湖州师范学院

2023 届毕业设计(论文)

课 题 名 称：基于 YOLO 模型的课堂抬头率检测研究

课 题 名 称（英文）：**A Study of CLass Head Rate Detection Based on YOLO**

学 生 姓 名： 张丁雨 学 号：2019082434

专 业 名 称： 计算机科学与技术

指 导 教 师： 吴茂念 职 称： 教授

所 在 学 院： 信息工程学院

完 成 日 期： 2023 年 3 月 22 日

教务处制表

基于 YOLO 模型的课堂抬头率检测研究

摘要：课堂抬头率是教育工作者用来衡量一节课堂教学质量的重要参数。针对课堂上的学生抬头情况不佳、参与度不高的情况，本项目旨在设计一个能够检测课堂抬头率的系统。从而帮助教育工作者更加客观，精准地获取课堂抬头率的数据，有助于教育工作者下一步的教学安排，该系统对于提升学生的课堂专注度也大有裨益。

目前计算机视觉领域的发展持续可观，卷积神经网络是一类包含卷积计算、具有深层结构的前馈神经网络。凭借目标识别又快又准的优点，YOLOv5 逐渐发展成为一种主流的高效识别的模型。项目通过对于数据集的训练、验证，从而生成了一个能够检测图片，视频和实时的课堂抬头率的模型。并且考虑到实际需求，可以对于单个学生的头部情况进行裁剪和分析。最终得到了一个检测精度高且速度快的课堂抬头率检测系统，在实际应用中可以起到优化教学，加速教育数字化转型的效果。

关键词：YOLO 模型，课堂抬头率，深度学习，卷积神经网络

A Study of Class Head Rate Detection Based on YOLO

Abstract: Classroom head-up rate is an important parameter used by educators to measure the quality of a classroom session. The aim of this project is to design a system that can detect the head-up rate of students in the classroom, in view of the poor head-up rate and low participation of students in the classroom. This will help educators to obtain more objective and accurate data on classroom head-up rates, which will help them in the next step of teaching and learning.

The current development in the field of computer vision continues to be considerable. Convolutional neural networks are a class of feed-forward neural networks that include convolutional computation and have a deep structure. With the advantage of fast and accurate target recognition, YOLOv5 has evolved into a mainstream model for efficient recognition. The project has been trained and validated on a dataset to produce a model that can detect images, videos and classroom heads-up rates in real time. The model can also be tailored and analysed for individual student head profiles, taking into account practical requirements. The result is a highly accurate and fast head-up rate detection system that can be used in practice to optimise teaching and learning and accelerate the digital transformation of education.

Keywords: YOLO model,class head rate,deep learning,convolutional neural networks

目 录

第一章 绪 论

随着计算智能和移动技术的更新迭代，互联网悄然“入侵”课堂。许多学生埋头沉浸在虚拟网络中，与课堂完全脱节，导致教师在课堂上饰演独角戏的结果。作为一项衡量课堂教学质量的指标—课堂抬头率，如若利用计算机技术使得该数据的采集智能化，便捷化，那么对于提高课堂的抬头率，减轻教师的教学负担都大有裨益。

1.1 选题的意义

移动技术在给我们提供巨大便利的同时，困扰也随之而来。“低头族”现象早已司空见惯，“机不离手，眼不离屏”似乎成为现代人的标配。这种现象也延伸至学校课堂中，严重影响教学质量，阻碍学生进步。

2022 年 3 月，我国教育部开办新闻发布会，报告了三年来贯彻落实学校思政理论课教师座谈会精神工作进展成效。针对高校思政课堂“抬头率”数据较低的问题，教育部社会科学司司长在会议中提出，要有针对性地加强教师培训，增加教学方法训练，提升课堂抬头率。其实不仅仅在思政课堂之中，各门学科的课堂中都出现了“课堂抬头率”低的现象。

课堂抬头率指的是课堂中能够抬头认真听讲的学生数量在课堂总学生人数中的占比。通过分析课堂抬头率这项指标，任课老师可以便利高效地完成课堂教学的评价，全面了解教学的过程，为改变教学策略提供依据与支撑。

但由于这项数据具有实时性的特征，单凭教师一人收集全班的课堂抬头情况，不仅加重教师的工作负担，而且数据的准确率也让人难以信服。计算机视觉技术发展势头大好，采用的是计算机模仿人类视觉系统的科学，使得计算机能够拥有类似人类读取、处理、理解、分析图像或是图像序列的能力，在全世界范围内得到了快速的普及应用。本项目借助 YOLO 模型帮助精确检测与计算课堂抬头率数据，此外还进一步实现课堂学生头部姿势的实时检测。这样不仅可以丰富课堂教学评价的手段，还可以优化教学过程。

1.2 研究现状及发展趋势

- 研究现状

在智慧教学的过程中，教育专家与各级教师对于课堂抬头率的关注度愈发提高，研究也逐渐深入。随着计算机技术的不断发展，在教学中我们通常通过搭载传感器，去捕捉收集学生们的课堂行为数据，以便作为评估课堂教学质量的依据。目前，各个领域的学者们还通过计算机视觉领域技术或凭借机器学习模型来实现课堂抬头率的检测。

针对课堂抬头率检测的实际应用，实现人脸识别仅需一个高清摄像头和一台电脑即可完成, 可以通过使用 Python + Open CV 调用百度人脸检测 API 实现人脸识别。Open CV 是开源的计算机视觉库,

而百度的人脸识别接口可以提供人脸检测功能,扫描出人脸的关键信息以及各属性值[1]。孙亚丽则是利用 C++在 Open CV 环境下编写人脸检测程序,检测和统计视频中的人脸。再使用转化的思想将抽象的人脸数据转换成具体的点,根据点聚集情况确定人脸数和人脸区域。选取绝大多数学生的行为意向作为专注意向,每 50 帧检测一次,逐次统计学生有效抬头次数和有效低头次数,将它们之和作为专注次数[2]。孙众团队成员用人脸检测法识别课堂教学视频中 S-T 行为、人体骨架信息提取法识别学生课堂行为。开发的课堂行为分析系统可以通过实现了关键帧提取、学生跟踪、动作识别和动作统计等模块增强识别的效率[3]。学者唐康回顾了计算机视觉人脸检测和人脸识别技术的发展历程，分析和吸取了先进的人脸检测和表情识别方法，结合现实学校的课堂教学场景，提出了一种基于深度学习的高效、精确的人脸检测方法。在人脸检测的基础上，提出了一种基于朴素贝叶斯分类的表情识别和评分方法，对人脸情绪进行正负面的分类及评分[4]。为了提取更多的图像特征，钱铠伦的团队将检测得到的头部区域候选窗口使用边界框归和非极大值抑制（NMS）算法对重叠的候选窗口进行剔除，并将剩余的候选窗口输入至下一个精细检测 CNN 网络中[5]。陈玥和李乐乐等人利用卷积神经网络图像处理技术捕捉学生的隐性消极课堂行为，将行为心理学和图像识别技术相结合，由人工智能算法准确判断学生课堂行为[6]。

巢渊与其团队成员针对室内应用中检测人脸角度不同、光照变化、部分遮挡、模糊等复杂工况，提出一种基于改进 YOLO-v4 的室内人脸快速检测方法。该系统基于深度可分离残差网络结构进行改进，提升模型检测效率；在构建特征金字塔过程中引入注意力机制，自适应调整通道特征与空间特征权重，提升模型特征提取能力[7]。为实现复杂场景下端对端实时视频监控异常目标检测与定位,胡正平等人借鉴目标检测思路,提出端对端 SSD 实时视频监控异常目标检测与定位算法[8]。邵延华的团队根据 YOLO 的模型结构、损失函数、交并比等的改进，对 YOLO 系列的几个重要版本进行了详细的分析与总结。并对经注意力、轻量化等方式改进后的 YOLO 算法进行了对比分析，体现了其被广泛使用的原因[9]。

范鹏飞选择通过主流算法对比实验后选择效果最佳的 Yolo-v3 算法作为检测工具对数据集进行训练测试,并将得到的网络模型进行优化。为了更加方便直观展示抬头率的统计结果,将历史和当前统计信息保存并通过可视化的系统模块展示出来,为高校的课堂抬头率的评价提供了参考,也为高校评价学生和教师开辟出一种新的方法。但是此系统只考虑到了讲授型课堂的抬头率而忽视了研讨型课堂抬头率的检测，存在实际应用的不足[10]。郭敏钢和宫鹤提出了一种新的激活函数 ReLU-Swish。将 ReLU 激活函数和 Swish 激活函数的正负 x 半轴进行分段组合，使 Swish 激活函数 x 正半轴成为线性函数。在 CIFAR-10 和 MNIST 数据集上的实验结果表明，ReLU-Swish 激活函数的准确率及收敛速度上相比 Swish 激活函数表现更优秀，有效地缓解了“坏死”现象以及收敛速度过慢的问题，起到了优化卷积神经网络的作用。该 ReLU-Swish 激活函数在收敛性及准确率上都有所提高，在模型训练中表现的更优异，达到了很好的预期效果，同时也为卷积神经网络的研究提供了一种新的方法及思路[11]。Kyrkou Christos 解决了智能相机应用中基于深度学习的高效行人检测在精度和速度之间取得良好折中的挑战，创造一种基于可分离卷积的计算高效架构，集成了跨层密集连接和多尺度特征融合，在减少参数和操作数量的同时提高了表示能力；分析了一种更加精细的损失函数用于改进定位；以及一种无锚的检测方法[12]。

