在随着类 transformer [17] 架构爆发出意料之外的智能，图像二分类领域的机器学习模型有了新的发展。谷歌在 2020 年提出的 VIT（Vision Transformer）[18] 也成为我们研究的重点之一。同时，我们也进行了多模型性能的比较，在第四部分实验篇有更详细的介绍。

本研究相较于其他二分类问题最重要的区别有两点，一是**数据集规模小**，经典图像数据网站，如 ImageNet [3]，其训练图可以达到 120 万张以上。然而，我们的测试数据仅有 712 张，这是我们进行研究的重要挑战和限制。二是**应用场景限制**。医学领域，分类模型对隐私性的要求更高，这意味着我们需要让边缘端承担大部分中心服务器的计算任务，这也是本项目重大挑战之一。从计算能力的角度来看，边缘设备通常具有有限的计算能力，与中心服务器相比，其计算资源相对匮乏。因此，如何在保证推理精度的同时，降低模型的复杂度，使其能够在边缘设备上高效运行，是一个重要的挑战。此外，随着模型的不断升级和优化，如何确保边缘设备能够跟上这种变化，也是需要考虑的问题。

为了追求更高的准确性，深度学习模型的结构变得越来越复杂，网络也越来越深。参数数量的增加意味着需要更多的数据来训练模型。然而，手动标注数据成本高昂，而且由于客观原因，在某些特定领域收集数据并不容易。因此，数据不足是一个非常普遍的问题。数据增强技术可以通过人工生成新数据来缓解这一问题。在计算机视觉领域的成功应用促使人们考虑将类似方法应用于序列数据。数据增强 [16] 在计算机视觉领域应用较多，主要是通过各种技术生成新的训练样本。例如，通过对图像的平移、旋转、压缩、调整色彩等方式，可以在不改变样本标签的前提下，

创造出更多样化的数据。这种方法有助于模型学习到更为全面的特征，提高其在不同场景下的泛化能力。

我们的贡献如下：在多个维度上对异构模型二分类任务上的表现进行了比较；利用数据增强的方法增强了模型的学习效果，取得了显著的性能提升；针对应用场景对模型部署作出了优化和调整。

## III. 研究背景

在机器学习和深度学习中，二分类任务是指将输入数据划分为两个预定义类别的问题。这类任务广泛应用于各种场景，如图像识别、垃圾邮件检测等。在处理二分类任务时，模型通常会首先学习从输入数据中提取有用的特征。模型通过训练过程，不断调整其内部参数，以更好地捕捉这些特征。对于图像特征提取，深度学习模型表现出了强大的能力。例如，卷积神经网络能够逐层提取图像的层次化特征。每一层都在前一层的基础上进一步抽象和提取信息，使得模型能够学习到越来越复杂的特征表示。在二分类任务中，模型会特别关注那些与两个类别区分度最高的特征。这些特征能够帮助模型在输入数据中找到区分两个类别的关键信息。通过优化损失函数，模型会逐渐学会更加精确地提取和利用这些特征来进行分类。此外，模型的性能还受到其结构和训练方式的影响。选择合适的网络结构、使用适当的激活函数和损失函数、以及采用有效的训练策略，都有助于提高模型在二分类任务上的性能 [19]。

