基于注意力机制的光线昏暗条件下口罩佩戴检测

摘要

自 2019 年新冠肺炎疫情暴发以来，口罩佩戴检测成为疫情防控的必备操作。针对在光线昏暗条件下口罩佩戴检测准确率较低的问题，提出了将注意力机制与 YOLOv5 网络模型相结合的口罩佩戴检测方法。首先对训练集图片使用图像增强算法进行预处理，然后将图片送入到引入了注意力机制的 YOLOv5 网络中进行迭代训练，完成训练后，将最优权重模型保存并在测试集上测试。实验结果表明，在注意力的加持下，该模型能有效的增强人脸和口罩等关键点信息的提取，提高模型的鲁棒性，在光线昏暗条件下对口罩佩戴的检测准确率能达到 92% ，能够有效满足实际需求。

关键词: 注意力机制；口罩检测；目标定位；目标识别；YOLOv5

Detection of Mask Wearing in Dim Light Based on Attention Mechanism

Abstract

Since the outbreak of COVID-19 in 2019, the detection of wearing masks has become a necessary measure for epidemic prevention and control. To solve the problem about low accuracy of mask wear detection under dim lighting conditions, a method of mask wear detection combining attention mechanism with YOLOv5 network model is proposed, which uses image enhancement algorithm to pre-process the training set pictures, and then put these pictures to YOLOv5 network with attention mechanism for iterative training. After training, the optimal weight is saved and the best model is used to test the accuracy on the test set. The experimental results show that the YOLOv5 network model with attention mechanism can effectively enhance the extraction of key points such as face and mask and improve the robustness of the model. The accuracy of mask wearing can reach 92% under dim lighting conditions, which can effectively meet the actual needs.

Keywords: attention mechanism; mask detection; object localization; object recognition; YOLOv5

1 引言

新冠肺炎疫情暴发后，人类健康受到巨大威胁，人们正常的工作与生活也受到了极大影响。为避免新冠肺炎疫情继续传播，人们在外出时规范佩戴口罩便成为了一项有效的防控手段。仅靠人工方式对流动人员进行口罩佩戴检测不仅效率较低，且会耗费大量人力物力，同时由于新冠肺炎传染性极强，近距离接触待检测人员更是存在较大的安全风险。因此，构建口罩自动检测系统检测活动人员在各类车站、大型商场等公共场合的口罩佩戴情况，对疫情防控具有重要的现实意义。

基于深度学习的目标检测算法十分丰富,主要分为两大类,一类是以 R-CNN[1]、Fast-RCNN[2] 为代表的两阶段检测算法，该类算法在特征提取的基础上，由独立的网络分支先生成大量的候选区，再对其进行分类和回归。另一类是以 SSD[3]、YOLO(You Only Look Once) 系列算法为代表的一阶段检测算法，该类算法在生成候选框的同时进行分类和回归任务。

针对口罩佩戴检测这一特定问题，目前国内外有多位学者进行了研究。邓黄潇等人[4] 提出了运用迁移学习和 ReinaNet 网络对口罩佩戴进行检测，验证集下 AP 达到 86.5%。肖俊杰等人[5] 将 YOLOv3 和 YCrCb 方法结合，正确佩戴口罩的识别率的达到 82.5%。牛作东等人[6] 对 RetinaFace

算法进行了相应的优化，同时引入了自注意力机制，实验结果表明检测效果较好。AIZOO[1]也提出了一种基于 Fast-RCNN 和 YOLOv3 的目标检测方法，在口罩佩戴检测任务上取得了不错的成绩，但对可见度不高、光照强度不强的昏暗条件下，其检测精度仍有待提高。

本文主要针对在可见度不高、光线昏暗的场景下，利用图像增强算法对图片进行预处理，将通道注意力和空间注意力结合，充分挖掘人脸口罩等关键特征点，同时对 YOLOv5 网络的损失函数进行相应的改进，提高模型在昏暗条件下的鲁棒性。

2 YOLOv5 网络模型简介

YOLO 是一个高性能的通用目标检测模型，YOLOv1[7]创造性地使用一阶段的检测算法直接完成目标位置定位和目标物体分类两个任务。随后，YOLOv2[8]相对 v1 版本，在继续保持处理速度的基础上，从预测更准确、速度更快、识别对象更多这三个方面进行了改进，其中识别更多对象也就是扩展到能够检测 9000 种不同对象，称之为 YOLO9000。YOLOv3[9]通过引入多尺度预测、改进基础网络和损失函数等进一步加速了目标检测在工业界的落地。YOLOv4[10]提出了一种高效而强大的目标检测模型，使得其能够在在一块普通的 GPU(如 GTX 1080Ti) 上训练超快速和准确的目标检测器。YOLOv5 相较于 YOLOv4, 其结构更加小巧灵活，图像推理速度更快，能够满足视频图像实时性检测的需求。YOLOv5 自身也有其版本迭代，现已更新迭代至 v5.0 版本，本文使用最新版本 v5.0。YOLOv5 提供了 4 个大小的版本模型，分别是 YOLOv5s、YOLOv5m、YOLOv5l、YOLOv5x, 这 4 个版本模型的大小越来越大, 对应的目标检测精度也越来越高。本文采用 YOLOv5s 作为基础模型进行研究。YOLOv5 的网络结构如图 1所示。