- 发展趋势

在知识经济时代，教育是实现人生价值和社会价值的唯一途径。提高课堂抬头率进而保证课堂教

学质量，对于去培养新时代的人才起到了非常重要的地位和作用。课堂抬头率这项数据贯穿在课堂教学的始终，贯穿在每个人学习的始终。因为这项数据有着极大的研究潜力，在检阅查找这方面信息，为文章打基础的阶段，我看到了许多学者对于课堂抬头率这个概念都有着深刻的理解。他们往往从现状，归因，策略和影响几方面展开讨论，关于课堂抬头率的研究也越来越深入，越来越细化。在当今的大数据时代，可以说谁掌握了越详细越全面的信息，谁就占得了先机，谁就有可能去成为业界的泰斗。在关于检测课堂抬头率这种项目，早就有学者开始并且持续研究。未来也会有越来越多的学者投身于目标检测这个领域并且大展宏图。

关于 YOLO 系统，它是公认的目标检测领域性能优秀的算法之一。自 YOLOv1 现世以来，YOLO 模型经久不衰，并且保持着一年更新一个版本的速度持续成长。YOLO 模型作为一种新颖的卷积神经网络，既继承了卷积神经网络的结构特点，又有自己独特之处。从个人学习角度来看，一位优秀的计算机视觉工程师，对于目标检测的学习避免不了，而目标检测的核心就是 YOLO 模型。当然 YOLO 系列模型也一直在更新发展，对于它的学习与研究迫在眉睫。从个人职业前景来看：YOLO 一直并且未来都会是应用面很广泛的主流算法，不仅是合格工程师的标配，更是技术进步的基础储备。

1.3 研究内容

本项目研究方向灵感来源于大二时期学院要求在每节课堂开始之前同学们能够主动将手机上交，进而改善课堂抬头情况。因此在考虑论文方向时，打算做一个能够实时检测听课学生抬头率的项目，以便帮助教学工作者能够更加高效地获取每节课堂学生的抬头情况。研究内容如下：

- 使用 YOLOv5 模型初步实现课堂抬头率检测的功能，最终得到实时的课堂抬头率数据；
- 通过比较与研究网络架能够实现模型性能的提升与改进，思考如何通过改变参数从而达到提高精确度的效果；
- 如何使得项目更加便利用户，提高使用满意度；怎样促进计算机技术与教学的全面融合，实现信息化教学。

第二章 主干特征提取网络

2.1 ConvNeXt 模型

2020 年，Google 团队提出将 Transformer 加入图像分类领域并且实际应用，结果超乎意料，性能甚至超过了众多存在已久的卷积神经网络。越来越多的研究学者开始拥入了 Transformer 的怀抱，次年各大平台发表的关于计算机视觉的文章基本上是基于 Transformer 模型的。但在大家认为卷积神经网络就要淡出计算机视觉领域的舞台中央时，2022 年的开年一篇由 Facebook AI Research 和 Berkeley 共同发表的文章 A ConvNet for the 2020s 瞬间打破了这一局面。在文章中学者们创造性地提出 ConvNext 这个纯卷积神经网络。在作者的一系列训练比照之后，在相同的环境下，ConvNext 无论在准确率方面，还是在推理速度方面，性能明显优于 Swin Transformer。在表 2-1 中，甚至在 ImageNet 22k 上 ConvNet-XL 的准确率高达 87.8%。

表 2-1 ConvNext 与 Swin 模型训练差异表

模型 model	图片尺寸 image size	每秒峰值速度 FLOPs	吞吐量 throughput	分类精度 (1K/22K)
Swin-T	224^2	4.5G	1325,6	81.3/-
ConvNeXt-T	224^2	4.5G	**1943.5**(+47%)	**82.1**/-
Swin-S	224^2	8.7G	857.3	83.0/-
ConvNeXt-S	224^2	8.7G	**1275.3**(+49%)	**83.1**/-
Swin-B	224^2	15.4G	662.98	83.5/85.2
ConvNeXt-B	224^2	15.4G	**969.0**(+46%)	**83.8/85.8**
Swin-B	384^2	47.1G	242.5	84.5/86.4
ConvNeXt-B	384^2	45.0G	**336.6**(+39%)	**85.1/86.8**
Swin-L	224^2	34.5G	435.9	-/86.3
ConvNeXt-L	224^2	34.4G	**611.5**(+40%)	**84.3/86.6**
Swin-L	384^2	103.9G	157.9	-/87.3
ConvNeXt-L	384^2	101.0G	**211.4**(+34%)	85.5/**87.5**
ConvNeXt-XL	224^2	60.9G	**424.4**	-/**87.0**
ConvNeXt-XL	384^2	179.0G	**147.4**	-/**87.8**

ConvNext 这一网络结构的出现，说明并不一定需要一个全新结构的出现，只需要对原有的卷积神经网络的技术与参数进行修改优化也能达到最高效能。ConvNet 结构首先从标准神经网络出发，依次从宏观设计，深度可分离卷积，逆瓶颈层，大卷积核，细节设计这五个角度借鉴 Swin Transformer 模型的概念，最后在 ImageNet-1K 上进行训练和评估，从而得到 ConxNext 的核心结构。

- Macro design(宏观设计)

变换阶段计算比率：在 ResNet-50 网络结构中，一共有四个网络块，每个网络块都有不同数量的网络层，堆叠 block（用于构建网络的基本单元）次数最多的一般都是第三个网络块。由表 2-2 可知，stage1 到 stage4 堆叠 block 的次数是(3，4，6，3)。而在 Swin Transformer 中，Swin-T 模型的比例是 1:1:3:1，对于结构较大的 Swin-L 模型的比例是 1:1:9:1。由此可见，在 Swin Transformer 中，stage3 堆叠 block 的占比是比 ResNet50 更高。所以作者将 Swin-T 模型的参数沿用至 ConvNeXt 结构，将堆叠次数改为(3，3，9，3)。在进行了这样的调整后，准确率由 78.8%提升到了 79.4%[13]。

表 2-2 ResNet-50 网络结构图

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		[3×3, 64; 3×3, 64] ×2	[3×3, 64; 3×3, 64] ×3	[1×1, 64; 3×3, 64; 1×1, 256] ×3	[1×1, 64; 3×3, 64; 1×1, 256] ×3	[1×1, 64; 3×3, 64; 1×1, 256] ×3
conv3_x	28×28	[3×3, 128; 3×3, 128] ×2	[3×3, 128; 3×3, 128] ×4	[1×1, 128; 3×3, 128; 1×1, 512] ×4	[1×1, 128; 3×3, 128; 1×1, 512] ×4	[1×1, 128; 3×3, 128; 1×1, 512] ×8
conv4_x	14×14	[3×3, 256; 3×3, 256] ×2	[3×3, 256; 3×3, 256] ×6	[1×1, 256; 3×3, 256; 1×1, 1024] ×6	[1×1, 256; 3×3, 256; 1×1, 1024] ×23	[1×1, 256; 3×3, 256; 1×1, 1024] ×36
conv5_x	7×7	[3×3, 512; 3×3, 512] ×2	[3×3, 512; 3×3, 512] ×3	[1×1, 512; 3×3, 512; 1×1, 2048] ×3	[1×1, 512; 3×3, 512; 1×1, 2048] ×3	[1×1, 512; 3×3, 512; 1×1, 2048] ×3
	1×1	average pool, 1000-d fc, softmax				

将 stem 结构换成“patchify”策略：通常情况下，stem 往往在开始输入时来处理图片。由于输入的原始图片的复杂冗余的特点，stem 会在结构中积极地将输入的图片进行下采样操作到适合结构大小的特征图。在网络中，最初的下采样模块 stem 大部分都包括了步距为 2，卷积核大小为 7x7 的卷积层和一个步距为 2 的最大池化下采样。而 Transformer 模型选择的是“patchify”策略，即在一个卷积核很大且邻近窗口没有重复的卷积层来进行下采样。所以作者将模型中的 stem 模块换成了 Swin Transformer 的 patchify。这样使得准确率提升了 0.1%，并且 GFLOPs 也有所降低[13]。