我们在研究中用到了以下四种模型。ResNet（残差网络），基于残差学习的概念，其中涉及跳过某些层的连接（也称为跳过连接）。ResNet-18 由 18 层组成，包括卷积层、池化层和残差块。其中，卷积层用于特征提取。残差块中每个区块由两个卷积层组成，具有批量归一化和 ReLU 激活功能，以及一个跳跃连接，可将输入添加到两个卷积层的输出中。池化层：通常用于降采样。与 ResNet-18 相比，ResNet-50 是更深的 ResNet 变体。它由 50 层组成，包括卷积层、池化层以及比 ResNet-18 更多的残差块。EfficientNet [15] 基于一种复合缩放方法，可统一缩放深度/宽度/分辨率维度。它使用移动倒置瓶颈卷积（MBConv）模块作为基本构建模块。EfficientNet 变体用复合缩放因子标示深度、宽度和分辨率。其中的 MBConv 块包含深度可分离卷积和用于通道关注的挤压-激发（SE）区块。VIT（Vision Transformer）与卷积神经网络（CNN）不同，ViT 采用纯粹基于变换器的架构进行图像分类。它将输入图像分割成固定大小的片段，将其线性嵌入，然后通过变换器编码器馈送。VIT 将图像补丁转换为嵌入。并使用特别的变换器-编码器结构（由多个变压器模块组成，每个模块都包含多头自注意和前馈神经网络）。

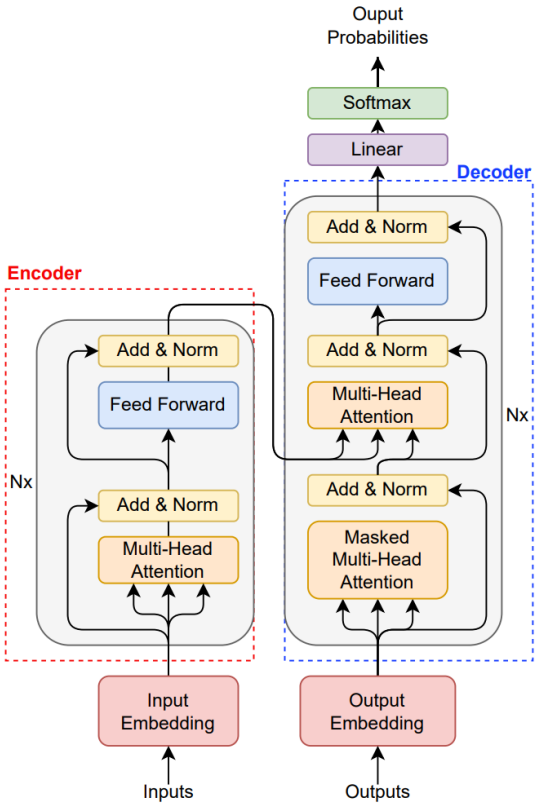


Fig. 1: The Structure of The Classic Transformer Model.

数据增强可以优化在小规模数据集上模型的训练效果，因为其可以弥补诸多小规模数据集的不足与限制。对于小规模的数据集，使用数据增强技术可以通过对原始数据进行变换，来扩充数据集的大小，从而增加模型训练的样本量。使得模型可以学习到更多的特征和信息，有助于模型更好地拟合数据分布。并且，数据增强可以通过增加数据的多样性和复杂度，来减少过拟合的风险。一些深度神经网络模型层数较多导致的学习能力较强，将图像数据样本中的特征学习的过于充分，使得神经网络模型在训练数据上出现过拟合现象。 [11] 通过随机变换操作，数据增强可以模拟真实场景中的变化 [10]，使模型能够更好地适应新数据，提高模型的鲁棒性和准确性。此外，数据增强可以减轻数据收集和标注的成本。数据增强技术可以在不增加额外成本的情况下，从有限的样本中生成更多样本，从而扩展数据集规模和多样性。