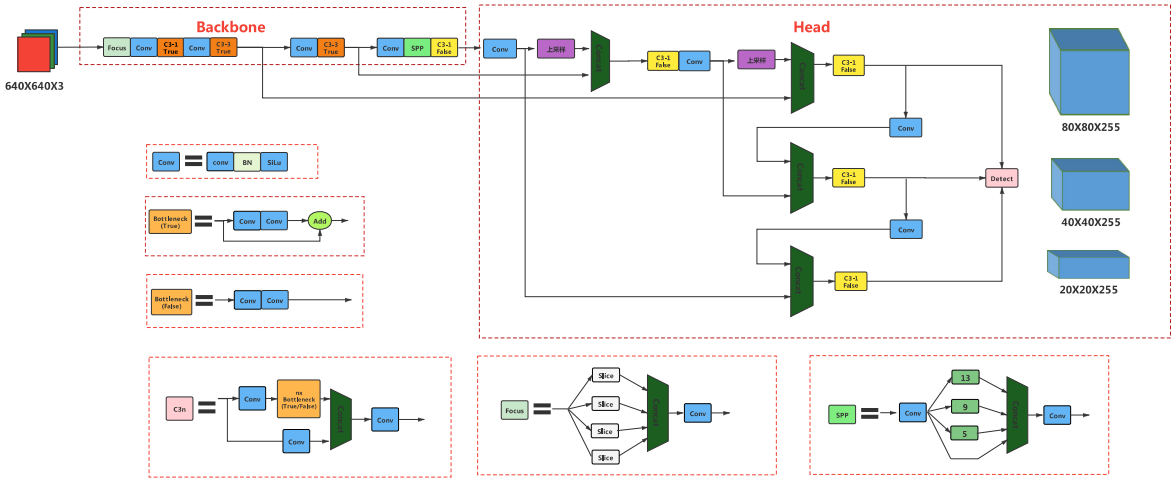


图 1: YOLOv5 网络结构图

在 YOLOv5 v5.0 版本中, 使用了新的激活函数 SiLU() 来替换先前版本中的激活函数 LeakReLU() 和 Hardwish(), 使得网络中的任何地方都只使用 SiLU() 激活函数，如公式 1所示, 其中 $\sigma(x)$ 是 Sigmoid 激活函数。SiLU() 激活函数及其导数的图像如图 2所示。

$$SiLU(x) = x \cdot \sigma(x) \tag{1}$$

[1]https://github.com/aky15/AIZOO_torch

同时，YOLOv5 中删减了先前版本中的 BottleneckCSP 中的部分 Conv 模块，经过改进后的 Bot-

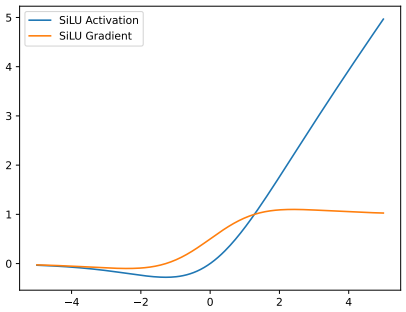


图 2: SiLU 激活函数及其导数

tleneckCSP 称为 C3 模块。C3 模块的详细结构见图 1。由于 C3 模块移除了每个瓶颈结构中的一个卷积，导致新版本的 YOLOv5 能够得到更小的模型，且推理速度也有少许提升。

YOLOv5 由 Backbone 与 Head 两部分组成，Backbone 主要有 Focus、C3 以及 SPP(Spatial Pyramid Pooling) 模块，Head 包括 PANet(Path Aggregation Network) 和 Detect 模块。在 Backbone 骨干网络中，由 1 个 Focus 模块及 4 个 Conv 模块实现 32 倍下采样，其中 Focus 模块将输入数据进行切片操作，在一张图片中每隔一个像素取一个值，类似于邻近下采样，切分为 4 份数据，每份数据都是相当于 2 倍下采样得到的，然后在通道维度上进行拼接，最后再进行卷积操作。这一过程并不会使原始图片信息丢失，只是将图片宽高维度上的信息集中到了通道维度。Focus 模块减少了卷积的成本，用重置张量维度的方法巧妙地实现了下采样并增加通道维度。

C3 模块参照 CSPNet(Cross Stage Partial Network) 结构[11]将一个阶段中基础层的特征图分成两部分，拆分和合并策略被跨阶段使用，将梯度的变化从头到尾地集成到特征图中，在减少了计算量的同时可以保证准确率。C3 模块使得 YOLOv5 网络能够有更好的学习能力，能够在轻量化的同时保持准确性，同时降低了计算瓶颈和内存成本。YOLOv5 中有两种 C3 模块，以有无残差边为区分标准分为 C3-False 和 C3-True，用 shortcut 的取值 False 或 True 来控制改变。