- ResNeXt-ify（深度可分离卷积）

由于 ResNeXt 在每秒峰值速度和精准度方面的表现比普通的 ResNet 更优秀，于是作者进行了一些借鉴操作。使用分组卷积是 ResNeXt 网络的一大特色，于是作者将瓶颈块中 3×3 卷积替换成了分组卷积。这个操作将 GFLOPs 从 4.4 降到了 2.4，与此同时，网络的宽度得到了扩展。这也就是 ResNeXt 奉行的原则是“划分成更多的组，以扩大宽度”。根据这一指导原则，作者将网络宽度增加到了与 Swin-T 模型一样的通道数。这一操作将 GFLOPs 增加了 5.3，准确率则是提升到了 80.5%。

- Inverted Bottleneck(逆瓶颈层)

如图 2-1(a)所示，在标准 ResNet 结构中使用的瓶颈层的维度形式是大一小一大，这样安排的目的是减小计算量。后来在 MobileNetV2 结构中使用到的是逆瓶颈结构,该结构如图 2-1(b)所示的大小维度位置，采用小一大一小这样的维度形式。学者们查阅的材料显示这样的维度形式有助于信息在转换不同维度的特征空间时降低压缩维度带来的信息损失，后来在 Transformer 块中也沿用了这样的结构，即中间粗两头细，使得 MLP 块隐藏维度是输入维度的 4 倍。因此作者在 ConvNeXt 结构中借鉴了小一大一小这种逆瓶颈结构，结果显示在较小的模型上准确率有小幅度的提升，在较大的模型上准确率的提升幅度远大于较小的模型。由此可见，在较大的模型上使用逆瓶颈结构能够取得更为理想的效果。

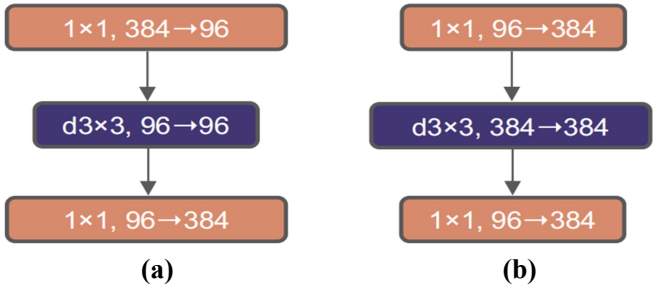

图 2-1 不同模型的 bottleneck 模式图

- Large Kernel Sizes（大卷积核）

在这个模块作者进行了两个方面的改动：首先是 Moving up depthwise conv layer，也就是说将深层卷积层的位置上移,具体上移变化如表 2-2 所示。

表 2-3 深层卷积层上移变化表

原来	1x1 conv	depthwise conv	1x1 conv
现在	depthwise conv	1x1 conv	1x1 conv

这一改动借鉴了 transformer 结构中 MSA 模块与 MLP 模块的放置位置。在这一改变下，准确率下降到 79.9％。

第二个操作是 Increasing the Kernel size，即将原本大小为 3×3 卷积核增大至了 7×7。在获取到 7 这个数字之前，作者也反复实验并且发现，内核大小取到 7 时准确率达到了一个饱和点[15]。到这步，准确率由 3×3 时候的 79.9%达到了 7×7 时候的 80.6%[13]。

- Micro Design（微观设计）

Replacing ReLU with GELU。近年来，在比较先进的 Transformer 中大部分使用的激活函数都是 GELU，它的函数图像比 ReLU 的函数图像更为平滑。于是作者又将原本简单高效的 ReLU 激活函数替换成了 GELU，检测到模型的性能没有发生明显的变化。

Fewer activation functions。在卷积神经网络中，每个卷积层都会调用一个激活函数。但在 Transformer 结构中并非如此。于是作者在 ConvNeXt Block 中又一次借鉴了 Transformer，降低激活函数的使用频次，接着作者发现准确率出乎意料地得到了增长。

Fewer normalization layers。与上面的情况类似，Transformer 中 Normalization 使用的频率也比较少。于是作者复制操作，减少了 ConvNeXt Block 中的归一化层，只保留了 depthwise conv 后的归一化层。此时准确率高达 81.4%。

Substituting BN with LN。BN 在卷积神经网络中是较为常见的操作，它的作用是加快模型的收敛速度，减少过拟合。但由于 BN 的复杂性，模型可能会受到影响。而且在 Transformer 中使用的是更简单的 LN，接着作者将 BN 全部替换成了 LN，发现准确率还有小幅提升达到了 81.5%。

综上所述，在 ConvNeXt 结构中，它的优化策略大部分借鉴了 Swin Transformer。它属于一个卷积神经网络，但其简单高效性可与现今较先进的 transformer 模型进行比较。在挖掘研究时，这个模型本身也不是一个新型产物。这一现象值得思考，一定要最大化利用好现有的资源，不仅要考虑开拓创新，还要顾忌回头再钻。

2.2 Swin Transformer

Swin Transformer 结构是微软研究院在 2021 年在国际计算机视觉大会呈现的一篇文章，并且迅速斩获同年 best paper 的荣誉称号。Swin Transformer 网络的出现使得视觉领域又添一员猛将。不管是物体分类，目标检测模型，还是语义分割模型，在使用频率榜单中该结构位列前几，这也使得学者们不得不继续学习，紧跟计算机领域的变化。如图 2-2(a)所示，Swin Transformer 使用的方法与卷积神经网络中的层次化构建方法类似，这样构建结构的优势在于：结构可以应用于分类，分割，检测等一系列任务。

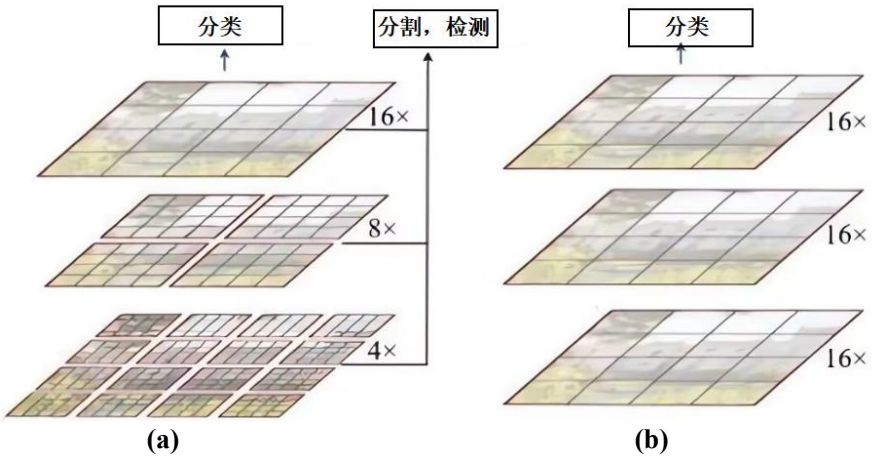

图 2-2 Swin Transformer 模型与 ViT 模型的建构方法对比图

如图 2-2 所示，可以清晰地看出这两个模型都使用了窗口的概念，但是两者在下采样时使用的方式有所区别。在四倍采样和八倍采样中，特征图被划分成了很多个不相交的区域，并且多头注意力机制操作只在每个窗口中进行[13]。这样的做法可以减少模型的计算量，特别是在特征图很大的时候效果尤其明显。如图 2-2(b)所示，ViT 模型选择一直进行 16 倍下采样，尺寸一直未变。对整个特征图进行多头注意力机制操作，比较普通的计算机内核根本不可能承担如此大的计算量。