医学领域对实时性和快速响应的要求非常高。因此，将模型部署在边缘设备上，能够极大地减少数据传输和处理的延迟，确保医生或医疗设备能够实时获取到模型的预测

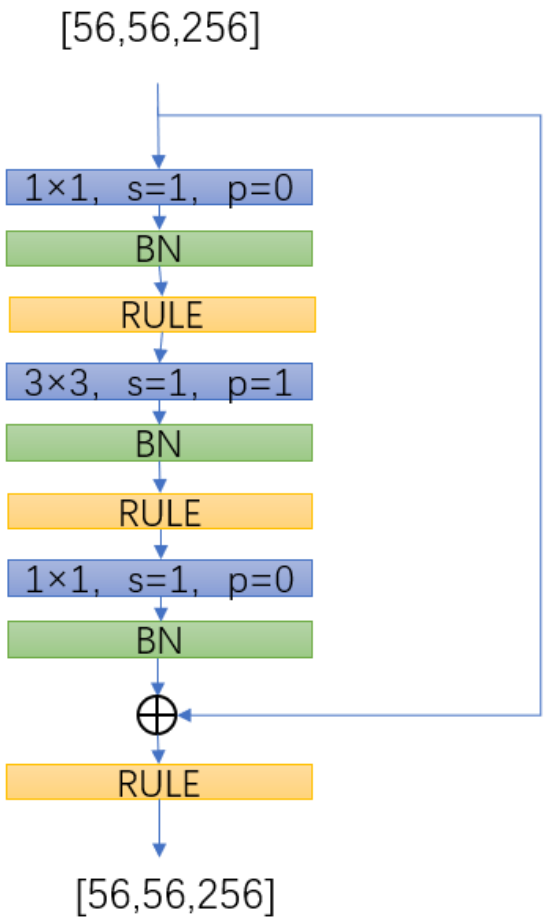


Fig. 2: The Structure of ResNet18.

结果，从而做出迅速且准确的决策。并且，医学领域的数据隐私和安全性至关重要。医疗数据包含大量个人隐私信息，如果将所有数据都传输到云端进行处理，将面临数据泄露的风险。而在边缘场景下部署模型，可以实现在本地设备上处理数据，避免了数据传输过程中的安全风险，同时也确保了数据的隐私性。此外，医学领域的数据多样性和复杂性也是推动在边缘场景下部署模型的重要因素。医疗数据还包括大量非结构化数据，如语音、文本等。这些数据在边缘设备上进行处理和分析，可以更好地满足医疗场景下的实际需求，提供更为个性化和精准的服务。

在实际生产中，有许多方法用来实现模型性能和成本的平衡，以下举几个例子作为说明：算法优化与模型简化：针对复杂的模型，通过算法优化来减少计算量，提高运行效率。同时，对模型进行简化，去除冗余部分，降低模型复杂度，从而减少成本。这种优化通常在不牺牲过多性能的前提下进行。轻量化网络结构：在深度学习领域，轻量化网络结构被广泛应用于实际生产中。这些网络结构在保持较高性能的同时，显著减少了模型参数和计算量，降低了部署成本。混合精度训练：在训练过程中采用不同精度的数据表示和计算，如使用低精度进行大部分计算，而在关键部分使用高精度，以在保持性能的同时降低计算成本和存储需求。实验表明，经过自动混合精度优化之后的代码，在减少精度冗余的基础上能够充分发挥其并行潜力，提升程序性能。 [] 分布式训练：利用多台机器并行训练模型，加速训练过程，同时分摊计算成本。这种方法适用于大规模数据集和复杂模型。

## IV. 实验设置

我们实验所用的 GPU 是英伟达 RTX4080，CPU 是英特尔 i5-13600KF 处理器。所有模型训练是在 pytorch 环境下进行，python 环境为 3.11。

**数据集及实验信息**：实验所用数据集来自于竞赛官网，包含 712 张眼底图像，其中 292 张患有高血压眼病，420 张未患有高血压眼病。本次实验的目标就是基于这个小规模的数据集，训练出一个深度学习模型，用以对测试集的图片进行分类，并获得相应评分。

**初步模型实验**：实验初期我们调研并部署了 ResNet-18、ResNet-50、EffientNet、Vision Transformer (VIT) 等各种模型，统一设置超参数，并进行 30 个 epoch 的训练。我们将在这一步初步筛选出效果较好的模型，并进行进一步实验。

**模型参数微调实验**：这一步我们需要对初步筛选出的模型进行超参数微调。我们设置在 1e-3, 1e-4, 1e-5, 1e-6 这 4 个不同的学习率下对两种模型进行训练，并记录各个 epoch 下的测试准确率，两种模型在不同学习率下的性能表现如图 3和图 4。