Head 部分里，通过将高层的特征信息通过上采样的方式与低层特征信息进行传递融合，传达强语义特征，实现自顶向下的信息流动，再通过步长为 2 的卷积进行处理，将底层特征与高层特征进行拼接操作，使底层的特征信息容易传到上层去，从而实现了 PANet[12]操作，更好地将底层与高层特征优势互补，强化定位特征，有效解决多尺度问题。

3 改进后的网络模型

YOLOv5 随着其版本的迭代更新，网络内部的各个模块针对物体检测中的常见问题都做了优化处理，选择 YOLOv5 作为人脸口罩检测模型的基础网络是可行的。

注意力机制最早在 2014 年率先被 Google DeepMind 团队引入到 RNN 模型上来实现图像的分类[13]，实现了图像中多个物体对象的高效准确的识别，使其在 MNIST 数据集上的分类任务中错误率下降了 4%, 验证了注意力机制在图像处理领域的有效性[14]，同时也使得结合注意力机制的神经网络成为目前的研究热点。在卷积神经网络中, 注意力机制作用在特征图之上, 通过获取特征图中的可用注意力信息, 能够达到更好的任务效果[15]。

在昏暗条件下，光照强度不大，可见度低，难以对人脸进行精确的定位，口罩佩戴检测任务的难度也更为困难，因此需要对 YOLOv5 的网络做进一步的结构优化和调整。

3.1 损失函数

模型损失函数 (Loss) 由分类损失 (Classification Loss)、定位损失 (Localization Loss) 和目标置信度损失 (Confidence Loss) 组成, 如公式 2所示。

$$Loss = Classification \quad Loss + Localization \quad Loss + Confidence \quad Loss \tag{2}$$

本文采用二元交叉熵损失函数来计算分类损失和目标置信度损失，分别如公式 3和公式 4所示。

$$\begin{aligned} Classification \quad Loss = & \sum_{i=0}^{K\times K}\sum_{j=0}^{M} I_{ij}^{\text{obj}} \left[\left(\hat{C}_i \log\left(C_i\right) + \left(1- \ \hat{C}_i\right)\log\left(1-C_i\right)\right)\right] \\ & - \lambda_{\text{no\_obj}} \sum_{i=0}^{K\times K}\sum_{i=0}^{M} I_{ij}^{\text{no\_obj}} \left[\left(\hat{C}_i \log\left(C_i\right) + \left(1- \ \hat{C}_i\right)\log\left(1-C_i\right)\right)\right] \end{aligned} \tag{3}$$

$$Confidence \quad Loss = \sum_{i=0}^{K\times K}\sum_{c\in \text{ lasses}} I_{ij}^{\text{obj}} \left[\left(\hat{p}_i(c)\log\left(p_i(c)\right) + \ \left(1-\hat{p}_i(c)\log\left(p_i(c)\right)\right)\right] \tag{4}$$

其中，K 表示网络最后输出的的特征图划分为 $K\times K$ 个格子,M 表示每个格子对应的锚框的个数, I_{ij}^{obj} 表示有目标的锚框, $I_{ij}^{\text{no\_obj}}$ 表示没有目标的锚框, $\lambda_{\text{no\_obj}}$ 表示对没有目标锚框的损失系数。

常用的计算定位损失函数有 GIOU Loss[16] 、DIOU Loss[17] 和 CIOU Loss[18] ，经过对比实验，本文采用 CIOU Loss 作为目标框回归的损失函数,CIOU Loss 公式如 5所示。

$$CIOU \quad Loss = 1 - \left(IoU - \frac{d_1^2}{d_2^2} - \beta\alpha\right) \tag{5}$$

其中，$\beta = \frac{\alpha}{(1-IoU)+\alpha}, \alpha = \frac{4}{\pi^2}\left(\tan^{-1}\frac{W^{gt}}{h^{gt}} - \tan^{-1}\frac{W}{h}\right)^2$，$d_1$ 表示预测框与目标框两个中心点的欧式距离，d_2 表示最小外接矩形的对角线距离。$\frac{W^{gt}}{h^{gt}}$ 和 $\frac{W}{h}$ 分别表示目标框和预测框各自的的宽高比，具体的推导过程详见参考文献[18]。由于 $CIOU \quad Loss$ 从重叠面积、中心点距离和长宽比三个角度进行衡量，故预测框回归的效果更佳。

3.2 引入卷积注意力模块

在原始 YOLOv5 的网络基础上，引入卷积注意力模块 CBAM[19] (Convolutional Block Attention Module Network)。CBAM 包含两个子模块，分别是通道注意力模块 CAM(Channel Attention Module) 和空间注意力模块 SAM(Spatial Attention Module)。CAM 汇总通道注意力信息，SAM 汇总空间注意力信息。图 3为 CBAM 的网络结构。