如图 2-3 所示，作者呈现了 Swin Transformer 网络的架构图，以便解释该框架工作的基本流程[13]。

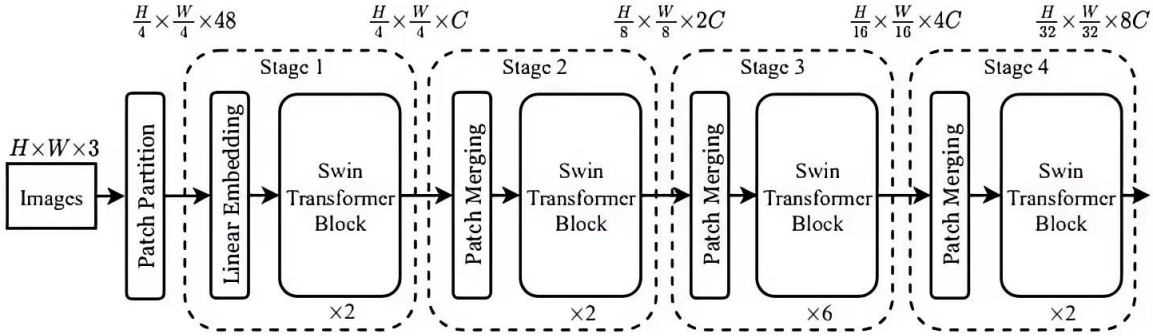

图 2-3 Swin Transformer 网络架构图

- 第一步是将图片传送到 Patch Partition 模块中实现分块操作，每 4x4 相邻的像素为一个 Patch。如果输入的是一个 RGB 三通道图片，因为“每 4x4 相邻近的像素为一个 Patch”这一规律，那么每个 patch 就有 4x4=16 个像素，然后又因为每个像素有 R、G、B 三个值，所以展平后每个 Patch 的尺寸为 16x3=48，所以通过该模块后图像由原来的 [H, W, 3]变成了 [H/4, W/4, 48][14]。
- 通过 Linear Embedding 层会对每个像素的 channel 数据做线性变换，这是一个全连接层，刚才特征图通过这一个全连接层后 48 会被映射为 C，即图像由 [H/4, W/4, 48]变成了 [H/4, W/4, C]。这里以上的两个模块是通过一个卷积层实现效果的。
- 接着是通过四个 Stage 操作来构建大小不同的特征图。除了 Stage1 中是通过一个 Linear Embedding 层外，剩下三个 stage 都会先通过一个 Patch Merging 层来进行下采样，然后都是重复堆叠 Swin Transformer Block。在后三个 stage 中的 Patch Merging 层中，我们会对特征图进行分辨率减半，通道数翻倍的操作，则输出的图像会变成 [H/8, W/8, 2C][14]。
- 最后对于分类网络来说，还会添加一个 Layer Norm 层、全局池化层以及全连接层，得到最终输出

结果。

从流程图可以看出，Swin Transformer 网络中有两个关键构建块：Patch Partition 和 Swin Transformer Block。这也是作者在论文中详细讲解了这两大板块。

- Patch Partition：如图 2-4 所示，可知通过操作这个模块可以使得特征图分辨率减半，通道数翻倍。如果将一个 4x4 大小的单通道特征图输入 Patch Merging，Patch Merging 会先分割为四个小的特征图，然后在每行每列隔一个取出一个小 Patch 组合在一起，这样就得到了 4 个 2x2 的特征图。接着将这四个特征图在深度方向进行拼接，则现在的特征图尺寸为 2x2x4，然后会通过一个 LN 层。最后一步是通过一个全连接层，将特征图的深度减半。

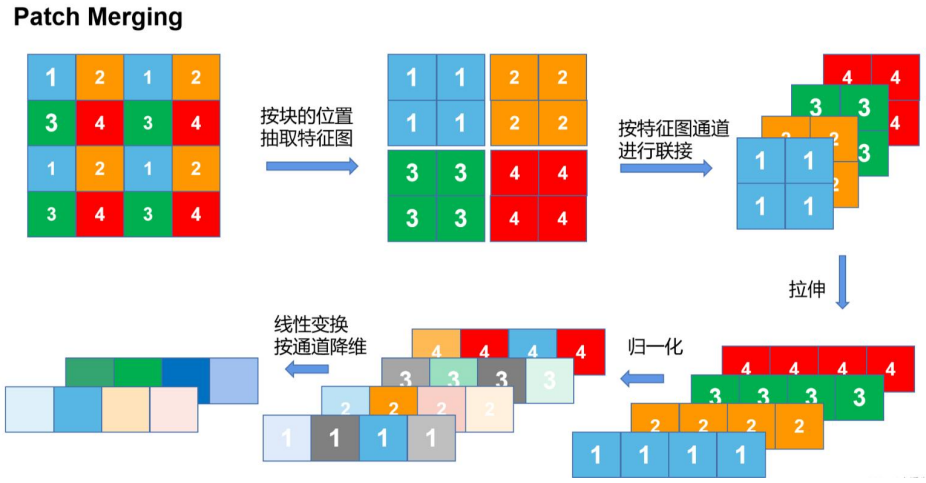

图 2-4 Patch Merging 原理图

- Swin Tranformer Block：Swin Transformer 使用 Window MSA (W-MSA) 和 Shifted Window MSA (SW-MSA) 代替了 ViT 所使用的多头自注意模块。此处的 Block 包含了两种结构，如图 2-5 所示，这两种结构的唯一区别在于使用的结构完全不同。这两个结构是成对使用的，因而堆叠 Swin Transformer Block 的次数一般都为双数。

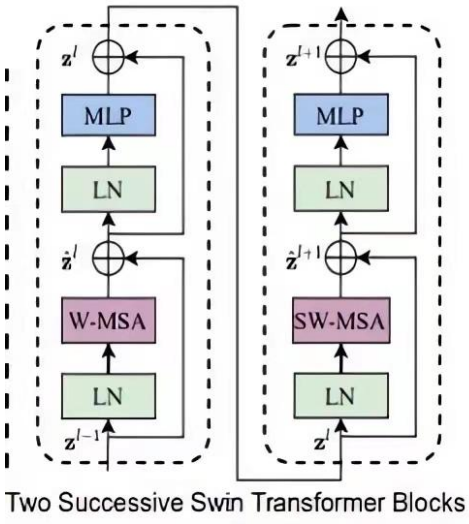

图 2-5 Swin Tranformer Block 两种结构图

2.3 CSPDarkNet

这是 YOLOv5 模型的一个主干特征提取网络，在 input 端预处理完的图片会在这里先进行特征提取，提取到的特征就叫做特征层。特征层这一概念，简而言之，也就是需要进行处理的图片的一个特征集合。在这一部分，在提取了三个特征层之后，下一步要进行的操作是网络的构建。这个主干特征提取网络具有如下几个特点:

- 使用了残差网络 Residual[15]。该网络的主要功能是帮助模型解决层数增加时出现梯度消失或者爆炸问题。backbone 中的残差卷积可以分为主干和残差边这两个部分，主干部分包含了一个 1X1 的卷积层和一个 3X3 的卷积层。通过前一个 1X1 的卷积层可以来减少参数量，通过后一个 3×3 的卷积层可以恢复到原来的大小。残差边部分不会进行任何的处理，而是直接将输入与输出相结合。YOLOv5 模型的主干部分都由残差卷积构成，该网络优点是易于优化，能够通过增加深度的方式来提高准确率。残差网络内部的成员残差块的连接方式是跳跃连接，这种方式也是残差网络能够解决梯度消失问题的关键所在。
- 使用了 CSPnet 网络结构。如图 2-6 所示，CSPnet 的任务就是拆分原本堆叠起来的残差块成为左右两部分，尽可能地降低计算量和提升卷积神经网络的学习能力。主干部分继续进行原来的残差块的堆叠，向后传播；另一部分则经过少量处理直接连接到最后进行合并，实现更为丰富的组合。它不仅仅只是一个网络结构，而且还潜藏着一种可移植的思想方法，可以与多种网络结合。

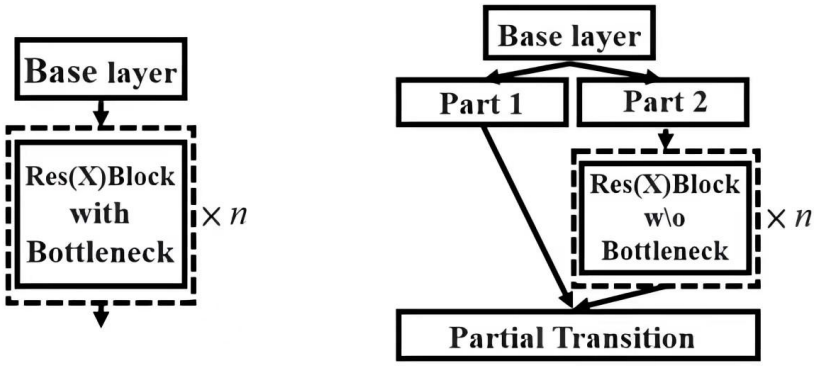

图 2-6 CSPnet 工作原理图

- 使用了 Focus 网络结构。该网络结构是想要让在图片进入 backbone 之前对其进行一个切片的操作。如图 2-7 所示，在图片中的每行一个隔一个取相同值，这与邻近下采样操作类似，重复此操作就可以形成了四个独立的特征层[15]。这四个特征层是互补的，外观类似，将这四个特征层进行堆叠操作，那么 W，H 信息就会集中到了通道空间，输入通道也由此扩充了四倍。那么相对于原先的 RGB 三通道模式会拓宽为十二个通道。接着再形成的新特征层经过卷积操作，最终可以得到没有信息丢失的下采样特征层。