**数据增强实验**：由于本次任务提供的数据集规模很小，仅有 712 张眼底图像，因此模型在训练过程中很容易出现过拟合现象。为了解决上述问题，我们在加载数据级时使用了一些数据增强的方法，包括随机旋转、随机水平与垂直翻转、随机裁剪和缩颜色扰动等。各个模型在数据增强前后的评分如表 I所示。

**模型聚合实验**：在完成选择模型的训练后，我们尝试了模型聚合的方法，通过将两种模型输出的概率分布加和，再求取最终的总体概率分布，获得最终的预测类别。我们还额外尝试了将每种网络结构训练出 2 个较优模型，并且相互组合起来实现模型聚合的效果。单独模型以及模型聚合后的评分如表 II所示。

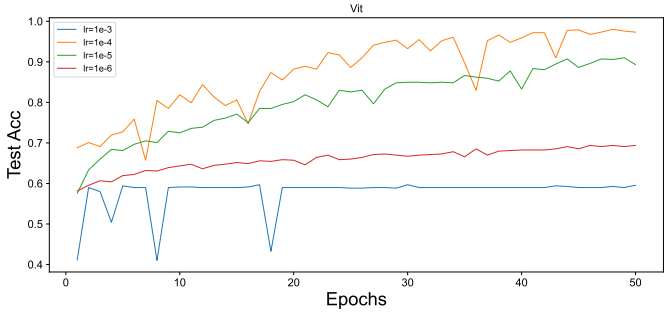


Fig. 3: Comparison of Learning Rate of ResNet-18.

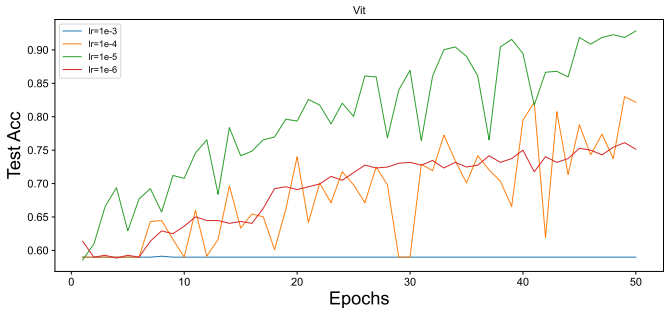


Fig. 4: Comparison of Learning Rate of VIT.

TABLE I: Comparison of Data Enhancement.

| Models | Score Before | Score After |
|---|---|---|
| ResNet-18 | 0.395 | 0.570 |
| Vision transformer | 0.443 | 0.600 |

TABLE II: Comparison of Model Assembly.

| Models | Score |
|---|---|
| ResNet-18 | 0.570 |
| Vision transformer | 0.600 |
| 1 * ResNet-18 + 1 * VIT | 0.646 |
| 2 * ResNet-18 + 1 * VIT | 0.614 |
| 1 * ResNet-18 + 2 * VIT | 0.625 |
| 2 * ResNet-18 + 2 * VIT | 0.615 |

## V. 结果分析

在初步实验中，ResNet-18 和 Vision Transformer (VIT) 表现出了领先于其他模型的性能。因此，我们选用这两个模型进行后续的进一步实验，并分别为它们确定了最佳学习率和学习轮数。从实验结果图 3可以看出，ResNet-18 的最优学习率为 1e-4，但值得一提的是，ResNet-18 在本地测试中获得最高准确率的学习率（1e-4）在官网测试中并未取得最好成绩，而 1e-5 的学习率表现更为良好。我们猜测是因为 1e-4 在小规模数据集上产生了过拟合的效果，达到了局部最优的状态。而 1e-5 泛化性更好，因此在面对新数据集的时候表现得更好。从实验结果图 4可以看出,VIT 在学习率为 1e-5 时收敛速度较快，但是测试准确率震荡现象严重；而学习率为 1e-6 时虽然能够上升平稳，但是模型收敛速度又太慢。因此最终为 VIT 确定的学习率为 5e-6 这个折中值，并且需要延长 VIT 的训练轮数至 80 轮，确保其收敛。