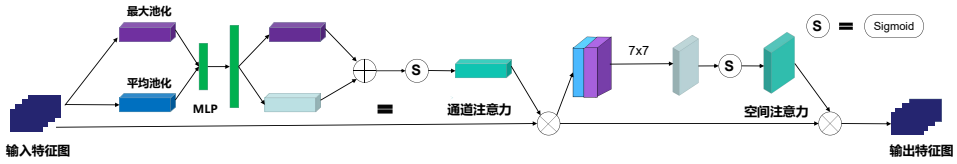


图 3: CBAM 网络结构图

CAM 为给定的任意中间特征图 $F \in R^{C\times H\times W}$，使用基于宽和高的最大池化操作 (global max pooling) 和全局平均池化操作 (global average pooling) 对特征映射在空间维度上进行压缩，得到 $F_{max}^c \in R^{C\times 1\times 1}$ 与 $F_{avg}^c \in R^{C\times 1\times 1}$ 两个特征图，这两个特征图共享一个两层的神经网络 MLP, 第一层神经元的个数为 C/r(r 为减少率)，激活函数是 ReLU, 第二层的神经元个数是 C, 然后对 MLP

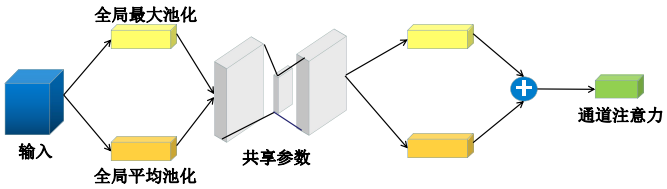


图 4: CAM 结构图

输出的两个特征图使用基于元素的加和操作，再经过 Sigmoid 激活函数进行归一化处理，得到最终的通道注意力特征图为 $M_c \in R^{C\times1\times1}$。CAM 的结构如图 4所示。

与通道注意力不同，SAM 主要关注于目标在图像上的位置信息，它将 CAM 的输出特征图作为本模块的输出特征图。它首先做一个基于通道的全局最大池化和全局平均池化，分别得到 $F^s_{max} \in R^{1\times H\times W}$ 和 $F^s_{avg} \in R^{1\times H\times W}$ 这两个特征图，然后将这两个特征图串联，基于通道做拼接操作，再经过一个 7×7 卷积操作生成空间注意力特征图 $M_s \in R^{1\times H\times W}$。SAM 的结构如图 5所示。

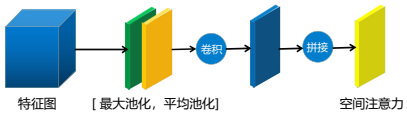


图 5: SAM 结构图

结合前文介绍，可以得到本文改进后的 YOLOv5 网络整体结构如表 1所示。表 1中 From 列的-1 是指输入来自上一层输出，Arguments 列的值分别代表该模块的输入通道数、输出通道数、卷积核大小和步长信息，经过计算，引入注意力机制的 YOLOv5 模型总计 367 层，7150056 个 parameters。

由于 CBAM 模型在通道注意力模块中加入了全局最大池化操作，它能在一定程度上弥补全局平均池化所丢失的信息。其次，生成的二维空间注意力图使用卷积核大小为 7 的卷积层进行编码，较大的卷积核对于保留重要大的卷积核对于保留重要的空间区域有良好的帮助。添加了 CBAM 的 YOLOv5 网络不仅能更为准确的对目标进行分类识别，而且能更为精准的定位目标所在的位置。

4 实验与结果分析

4.1 数据集及实验环境介绍

目前开源的口罩佩戴检测数据集较少，因此实验通过网络爬取与自行拍摄相结合制作数据集，数据集中 80% 来源源于网络，20% 来源于实际拍摄。在昏暗条件下的口罩佩戴图片相对缺乏，实际拍摄主要获取的是这类图片，实验过程中从楼道、室内等光线昏暗的场所进行了图片采集，同时也在傍晚和清晨等光线较弱的环境下进行了拍摄。数据集部分图片如下图 6 所示。

实验数据集共包含 9000 张图片，对其进行手工标注。该数据集分为两种类别，分别是 bad 和 good，bad 表示人员未佩戴或未按规范佩戴口罩，good 表示正确佩戴口罩。数据集采用 YOLO 格式，图片标注使用 LabelImg，标注后的文件以 txt 作为后缀，文件名称和图片名称一致。训练集和测试集所占的比例为 8：1，如表 2所示。