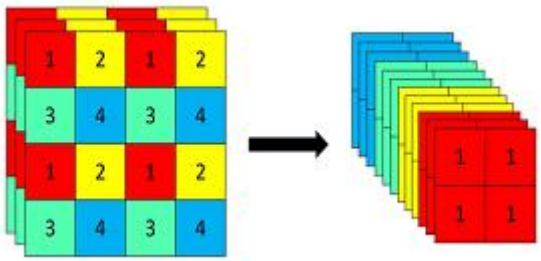

图 2-7 Focus 网络工作效果图

- 调用 SiLU 激活函数。该函数是 Sigmoid 和 ReLU 这两个函数的结合升级版。ReLu 激活函数弥补了 Sgmoid 激活函数的缺点：可以解决在正区间梯度消失问题，收敛速度也比 Sigmoid 函数更快。如图 2-8 所示，SiLU 具备无上界有下界的特点和平滑、非单调的特性，这样的特点使得模型可以收敛更迅速，而且更加稳定。

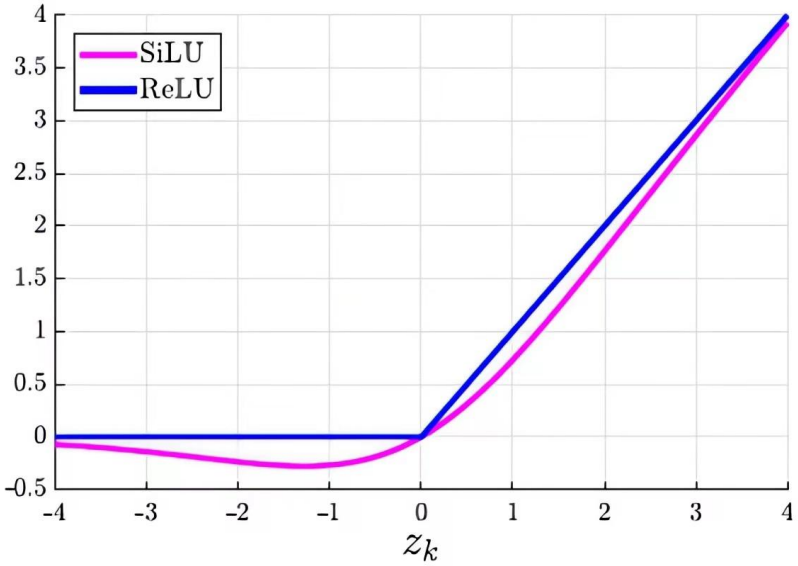

图 2-8 SiLU 激活函数图

- 使用了 SPP 结构。SSP 结构又可以叫做是空间金字塔池化，可以将任何大小的特征层输出成为固定大小的特征向量。并且通过不同大小的池化核的池化操作进行特征提取，来提高网络的感受野。在 YOLOv4 中，SPP 结构仅用在 FPN 里面的，在 YOLOv5 中，SPP 模块则被延伸到了 CSPDarkNet 中。

2.4 网络比较

评价一个目标检测系统有两个指标，分别为精度评价指标和速度评价指标。为了得到一个性能更强的系统，通常优先考虑特征提取能力较强的主干特征提取网络，其次结构的大小也要适中。结构过大的话，会影响目标检测的速度；结构也不能过小，否则特征提取效果会不尽如人意。在对比了以上三种网络结构，并且考虑到电脑显存的局限性，最后选用的是 CSPDarknet 网络。并用该网络进行训练验证，表现与预期效果一致。

第三章 YOLOv5 目标检测模型

3.1 YOLOv5 模型简介

YOLOv5 模型是由 glenn-jocher 在 2020 年首次提出的。迄今为止，该系统沿用至今并且仍然还在更新迭代。如图 3-1 所示，YOLOv5 有 YOLOv5n、YOLOv5s、YOLOv5m、YOLOv5l、YOLOv5x 这五个最常用的版本。由下图可知模型的结构基本一致，最大的差异在于模型深度和模型宽度这两个参数。

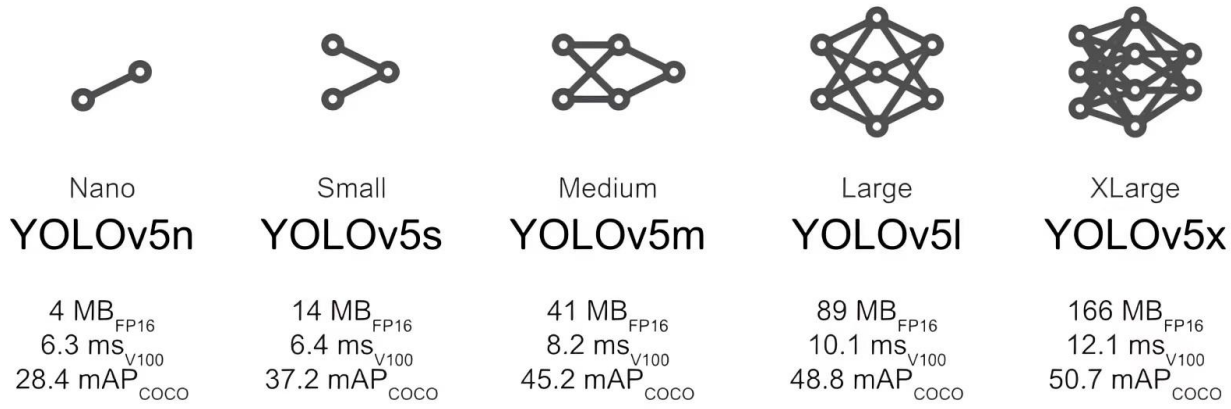

图 3-1 YOLOv5 模型版本差异图

3.2 YOLOv5 模型的特色

YOLOv5 模型属于一种单阶段目标检测算法。YOLOv5 在 YOLOv4 模型基础上做了一些改进措施，使得整个模型在速度与精度方面都得到了极大地提升。主要的改进点如下：

- input：模型通过增强 Mosaic 数据从而提升模型的训练速度和预测精度，还增加了自适应锚框计算和自适应图片缩放的一系列方法。
- backbone：模型在主干网络中引入了 Focus 结构和 CSPnet 结构，来增强网络的学习能力。
- neck：使用 FPN+PAN 结构，使得网络捕捉的特征信息更加全面可靠。
- head：改进训练时的损失函数。
- 模型使用的是 Pytorch 框架，更加方便的训练数据集。模型也能更加简单地投入各领域的生产创造。
- 模型能够直接对图片，视频甚至摄像头等端口的输入都能进行有效预测推理。
- 代码简单易读，融合了大量的计算机视觉技术，有利于学习和借鉴。