对比两种模型在数据增强实验中的结果表 I可以看出，我们所采用的数据增强方式能够显著提升模型的分类性能。我们推测这是因为数据集太小导致数据离散程度大，包含的信息量较少。而进行一些旋转、翻转、剪裁以及缩放操作，等效于扩充了数据集中的图像，在一定程度上提升了神经网络学习到的信息量，减弱了在小规模训练集上的过拟合效应，从而提升模型的泛化效果，得到更高的评分。在进行数据增强前后,ResNet-18 和 VIT 两种网络结构训练出的模型准确率差距不大，这可能是由于神经网络的训练过程具有随机性和不确定性，且初始化的参数也一定程度决定了网络的最终性能，因此无法判断两者的分类效果是否存在显著差距。但是考虑到 ResNet-18 的参数量仅有 11.2M 而 VIT 的参数量达到了 85.8M，ResNet-18 以相对简单地多的网络结构达到了与 VIT 相近的分类效果，我们可以认为 ResNet-18 网络更加适配于本次实验中的二分类任务。

模型聚合实验本质上相当于综合了异构模型特征提取的能力。由于神经网络的训练具有不确定性和随机性，因此不同网络结构训练出的模型，甚至同样网络结构前后训练出的不同模型，都存在着提取特征的注意力差别。因此我们可以通过模型聚合的方法，综合不同模型最终输出的概率分布，提取到整体的相对较完整的特征，以实现更好的分类性能。从模型聚合实验的结果表 II可以看出，不同的模型聚合方法相比于单独的模型性能都所有提升，1 个 ResNet-18 和 1 个 VIT 进行聚合时得到最了高的评分 0.646，而若继续增加参与聚合的模型数量，则将降低一定的性能。这可能是因为当模型数量增多时，单个模型的独特性将会被削弱，最优模型的表现也会受到次优模型的拖累，最终达到一个“平均”的状态。

最终我们的解决方案在竞赛官网上获得了 0.646 的成绩，是截至报告编写日期的最高分，具体的提交与分数截图可参考附录中的图 5, 图 6, 图 7, 图 8, 图 9 (此处仅展示一些较好的成绩，并非实验过程中的全部提交记录)。排行榜截图可参考附录中的图 10。

## VI. 相关工作

在图像识别方面，有许多论文在算法方面进行改进，达到了减轻运算量、提高效率，或提高准确率的成果。[9] 使用卷积神经网络（CNN）来实现物体探测，将识别准确率从 31% 提高到 53%。该论文采用了上下文识别物体的方法，通过识别照片所在的环境，选择出最可能在环境中出现的预测结果，因此大大提高了命中率。[6] 使用 CNN，在实现和现今技术相近的识别成功率的同时，免去了繁琐的准确面部标志探测，降低了特征维数，也不需要额外生成新的组合以辅助识别。该论文对比了多个 CNN 方法的结果，总结出了各自的优势。[2] 使用 Part Affinity Fields（PAF）实现了多人 2D 动画识别。与已有的方法相比，其最大的优势在于检测的速度对人物的数量不敏感，在保持检测精度的情况下大幅提升了速度。[5] 使用神经网络将高维数据转化为低维，便于将其分类、存储等操作。该论文提出了一种初始化权重的方法，使得神经网络能够处理高维数据，在数据降维方面拥有比 PCA 算法更加优秀的性能。[7] 实现了梯度学习在文档的文字识别中的应用。改论文使用反向传播算法训练多层神经网络。在适当的网络架构下，基于梯度的学习算法可以用于合成复杂的决策面，该决策面可以用最少的预处理对高维模式进行分类，同样是高效处理高维对象的优秀方法。