表 1: CBAM-YOLOv5 网络结构

序号	From	Params	Module	Arguments
0	-1	3520	Focus	[3,32,3]
1	-1	18560	Conv	[32,64,3,2]
2	-1	18116	C3	[64,64,1]
3	-1	679	CBAM	[64,64]
4	-1	73984	Conv	[64,128,3,2]
5	-1	156928	C3	[128,128,3]
6	-1	2283	CBAM	[128,128]
7	-1	295424	Conv	[128,256,3,2]
8	-1	625152	C3	[256,256,3]
9	-1	8563	CBAM	[256,256]
10	-1	1180672	Conv	[256,512,3,2]
11	-1	656896	SPP	[512,512,[5,9,13]]
12	-1	1182720	C3	[512,512,1,False]
13	-1	33411	CBAM	[512,512]
14	-1	131584	Conv	[512,256,1,1]
15	-1	0	Upsample	[None,2,'nearest']
16	[-1, 7]	0	Concat	[1]
17	-1	361984	C3	[512,256,1,False]
18	-1	33024	Conv	[256,128,1,1]
19	-1	0	Upsample	[None,2,'nearest']
20	[-1, 4]	0	Concat	[1]
21	-1	90880	C3	[256,128,1,False]
22	-1	2283	CBAM	[128,128]
23	-1	147712	Conv	[128,128,3,2]
24	[-1, 18]	0	Concat	[1]
25	-1	296448	C3	[256,256,1,False]
26	-1	8563	CBAM	[256, 256]
27	-1	590336	Conv	[256,256,3,2]
28	[-1, 14]	0	Concat	[1]
29	-1	1182720	C3	[512, 512, 1, False]
30	-1	33411	CBAM	[512, 512]
31	[17, 20, 23]	13503	Detect	[2,[[10,13,16,30,33,23],[30,61,62,45,59,119],[116,90,156,198,373,326]], [256,256,128]]

本文数据集存在轻微类间不平衡[20]的问题，如图 7a 所示，这是物体检测的常见问题。模型训练时，当某一类别的样本数量较少时，可能会使模型对该类别的关注较小，模型会主要关注样本数量较多的类别，模型网络参数也会主要根据样本数量较多的类别进行调整，从而导致模型对样本数量较少的类别的识别准确率降低。为了解决该问题，本文使用图像增强技术，对标签为 good 的样本图片进行图像平移、翻转、旋转、缩放，分离三个颜色通道并添加随机噪声来有效缓解类间不平衡的问题, 图像增强后的数据集类别分布如 7b 所示。

对增强后的数据集进行可视化分析如图 8所示，图 8a 中 x、y 是指中心点的位置, 颜色越深代表该点位置目标框的中心点越集中；图 8b 中 width、height 分别代表图片中物体的宽高。可以看出增强后的数据集物体分布相对均匀, 且中小物体占比更大, 存在物体间遮挡的情况, 符合日常实际应用场景。

实验环境使用 Ubuntu18.04 操作系统，采用 Pytorch 架构，使用 GeForce GTX 1080Ti 显卡进行运算，显存大小为 11GB。具体实验配置如表 3所示。

4.2 模型评估指标

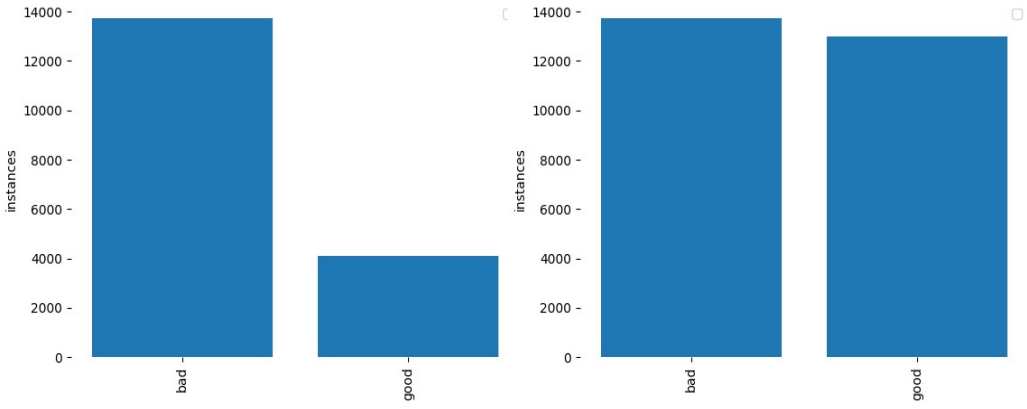
模型评估指标主要使用平均精度均值 (mAP)、召回率 (Recall)、准确率 (Precision) 如图 9所示。平均精度均值 (mAP)，即所有类别的平均精度求和除以数据集中所有类的平均精度，如公式