3.3 YOLOv5 系统总体框架

如图 3-2 所示，整个 YOLOv5 模型大致包括四大部分，分别为 input，backbone，neck 以及 head。

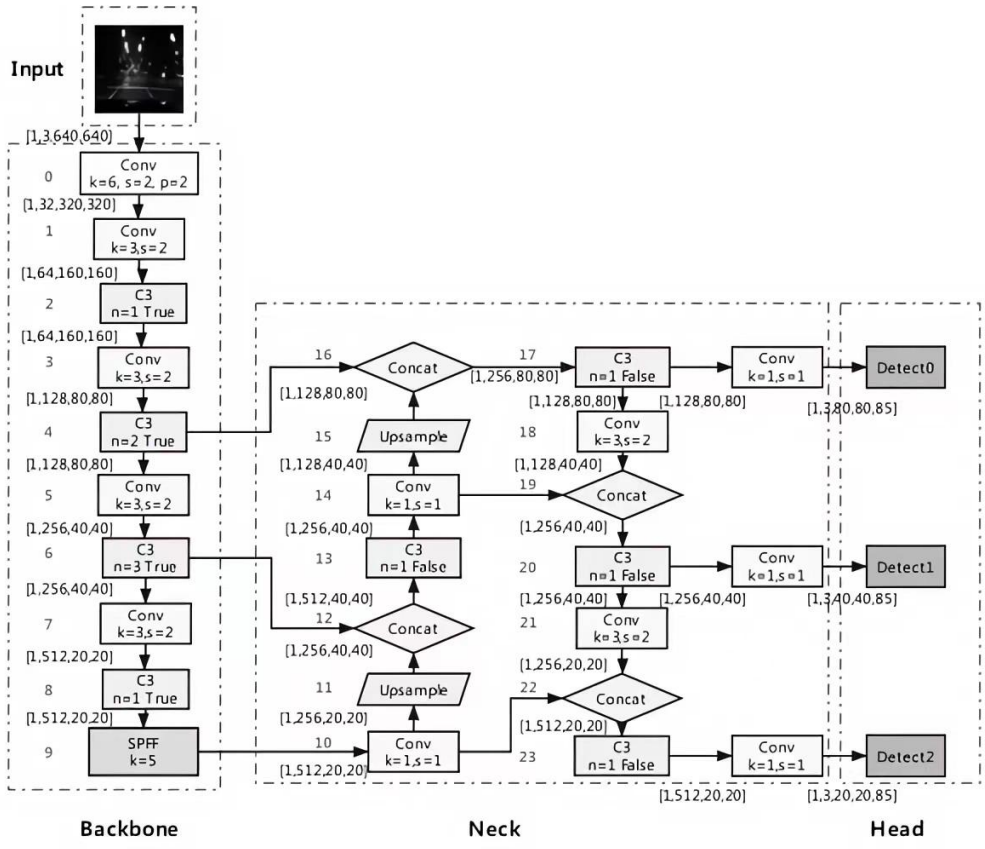

图 3-2 系统总体框架图

- Input

  在输入端对于想要进行检测的图片进行一个预先处理。预先处理的具体操作就是将输入的图片的大小缩放为符合网络大小的图片，并且进行归一化等操作。

- Backbone

  详细讲解可见 2.3 CSPDarkNet。

- Neck

  如图 3-3 所示，YOLOv5 模型的 neck 网络采用的是 PANet 的 FPN+PAN 结构（自顶向下+自底向上）[16]。FPN 是 YOLOv5 模型中的一个加强特征提取网络，目的在于融合高层特征与底层特征的特征信息。在 FPN 部分，已经获得的有效特征层会被用于继续提取特征。它是自顶向下的一个特征金字塔，传递高层的强语义特征，来增强整个金字塔。这个结构只实现了语义信息的增强，忽视了定位信息的加强[17]。那么增加 PAN 就是要去解决这一问题，这样的一个自底向上的金字塔可以对 FPN 起到补充作用，将底层的定位特征传递上去。这样使用 FPN+PAN 两种结构形成的金字塔提取的特征既具有丰富的语义信息又拥有精准的定位信息。

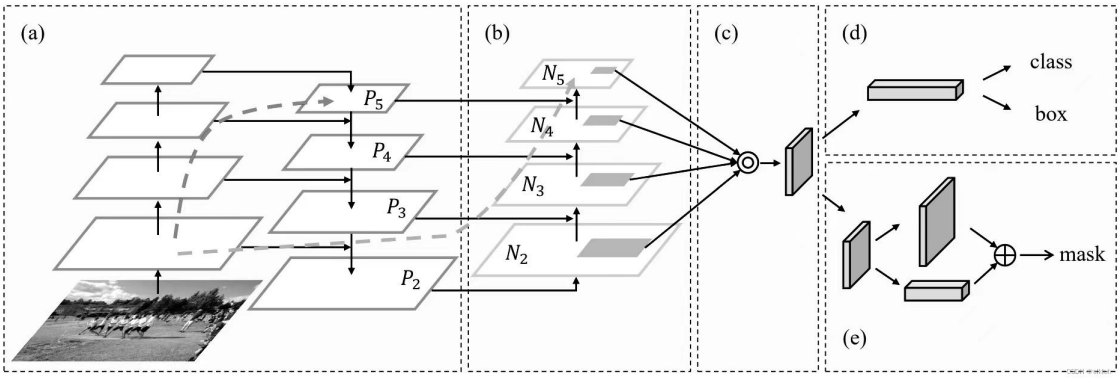

图 3-3 PANet 原理图

- Head

  这是 YOLOv5 模型的分类器与回归器。进入了 backbone 与 neck 层，我们就有了三个加强的有效特征层。每一个特征层都有其宽、高和通道数，又因为特征图可以看作特征点的集合，那么每一个特征点都有其通道数个特征。head 的主要工作的检测目标，对输入物体的特征点进行判断，判断训练出来的特征点是否与物体中的特征点一致。

3.4 YOLOv5 模型的总结

如图 3-4 所示，整个 YOLOv5 网络的一个工作流程就是：输入－特征提取－特征加强－预测特征点对应的物体情况。

图 3-4 YOLOv5 网络操作流程图

第四章 课堂抬头率检测系统的构建

4.1 项目克隆和环境配置

由于 YOLOv5 模型在各大网站都是开源，在已经学习完模型框架原理的基础上，可以在网站上去克隆其源码。在反复比较代码后，最后在 github 这个比较成熟的平台上去下载 YOLOv5 的代码，并使用 Pycharm 打开。

接下来就是环境的搭建工作：

- 首先在确保电脑已经下载了 python 的基础上去浏览器官网下载对应了 python 版本的 Anaconda，在下载的过程中注意权限以及环境变量的自动添加。
- 完成之后打开 Anaconda，使用 conda env list 指令可以查看环境，可以发现新安装的 Anaconda 只有一个 base 环境。接着可以在这个 base 环境中去创建一个新的小环境，然后在小环境中去安装系统运行所需要的包，可以使用 conda creat -n pytorch python=3.7 这行语句去创建一个名字叫做 pytorch，python 版本为 3.7 的虚拟环境。
- 在安装包这个步骤完成之后，我们需要用到 activate pytorch 语句来激活新环境。
- 接下来的步骤是安装模块。在安装模块之前，最好先更换 pip 源为豆瓣源或阿里源，然后安装 YOLOv5 模型需要的模块，工作路径要在 YOLOv5 文件夹下打开，在命令框中输入 python -m pip install r requirements.txt。
- 如果没有安装 cuda 可以安装 pytorch-cpu 版，如果有电脑带有大容量的 gpu，那么优先安装 gpu 版本。

4.2 课堂抬头率检测系统的预训练

YOLOv5 已经在 COCO 数据集上训练好，COCO 数据集一共有 80 个类别，如果项目需要的类别在类别中，可以直接用训练好的模型进行检测。由于本项目主要检测的是人脸的状态，如图 4-1 所示，在 COCO 数据集中可以找到[person]这个类别，那么就可以先将图中的人的大类检测出来。

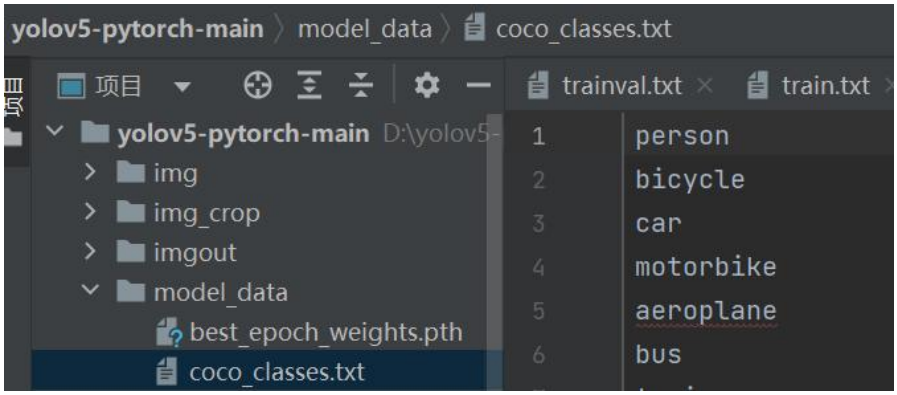

图 4-1 COCO 数据集

接着用预训练模型来进行测试。预训练可以缩减网络的训练时间，还有助于高精度效果的获取。YOLOv5 自开发出来就给大众提供部分供参考的预训练权重，根据项目要求可以去选择不同的预训练权重。如表 4-1 所示，可以了解权重的名字和大小信息，再加上预训练权重越与训练出来的精度成正比，与检测的速度成反比这一规律，本系统最后选择用 YOLOv5s 模型来预训练自己的数据集。

表 4-1 YOLOv5 模型版本差异表

Model	size (pixels)	mAP^{val} 0.5:0.95	mAP^{val} 0.5	Speed CPU b1 (ms)	Speed V100 b1 (ms)	Speed V100 b32 (ms)	params (M)	FLOPs @640 (B)
YOLOv5n	640	28.0	45.7	**45**	**6.3**	**0.6**	**1.9**	**4.5**
YOLOv5s	640	37.4	56.8	98	6.4	0.9	7.2	16.5
YOLOv5m	640	45.4	64.1	224	8.2	1.7	21.2	49.0
YOLOv5l	640	49.0	67.3	430	10.1	2.7	46.5	109.1
YOLOv5x	640	50.7	68.9	766	12.1	4.8	86.7	205.7
YOLOv5n6	1280	36.0	54.4	153	8.1	2.1	3.2	4.6
YOLOv5s6	1280	44.8	63.7	385	8.2	3.6	12.6	16.8
YOLOv5m6	1280	51.3	69.3	887	11.1	6.8	35.7	50.0
YOLOv5l6	1280	53.7	71.3	1784	15.8	10.5	76.8	111.4
YOLOv5x6 + TTA	1280 1536	55.0 **55.8**	72.7 **72.7**	3136 -	26.2 -	19.4 -	140.7 -	209.8 -