其他本领域内的贡献包括 [8]，[1]，[3] 等。

## VII. 总结

本研究以基于小规模眼底图像数据集的高血压眼病诊断这个二分类问题为落脚点，通过论文调研和实验分析，综合考量了各种深度学习模型的参数量、有效性、独特性，选择了若干模型进行实际部署用以解决该二分类问题。并且我们在实验中使用模型对比、超参数微调、数据增强和模型聚合等优化方法，在一定程度上解决了数据集规模过小这个问题，有效提升了模型的泛化性与识别准确率，提出了在当前研究范围内的较好解决方案。最终我们的解决方案在竞赛官网测试中取得了 0.646 的评分（也是截至报告编写时间的最高分），在本地测试中也达到了最高 0.98 的准确率。同时，我们在本次研究与实验中使用的思路和方法也具有很高的可扩展性与可移植性，能够便利地迁移到邻近领域的类似分类任务上，为同类问题的解决提供了一种系统性、综合性的解决方案。

| # | SCORE | FILENAME | SUBMISSION DATE | SIZE (BYTES) | STATUS | ✔ | |
|---|---|---|---|---|---|---|---|
| 1 | --- | Task2.zip | 04/10/2024 17:35:36 | 4507 | Failed | | + |
| 2 | --- | Task2.zip | 04/11/2024 02:44:01 | 4508 | Failed | | + |
| 3 | 0.3333333333 | Task2.zip | 04/11/2024 02:55:18 | 55541 | Finished | | + |
| 4 | 0.343298656 | Task2.zip | 04/11/2024 08:03:24 | 55981 | Finished | | + |
| 5 | 0.4364380414 | Task2.zip | 04/24/2024 12:17:19 | 63924 | Finished | | + |
| 6 | 0.3333333333 | Task2.zip | 04/25/2024 11:33:16 | 67173 | Finished | | + |
| 7 | 0.4485333705 | Task2.zip | 04/25/2024 12:02:25 | 67883 | Finished | | + |
| 8 | 0.3948524096 | Task2.zip | 04/25/2024 12:13:04 | 68045 | Finished | | + |
| 9 | 0.1818181818 | Task2.zip | 04/25/2024 12:21:28 | 67982 | Finished | | + |
| 10 | 0.4213677108 | Task2.zip | 04/26/2024 10:51:14 | 73221 | Finished | | + |
| 11 | 0.3628456042 | Task2.zip | 04/26/2024 11:45:40 | 73484 | Finished | | + |
| 12 | 0.4783017741 | Task2.zip | 04/26/2024 14:26:35 | 74523 | Finished | | + |
| 13 | 0.5304696005 | Task2.zip | 04/26/2024 15:05:43 | 75025 | Finished | ✔ | + |

Fig. 5: Screenshot of Competition Scores.

| # | SCORE | FILENAME | SUBMISSION DATE | SIZE (BYTES) | STATUS | ✔ | |
|---|---|---|---|---|---|---|---|
| 1 | --- | Task2.zip | 04/25/2024 13:29:59 | 3698 | Failed | | + |
| 2 | --- | Task2.zip | 04/25/2024 13:35:14 | 3698 | Failed | | + |
| 3 | --- | Task2.zip | 04/25/2024 14:14:29 | 3698 | Failed | | + |
| 4 | 0.2041298571 | Task2.zip | 04/26/2024 09:39:08 | 72536 | Finished | | + |
| 5 | 0.2277339299 | Task2.zip | 04/26/2024 09:48:16 | 72309 | Finished | | + |
| 6 | 0.2199257008 | Task2.zip | 04/26/2024 09:59:10 | 72733 | Finished | | + |
| 7 | 0.4037553233 | Task2.zip | 04/26/2024 10:16:24 | 73026 | Finished | | + |
| 8 | 0.5703588133 | Task2.zip | 04/27/2024 07:20:27 | 78989 | Finished | ✔ | + |
| 9 | 0.4429355281 | Task2.zip | 04/27/2024 07:39:39 | 79259 | Finished | | + |
| 10 | 0.4211538462 | Task2.zip | 04/27/2024 07:48:18 | 79402 | Finished | | + |
| 11 | 0.5669312169 | Task2.zip | 04/27/2024 08:15:57 | 79694 | Finished | | + |