a. 昏暗 b. 较昏暗 c. 正常光照

图 6: 数据集部分图片

表 2: 口罩数据集划分

参数	数值 (张)
总数据集	9000
训练集	8000
测试集	1000



a. 原始数据集类别分布 b. 图像增强后数据集类别分布

图 7: 数据集类别分布

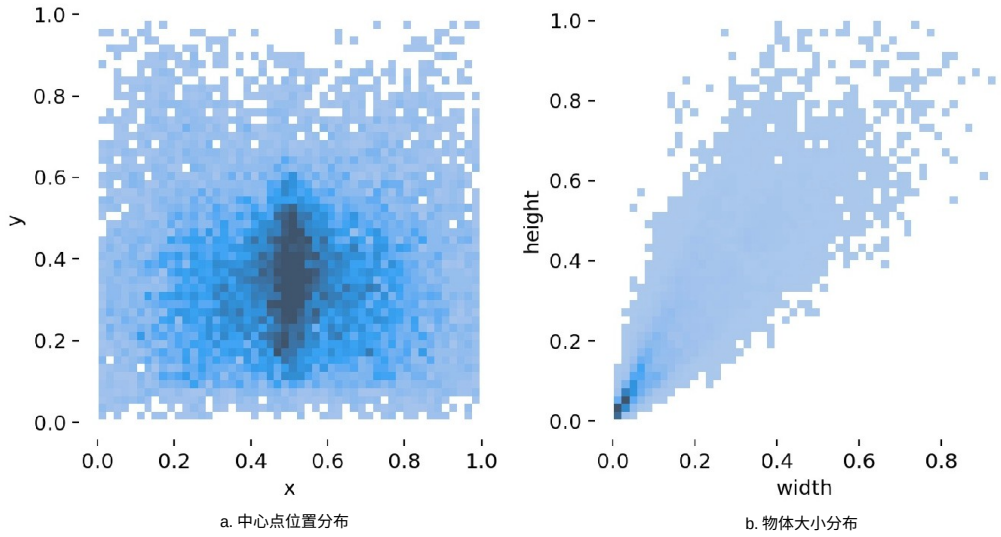


图 8: 图像增强后数据集可视化分析

表 3: 实验环境配置

参数	配置
操作系统	Ubuntu18.04
CPU	Intel(R) Xeon(R) CPU E5-2620 v3 @ 2.40GHz
GPU	GeForce GTX 1080Ti
语言	Python(版本-3.7.4)
架构	Pytorch(版本-1.9.0)
加速环境	CUDA(版本-11.4)

6所示，其中 AP 的值为 P-R 曲线的面积。

$$mAP = \frac{\sum_{i=1}^{N} AP_i}{N} \tag{6}$$

召回率，即样本中的正确类别被模型预测正确的概率，如公式 7所示，其中 TP 表示将正类别预测为正类别的个数，FN 表示将正类别预测为负类别的个数。

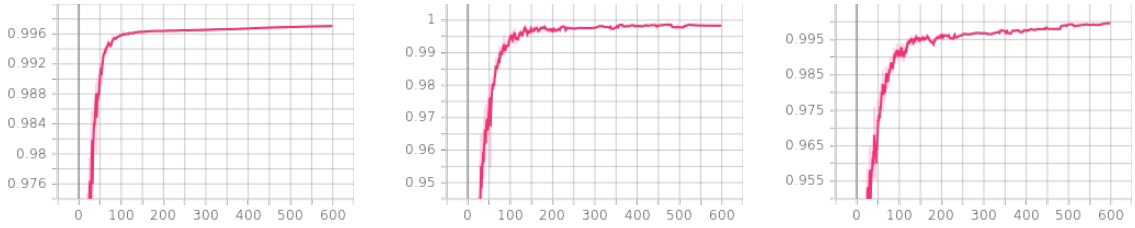
$$Recall = \frac{TP}{TP + FN} \tag{7}$$

准确率，即预测数据集中预测正确的正样本个数除以实际的正样本个数，如公式 8所示，其中 FP 表示将负类别预测为正类别的个数。

$$Precision = \frac{TP}{TP + FP} \tag{8}$$

从图 9a 中可以看出，当迭代次数接近 400 次左右时，平均精度均值的数值接近于 0.996; 从图 9b 中可以看出，当迭代次数接近 450 次左右时，召回率的数值接近于 1; 从图 9c 中可以看出，当迭代次数接近 500 次时，准确率的数值接近于 0.995。

网络模型训练阶段，迭代批量大小设置为 32, 总迭代次数为 600 次。初始学习率设置为 0.001, 采用小批量梯度下降法，并使用 Adam 优化器计算每个参数的自适应学习率。大约在 350 次迭代后，模型开始逐渐收敛。



(a) mAP (b) Recall (c) Precision

图 9: 模型性能评估

从图 9中可以看出，引入了注意力机制的 YOLOv5 网络模型在训练阶段的结果比较理想。

4.3 实验结果对比

模型训练完成后，将得到的模型与参考文献[5]中的方法和 AIZOO 方法的口罩检测模型进行对比实验，分别在光照强度为 30-75Lux（昏暗），75-250Lux（较昏暗）和 250-1000Lux（正常光照）的条件下进行对别实验，实验结果如图 10所示。其中，光照强度是指单位面积上所接受可见光的能量，常用于指示光照的强弱和物体表面积被照明程度的量，单位是 Lux，光照强度越大，表明光照越强，物体表面被照的越亮。