可以将要检测的数据放在'D:\yolov5-pytorch-main\VOCdevkit\VOC2007\JPEGImages'路径下，然后输入 python detect.py --source example yolov5-pytorch-main\VOCdevkit\VOC2007\JPEGImages.jpg --weights weights/yolov5s.pt --conf-thres 0.25，一般没有报错就说明图像检测成功。此外，通过修改命令行我们还可以检测视频。在以上步骤完成后，需要检测文件参数说明，YOLOv5 的参数是由 argparse 包传入的，可以通过命令行传入参数，也可以直接设置参数默认值。其中，weights 参数是训练好的权重文件，source 参数为检测数据路径，img-size 参数为检测时图像大小，conf-thres 为检测置信度阈值，iou-thres 是 NMS 的 IOU 阈值。

4.3 课堂抬头的数据集的整理与训练

在搜索各大数据集网站的过程中，发现还没有专门关于学生课堂抬头的数据集，于是使用了爬虫技术对于带有关键词的照片进行爬取收集。爬虫技术是做从网页上抓取数据信息，按照要求清晰数据并保存的自动化程序，其原理就是模拟浏览器发送网络请求，接受请求响应，然后按照一定的规则自动抓取互联网数据。在获得的文件夹中去筛选更符合要求的数据集。爬虫得到的数据集不像直接下载得到的数据集一样已经标注好了特征，需要利用图形图像注释工具 LabelImg 来手动标注数据集。本项目将处于课堂中的学生分为两类，分别为 taitou 类和 ditou 类。如图 4-2 所示，标注好的注释默认以 PASCAL VOC 格式保存在 XML 文件中，在此项目中我们需要手动改为 YOLO 格式。鉴于收集到的数据集较少，因为模型只划分了训练集和验证集。在准备数据时，一定要注意标签中的文件名一定要与图片相对应，否则训练就进行不下去了。

图 4-2 LabelImg 图形图像注释工具

下面需要修改 YOLOv5 的配置文件，需要修改的配置文件有两个。修改 coco.yaml 中的参数：将 train 和 val 修改为模型自己的路径，将 names 修改为数据的类别数。如图 4-3 所示，即修改数据类名为 ditou 和 taitou。

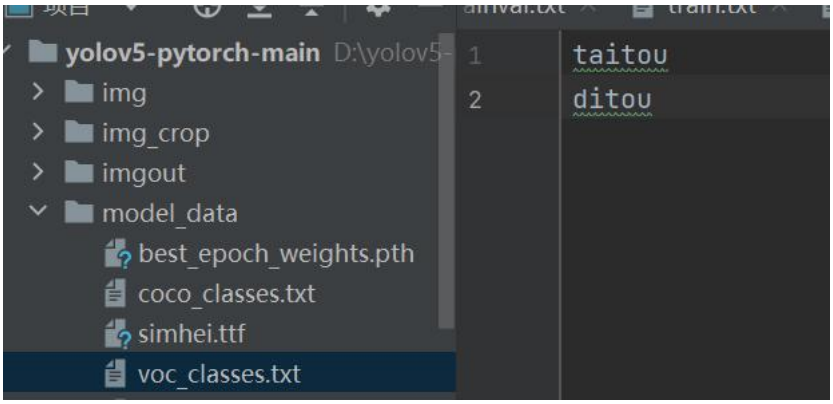

图 4-3 YOLOv5 模型版本差异图模型的数据类名

另外还需要修改的文件为模型配置文件，在之前选择的 YOLOv5s 的文件中去修改文件中的 nc 为项目的类别数即可。

在以上步骤完成之后，只要在命令输入框输入命令即可训练项目的数据集。在使用电脑自带的 CPU 训练后，发现训练速度不理想。在查阅资料后发现 CPU 仅符合处理少量复杂运算的情况，而性能更加强大的 GPU 才适合处理大量运算。使用带有大容量 GPU 的电脑训练，只需将代码中的－-device cpu 改为－-device 2 即可，具体编号可以根据每台电脑的任务管理器的信息进行修改。

4.4 课堂抬头率检测模型训练结果

在计算机训练数据集结束后，可以发现 train 文件夹下会自动生成模型的训练结果。其中：

- weights：训练生成权重。包含 best.py（最好）和 last.py（最新）。
- confusion：混淆矩阵。通过分析混淆矩阵我们可以发现模型所犯的错误，甚至可以了解正在发生的错误类型。

- F1_curve：F1 分数和置信度的关系图。
- P_curve：准确率和置信度的关系图。
- R_curve：召回率和置信度的关系图。
- PR_curve：P 代表的是精准率，R 代表的是召回率，曲线表现的是精准率和召回率之间的关系。
- labels：左上图表示类别的数量；右上图表示标签；左下图表示中心坐标；右下图表示标签的长和宽。

验证过程与训练过程步骤一致，要注意的是现在使用到的是训练好的文件(best.py)，而不是之前的预训练文件。扫描结束后同样也会生成结果文件。如图 4-4 所示，这是本系统模型运行时的所有配置。

```
Configurations:
----------------------------------------------------------------------
|                   keys |                                values|
----------------------------------------------------------------------
|             model_path |       model_data/best_epoch_weights.pth|
|            classes_path |              model_data/voc_classes.txt|
|            anchors_path |             model_data/yolo_anchors.txt|
|            anchors_mask |       [[6, 7, 8], [3, 4, 5], [0, 1, 2]]|
|             input_shape |                            [640, 640]|
|                backbone |                            cspdarknet|
|                     phi |                                     s|
|              confidence |                                   0.5|
|                 nms_iou |                                   0.3|
|         letterbox_image |                                  True|
|                    cuda |                                 False|
----------------------------------------------------------------------
```

图 4-4 模型目标检测的配置

第五章课堂抬头率检测系统的优化

5.1 课堂抬头率检测系统的性能优化

在模型运行过程中，发现了模型存在的一些问题:如使用电脑自带摄像头目标检测分析时，镜头捕捉人脸的速度太慢，导致用户体验感太差。

因此在查阅资料学习后发现有以下几种方法来对模型进行优化：

- 加强训练数据：使用更多的训练课堂照片使 YOLOv5 模型更好地学习到目标特征。
- 数据本身增强：使用数据增强技术，如缩放、旋转、翻转等。通过增加数据的多样性，提高目标检测模型的鲁棒性[15]。
- 优化参数：根据训练模型生成的学习率、迭代次数等超参数进行代码中参数的调整改进，来优化 YOLOv5 的训练效果。
- 网络架构调整：尝试替换 YOLOv5 网络的结构或增加别的层次结构，以提高模型其对检测图像的理解能力。
- 增加计算资源：使用更强的 GPU 或分布式计算环境可以提高 YOLOv5 的训练速度和检测精度。

考虑到现实的可行性，本系统从数据集入手，由原来的 200 张数据集扩充到 400 张数据集，并且对于图片中人脸的残缺状况进行了更为精细的处理。此外，对于数据集进行了旋转的操作，如图 5-1 所示。并且由系统训练得到的 P_curve 图显示，模型性能确实得到了提升。

图 5-1 数据增强操作

在考虑到实际教学情况，教学工作者会对于某个学生的抬头情况进行详细的关注与记录，所以系统多进行了一个对于图片中检测到的单个人脸进行裁剪突出的工作，如图 5-2 所示。

```
mode = "predict"

crop            = False#是否在单张图片预测后对目标进行截取
count           = True#是否对目标计数
#----------------------------------------------------------------------------------------------------------
#   video_path          用于指定视频的路径，当video_path=0时表示检测摄像头
#                       想要检测视频，则设置如video_path = "xxx.mp4"即可，代表读取出根目录
#   video_save_path     表示视频保存的路径，当video_save_path=""时表示不保存
#                       想要保存视频，则设置如video_save_path = "yyy.mp4"即可，代表保存为

video_path      = "C:/Users/HUWAEI/Desktop/yolov5-pytorch-main/img/ketang.mp4"
video_save_path = ""#不保存   保存视频时需要ctrl+c退出或者运行到最后一帧才会完成完整的保存步骤。
video_fps       = 25.0
```