Fig. 6: Screenshot of Competition Scores.

| # | SCORE | FILENAME | SUBMISSION DATE | SIZE (BYTES) | STATUS | ✔ | |
|---|---|---|---|---|---|---|---|
| 1 | 0.5421996042 | Task2.zip | 04/27/2024 08:26:37 | 79804 | Finished | | + |
| 2 | 0.5042012066 | Task2.zip | 04/27/2024 08:49:54 | 81174 | Finished | | + |
| 3 | 0.599668237 | Task2.zip | 04/27/2024 09:03:20 | 81314 | Finished | ✔ | + |
| 4 | --- | Task2.zip | 04/27/2024 09:18:00 | 3242 | Failed | | + |
| 5 | 0.3390781859 | Task2.zip | 04/27/2024 09:20:28 | 81393 | Finished | | + |

Fig. 7: Screenshot of Competition Scores.

| # | SCORE | FILENAME | SUBMISSION DATE | SIZE (BYTES) | STATUS | ✔ | |
|---|---|---|---|---|---|---|---|
| 1 | 0.4869878867 | Task2.zip | 04/27/2024 11:43:37 | 82935 | Finished | | + |
| 2 | 0.5492358033 | Task2.zip | 04/27/2024 11:46:49 | 83156 | Finished | | + |
| 3 | 0.4872598094 | Task2.zip | 04/27/2024 11:52:04 | 83161 | Finished | | + |
| 4 | 0.6148257523 | Task2 (2).zip | 04/27/2024 11:56:33 | 83336 | Finished | ✔ | + |

Fig. 8: Screenshot of Competition Scores.

| # | SCORE | FILENAME | SUBMISSION DATE | SIZE (BYTES) | STATUS | ✔ | |
|---|---|---|---|---|---|---|---|
| 1 | 0.5057355356 | Task2.zip | 04/27/2024 13:59:02 | 86281 | Finished | | + |
| 2 | 0.6000017003 | Task2.zip | 04/27/2024 14:13:09 | 86759 | Finished | | + |
| 3 | 0.5997606603 | Task2.zip | 04/27/2024 14:21:21 | 87155 | Finished | | + |
| 4 | 0.6462091754 | Task2.zip | 04/27/2024 14:27:03 | 87328 | Finished | ✔ | + |
| 5 | 0.6136200717 | Task2.zip | 04/28/2024 04:06:01 | 92457 | Finished | | + |
| 6 | --- | Task2.zip | 04/28/2024 04:07:35 | 2677 | Failed | | + |
| 7 | 0.6246079571 | Task2.zip | 04/28/2024 04:09:52 | 92684 | Finished | | + |
| 8 | --- | Task2.zip | 04/28/2024 04:14:44 | 3258 | Failed | | + |
| 9 | 0.6147157853 | Task2.zip | 04/28/2024 04:18:41 | 92856 | Finished | | + |

Fig. 9: Screenshot of Competition Scores.