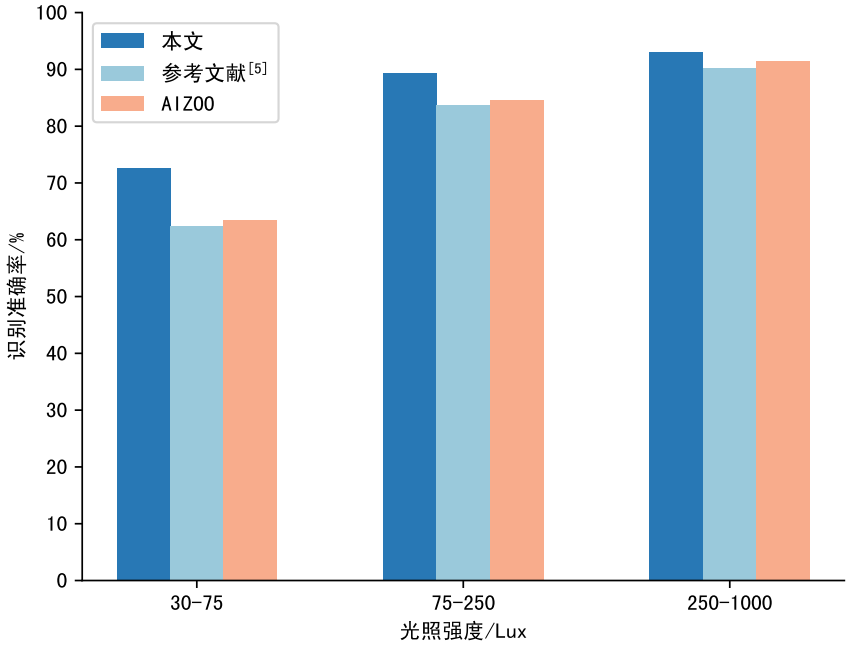
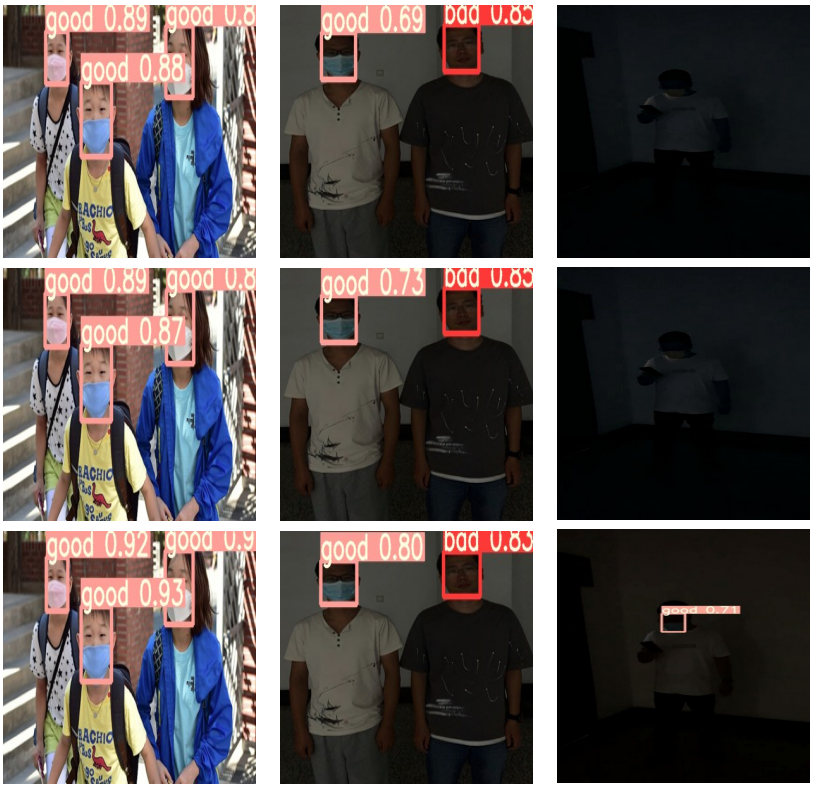


图 10: 3 种方法试验结果

可以看出，在一定的光照强度范围内，随着光照强度的增加，本文方法与参考文献[5]和 AIZOO 方法对口罩佩戴检测的准确率也逐渐增加。在光照强度为 250-1000Lux 的正常光照条件下，本文方法与参考文献[5]和 AIZOO 方法对口罩佩戴检测的准确率接近。在光照强度为 75-250Lux 的较昏暗条件下，参考文献[5]和 AIZOO 方法对口罩佩戴检测的准确率比较接近，但本文方法对口罩佩戴检测的准确率比参考文献[5]和 AIZOO 方法分别高 5.6%、4.8%。在光照强度为 30-75Lux 的昏暗