图 5-2 截取单张人脸操作

通过将图 5-2 代码中的 False 改成 True 即可完成转换，效果如图 5-3。

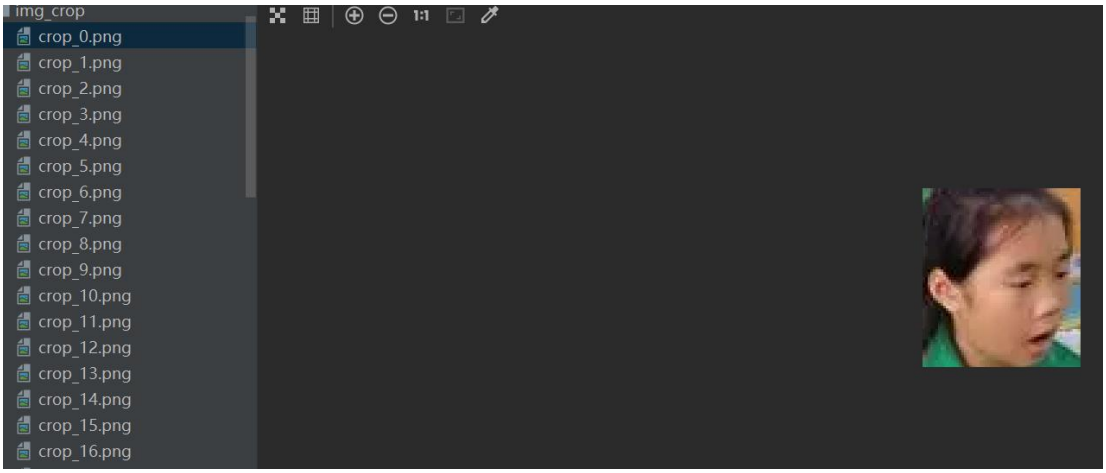

图 5-3 截取单张人脸效果图

5.2 课堂抬头率检测系统的实际应用场景优化

该系统可用于智慧教室课堂教学的过程中，通过系统大队学生抬头率的监测和分析，有助于更好地掌握学生的学习行为和学习习惯，从而帮助教师更好地帮助学生提高学习效果。另外，学生抬头率的监测和分析也可以作为一种教学行为分析工具，以有效地引导教师改进课堂教学，进而提高教学质量[18]。

再考虑到学生抬头情况的反面，也就是低头，该系统可以应用于学生考试与复习场景。通过对于学生低头率的检测与分析，可以了解学生在考试与复习时候的专注度，为最终的成绩分析提供可靠的支撑与证明。此外，在检测过程中出现的异常情况，系统也可以增加判定指标来进行优化与完善。

教学评价是教学过程的重要一环，不仅对于教师，对于学生也意义重大。但是光从最终一个书面成绩还无法说明学生或者教学一直以来的工作成效。因为，课堂抬头率检测的数据可以通过个人归档记录的方式来作为综合成绩评价的一项指标，为最终的学习结果增加说服力。

第六章 总结与展望

6.1 研究结论

项目主要使用的是基于 YOLOv5 注意力机制与感受野模块来对于课堂抬头率进行检测研究，可以做到对图片、视频乃至摄像头拍摄到的课堂学生的头部姿势进行精准地捕捉与检测。对于检测结果也用精准的数字来表明，教育工作者们可以根据这些数据评估课堂教学质量，为实施个性化的教学提供参考与依据。为了实现这个最终的目标，本文大致分为三部分：

- 在网络上搜罗关于 YOLOv5 模型的架构与原理，了解选用该模型的原因。学习模型的原理，对于模型是如何帮助检测课堂抬头率有一个系统可达到的预想，为后续项目的完成打好基础。
- 借助 Labelimg 工具去标注爬虫收集整理得到的数据集，并且通过训练，测试来生成一个能够进行精准目标检测的模型。
- 为了获得更好的检测效果，对于 ConvNeXt，Swin Transformer 和 CSPDarknet 三种网络进行斟酌和实际的操作，进行取舍。此外，对于模型的参数也有所调整，以达到更高的精度和更快的速度的效果。
- 对于最后的检测结果进行可视化处理，能够对于检测出来的数据进行分析。

本项目实现的课堂抬头率检测系统的一大特点是可以实现学生头部姿势情况的实时检测，这也正是在如今中小学乃至大学课堂实际应用的空白之处。我们的技术只有切合实际需要，才能发挥出它真正的用途与价值。

6.2 研究展望

该项目在教育大背景下结合了计算机技术，选题也是社会的重点关注问题，充分体现了时代性和现实的社会价值。并且随着科技的日新月异，其检测出的结果会越来人性化，应用的领域也会越来越广泛。

- 在实际课堂教学环境中存在讲授型课堂和研讨型课堂两种教学场景，该项目对于讲授型课堂比较适用。随着素质教育的普及，对于研讨型的课堂也需要进行数据化的课堂质量评价。因此项目还有待优化完善，兼容两种教学场景的课堂抬头率的检测[19]。
- 该网络由于电脑性能的缺陷，所以选用的网络是规模较小的 CSPDarknet。在该模型的基础上还可以进行剪枝操作，将冗余的部分删去，从而达到提高模型性能的效果。此外，图像中存在一定数量的小目标，占整幅图像比例较小，易受背景等因素的影响。为进一步加强图像中位置信息与检测特征的关联程度，可以引入能够将坐标信息嵌入通道信息的 CA 注意力机制[20]。
- 如果出现性能更加强大的目标检测参考模型，项目就要紧跟前沿变化，考虑新需求，用新技术去挖掘项目更大的潜能。

随着知识经济的到来，教育显得尤其重要，我们要更加注重促进信息化时代下教育领域的进步。

参考文献

[1] 何佳宸,张虹. 人脸识别技术在高校学生听课质量监控中的应用研究[J]. 信息系统工程,2019(03):102.

[2] 孙亚丽. 基于人脸检测的小学生课堂专注度研究[D]. 湖北师范大学,2016:6-26.

[3] 孙众,吕恺悦,骆力明等. 基于人工智能的课堂教学分析[J]. 中国电化教育,2020(10):15-23.

[4] 唐康. 人脸检测和表情识别研究及其在课堂教学评价中的应用[D]. 重庆师范大学,2019:6-18

[5] 钱铠伦,谢凯,姜宏屏等. 基于 Python 语言的课堂抬头率检测方法研究[J]. 电子世界,2020(03):39-40.

[6] 陈玥,李会会,韩嘉彬等. 基于卷积神经网络技术的大学生隐性消极课堂行为识别研究[J]. 太原城市职业技术学院学报,2020(08):89-91.

[7] 巢渊,刘文汇,唐寒冰等. 基于改进 YOLO-v4 的室内人脸快速检测方法[J]. 计算机工程与应用,2022,58(14):105-113.

[8] 胡正平,张乐,李淑芳等. 端对端 SSD 实时视频监控异常目标检测与定位算法[J]. 燕山大学学报,2020,44(05):493-501.

[9] 邵延华,张铎,楚红雨等. 基于深度学习的 YOLO 目标检测综述[J]. 电子与信息学报,2022,44(10):3697-3708.

[10] 范鹏飞. 基于深度学习的课堂抬头率研究[D]. 辽宁科技大学,2021:12-34.

[11] 郭敏钢,宫鹤. 基于 Tensorflow 对卷积神经网络的优化研究[J]. 计算机工程与应用,2020,56(01):158-164.

[12] Kyrkou Christos, . YOLOpeds: efficient real-time single-shot pedestrian detection for smart camera applications[J]. IET Computer Vision, 2020, 14(7):417-425.

[13] 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.

[14] 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.

[15] 晁晓菲,池敬柯,张继伟等.基于 PSA-YOLO 网络的苹果叶片病斑检测[J].农业机械学报,2022,53(08):329-336.

[16] 王亮,张超.一种基于 YOLOv5 的轻量型行人检测方法[J].工业控制计算机,2023,36(04):84-89.

[17] 孟帅.基于 YOLOv5 的绝缘子目标检测算法[J].现代信息科技,2023,7(08):107-110.

[18] 田嫚嫚. 中学政治课堂抬头率问题研究[D].苏州大学,2015:1-6.

[19] 郭春麟. 一种基于计算机视觉的课堂注意力模型的构建与实现[D]. 华中科技大学,2021:6-26.

[20] 祁泽政,徐银霞.改进 YOLOv5s 算法的安全帽佩戴检测研究[J/OL].计算机工程与应用:1-10

致谢

言多为虚，恭恩惟心。

时光荏苒，写完这篇毕业论文，也意味着我的大学四年时光即将画上句号。构思书写这篇论文期间，也是我备战考研之际。本以为兵荒马乱，结果好像也尽如人意。

感谢母校四年来的悉心培养，感谢学习以来各位任课教师的真挚指点，感谢论文指导老师吴茂念老师的宽容与鼓励。

感谢父母家人在我背后的默默支持，爱不挂口，常记于心。

感谢朋友一直以来的陪伴，耐心倾听我的情绪，给予我源源不断的动力。

感谢自己坚持到底，即使一路荆棘，也绝不放弃。愿以后的每一刻，都能铭记 22 岁为未来拼搏的自己。