Results

| # | User | Entries | Date of Last Entry | Team Name | Kappa ▲ | F1 ▲ | Specificity ▲ | Average ▲ | CPU Time ▲ |
|---|---|---|---|---|---|---|---|---|---|
| 1 | tiandoufayale | 4 | 04/27/24 | | 0.4938 (1) | 0.7004 (1) | 0.7444 (12) | 0.6462 (1) | 0.2503 (22) |
| 2 | Vinylcat | 11 | 04/27/24 | | 0.4395 (2) | 0.6294 (5) | 0.8500 (5) | 0.6397 (2) | 0.2725 (23) |
| 3 | Snorlax | 6 | 04/25/24 | | 0.4029 (6) | 0.5576 (14) | 0.9389 (3) | 0.6331 (3) | 0.5978 (28) |
| 4 | wxydada | 18 | 04/27/24 | | 0.4220 (3) | 0.6154 (7) | 0.8500 (5) | 0.6291 (4) | 0.3016 (24) |
| 5 | PressurePoints | 9 | 04/24/24 | | 0.4152 (5) | 0.6032 (9) | 0.8667 (4) | 0.6283 (5) | 0.1093 (12) |
| 6 | Genshin | 4 | 04/27/24 | | 0.4201 (4) | 0.6355 (3) | 0.7889 (8) | 0.6148 (6) | 0.2496 (21) |
| 7 | a1b2c3 | 5 | 04/27/24 | | 0.3921 (7) | 0.6124 (8) | 0.7944 (7) | 0.5997 (7) | 0.2493 (20) |
| 8 | Ace_Taffy | 11 | 04/27/24 | | 0.3884 (8) | 0.6449 (2) | 0.6778 (18) | 0.5704 (8) | 0.0157 (2) |
| 9 | MeteorVanish | 13 | 04/27/24 | | 0.2743 (16) | 0.4267 (23) | 0.9444 (2) | 0.5485 (9) | 0.5903 (27) |
| 10 | buruyuanshen | 5 | 04/27/24 | | 0.3521 (9) | 0.6230 (6) | 0.6667 (19) | 0.5472 (10) | 0.0170 (3) |

Fig. 10: Screenshot of Competition Rank.

## VIII. 附录

## References


[1] Amélie Beucher, Anders Bjørn Møller, and Mogens Humlekrog Greve. Artificial neural networks and decision tree classification for predicting soil drainage classes in denmark. *Geoderma*, 352:351–359, 2019.

[2] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7291–7299, 2017.

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[5] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.

[6] Guosheng Hu, Yongxin Yang, Dong Yi, Josef Kittler, William Christmas, Stan Z Li, and Timothy Hospedales. When face recognition meets with deep learning: an evaluation of convolutional neural networks for face recognition. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 142–150, 2015.

[7] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[8] Yang Liu, Shuangshuang Zhao, Qianqian Wang, and Quanxue Gao. Learning more distinctive representation by enhanced pca network. *Neurocomputing*, 275:924–931, 2018.

[9] Wanli Ouyang, Xingyu Zeng, Xiaogang Wang, Shi Qiu, Ping Luo, Yonglong Tian, Hongsheng Li, Shuo Yang, Zhe Wang, Hongyang Li, et al. Deepid-net: Object detection with deformable part based convolutional neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(7):1320–1334, 2016.

[10] Sylvestre-Alvise Rebuffi, Sven Gowal, Dan Andrei Calian, Florian Stimberg, Olivia Wiles, and Timothy A Mann. Data augmentation can improve robustness. *Advances in Neural Information Processing Systems*, 34:29935–29948, 2021.

[11] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48, 2019.

[12] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[13] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[14] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

[15] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

[16] David A Van Dyk and Xiao-Li Meng. The art of data augmentation. *Journal of Computational and Graphical Statistics*, 10(1):1–50, 2001.

[17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[18] Bichen Wu, Chenfeng Xu, Xiaoliang Dai, Alvin Wan, Peizhao Zhang, Zhicheng Yan, Masayoshi Tomizuka, Joseph Gonzalez, Kurt Keutzer, and Peter Vajda. Visual transformers: Token-based image representation and processing for computer vision, 2020.

[19] Tete Xiao, Mannat Singh, Eric Mintun, Trevor Darrell, Piotr Dollár, and Ross Girshick. Early convolutions help transformers see better. *Advances in neural information processing systems*, 34:30392–30400, 2021.