条件下，参考文献[5]和 AIZOO 方法对口罩佩戴检测的准确率相差不大，在图像增强和注意力机制的加持下，经过改进的 YOLOv5 模型实现了口罩佩戴的高效检测，对是否正确佩戴口罩作出了正确的判断。在光线昏暗的条件下，本文方法的检测精度要比参考文献[5]中的方法和 AIZOO 方法高 10.2%、9.3%。

进一步，我们将算法应用在在实际场景中，进行实验的结果如图 11所示，图 11中的每一列分别是使用参考文献[5]、AIZOO 与本文方法在同一种光照条件下的检测效果。从图 11a 中可以看出，在光照正常的条件下，本文方法与参考文献[5]中的方法和 AIZOO 方法均实现了口罩佩戴的检测，正确地检测出了图片中的目标。图 11b 中，在光线较昏暗的条件下，参考文献[5]中的方法和 AIZOO 方法能检测出人脸信息和口罩佩戴状态，但本文方法检测出的目标的置信度比参考文献[5]和 AIZOO 方法要高。图 11c 中，在可见度不高，光线昏暗的条件下，识别人脸信息和口罩的难度增大，参考文献[5]中的方法和 AIZOO 方法无法获取到人脸口罩关键点信息，导致目标检测失败，但本文方法不仅成功地检测到了人脸信息，而且正确地检测出口罩佩戴的状态，并给出了对应的置信度。



a. 正常光照 b. 较昏暗 c. 昏暗

图 11: 口罩佩戴检测效果实际场景对比示例图

实验结果表明，在可见度不高、光照强度不大的昏暗条件下，与参考文献[5]和 AIZOO 的方法相比，本文方法使用图像增强能改善图片的质量，然后通过注意力机制能更加准确的提取人脸口罩关键点特征，从而使得检测的准确率更高，具有较强的鲁棒性和扩展性，基本能够达到视频图像实时性的要求。

5 结束语

本文提出的基于注意力机制的光线昏暗条件下的口罩检测方法可应用在口罩佩戴识别系统中，具有较强的鲁棒性和可扩展性，对推进口罩佩戴检测的自动化、智能化，实现疫情防控和公共卫生安全具有重要的现实意义。

参考文献

[1] GIRSHICK R, DONAHUE J, DARRELL T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 580-587.

[2] REN S, HE K, GIRSHICK R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[J]. Advances in neural information processing systems, 2015, 28: 91-99.

[3] LIU W, ANGUELOV D, ERHAN D, et al. Ssd: Single shot multibox detector[J]. European Conference on Computer Vision, 2016.

[4] 邓黄潇. 基于迁移学习与 RetinaNet 的口罩佩戴检测的方法[J]. 电子技术与软件工程, 2020, 000(005): P.209-211.

[5] 肖俊杰. 基于 YOLOv3 和 YCrCb 的人脸口罩检测与规范佩戴识别[J]. 软件, 41(7): 6.

[6] 牛作东, 覃涛, 李捍东, 等. 改进 RetinaFace 的自然场景口罩佩戴检测算法[J]. 计算机工程与应用, 2020.

[7] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: Unified, real-time object detection[J]. IEEE, 2016.

[8] REDMON J, FARHADI A. Yolo9000: Better, faster, stronger[J]. IEEE Conference on Computer Vision & Pattern Recognition, 2017: 6517-6525.

[9] REDMON J, FARHADI A. Yolov3: An incremental improvement[J]. arXiv e-prints, 2018.

[10] BOCHKOVSKIY A, WANG C Y, LIAO H. Yolov4: Optimal speed and accuracy of object detection[J]. 2020.

[11] WANG C Y, LIAO H, YEH I H, et al. Cspnet: A new backbone that can enhance learning capability of cnn[J]. 2019.

[12] LIU S, QI L, QIN H, et al. Path aggregation network for instance segmentation[J]. IEEE, 2018.

[13] MNIH V, HEESS N, GRAVES A, et al. Recurrent models of visual attention[C]//Advances in neural information processing systems. 2014: 2204-2212.

[14] BA J, MNIH V, KAVUKCUOGLU K. Multiple object recognition with visual attention[J]. arXiv preprint arXiv:1412.7755, 2014.

[15] 朱张莉, 饶元, 吴渊, 等. 注意力机制在深度学习中的研究进展[J]. 中文信息学报, 2019, 33(6).

[16] REZATOFIGHI H, TSOI N, GWAK J, et al. Generalized intersection over union: A metric and a loss for bounding box regression[C]//2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2019.

[17] ZHENG Z, WANG P, LIU W, et al. Distance-iou loss: Faster and better learning for bounding box regression[C]//AAAI Conference on Artificial Intelligence. 2020.

[18] ZHENG Z, WANG P, REN D, et al. Enhancing geometric factors in model learning and inference for object detection and instance segmentation[J]. 2020.

[19] WOO S, PARK J, LEE J Y, et al. Cbam: Convolutional block attention module[M]. Springer, Cham, 2018.

[20] OKSUZ K, CAM B C, KALKAN S, et al. Imbalance problems in object detection: A review [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, PP(99): 1-1.