

# 一种检测安全帽佩戴的深度学习算法

周筱雨, 王朝立, 孙占全

上海理工大学, 上海

收稿日期: 2021年12月31日; 录用日期: 2022年2月1日; 发布日期: 2022年2月8日

## 摘要

在生产流水线过程中, 工人因未佩戴安全帽而引发的悲剧不计其数, 为了帮助工厂监督人员督促工人们佩戴安全帽以保障自身人身安全, 传统的检测方式往往会分离安全帽与人体, 导致无法判断人与安全帽之间的佩戴关系, 为此本文提出了一种基于深度学习的改进YOLO算法的检测算法, 在现有的YOLO算法基础上, 对其损失函数进行改进, 将原本的GIoU\_Loss函数替换为CIoU\_Loss函数, 使得算法识别准确率得到提升且收敛速度更快。针对YOLO算法在检测时对安全帽的定位易受噪声等多方面的影响, 为此本文设计了安全帽检测的多阶段算法, 先用YOLO算法对工人所在区域进行定位, 获取工人位置信息; 然后选取工人的头部区域, 区域面积稍加扩大以提高算法容错率; 最后, 通过一个基础的神经卷积网络对是否佩戴安全帽进行判断。通过实验结果表明, 本文的方法相较之前的方法, 在安全帽识别的准确率方面达到了92.73%的效果, 在识别速度上此方法的检测速度也比之前的方法提升了一倍, 证明本文的方法能够满足期望要求。

## 关键词

深度学习, 神经网络, 安全帽检测, 生产安全, YOLO算法

# A Deep Learning Algorithm for Detecting Helmet Wearing

Xiaoyu Zhou, Chaoli Wang, Zhanquan Sun

University of Shanghai for Science and Technology, Shanghai

Received: Dec. 31<sup>st</sup>, 2021; accepted: Feb. 1<sup>st</sup>, 2022; published: Feb. 8<sup>th</sup>, 2022

## Abstract

In the process of production line, there are countless tragedies caused by workers not wearing safety helmets. In order to help factory supervisors urge workers to wear safety helmets to ensure

their personal safety, the traditional detection method often separates the safety helmets from the human body, resulting in the inability to judge the wearing relationship between people and safety helmets. Therefore, this paper proposes a detection algorithm of improved Yolo algorithm based on deep learning. Based on the existing Yolo algorithm, its loss function is improved, and replaces the original GIoU\_Loss with CIoU\_Loss, so that the recognition accuracy of the algorithm is improved and the convergence speed is faster. In view of the fact that the positioning of safety helmet by Yolo algorithm is easily affected by noise and other aspects, a multi-stage algorithm for safety helmet detection is designed in this paper. Firstly, the area where workers are located is located by Yolo algorithm to obtain workers' location information; then, the head area of the worker is selected, and the area is slightly expanded to improve the fault tolerance of the algorithm; finally, a basic neural convolution network is used to judge whether to wear a helmet. The experimental results show that compared with the previous methods, this method achieves 92.73% in the accuracy of helmet recognition, and the detection speed of this method is twice as fast as the previous methods, which proves that this method can meet the expected requirements.

## Keywords

Deep Learning, Neural Network, Helmet Detection, Production Safety, Yolo Algorithm

Copyright © 2022 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## 1. 引言

作为建筑业、制造业等工业生产中重要的劳保工具，安全帽能够有效减少头部和顶部受到的压力，保护人的头部免受侵害[1]。虽然安全帽的应用十分广泛，但是在各个建筑工地或工厂流水线加工时，大多数施工人员依然会忽视安全帽的重要性，进而因未佩戴安全帽而引发的工程安全事故频有发生。因此为了消除在日常施工现场存在的安全隐患，对施工现场进行智能化管理，对工作人员进行安全帽佩戴状况的实时检测是非常重要的。

由于在生产建筑过程中，施工场所环境十分复杂且范围一般非常广泛、作业时间长，安排人工监控安全帽的佩戴情况不仅会消耗大量人力而且往往会造成漏检的风险。

近年来，随着计算机视觉技术的迅猛发展，目标检测算法的研究备受关注。当前目标检测领域的深度学习方法主要分为两大类：两阶段式(Two-stage)目标检测算法，如 R-CNN [2]、Fast-RCNN [3]、Mask RCNN [4]等和单阶段式(One-stage)目标检测算法，如 YOLO 系列算法、SSD [5]、OverFeat [6]等。两阶段式模型具有独立提取候选区域的过程，即对输入图像可能存在待检测物体的区域进行提取，然后针对提取出来的区域，再进行位置的微调和目标分类；相反地，单阶段式模型没有独立的提取候选区域的过程，它直接由输入的图像通过卷积神经网络得到物体类别和位置信息，这种模型大大提升了检测的速度。YOLO 系列算法无论在速度还是精度上都是当前目标检测算法中的佼佼者。YOLOv5 算法在 YOLOv4 算法的基础上进行了改进，相较于 YOLOv4 在 COCO [7]数据集上得到的 43.5%的准确率，YOLOv5 的四个版本都至少有了 2.5%的提升，且检测速度更快。

运用基于深度学习的目标检测算法对施工现场的安全帽佩戴情况进行监测已然成为一种趋势。张勇等人[8]对 YOLOv3 算法进行改进，采用 DenseNet 处理低分辨率的特征层，从而提高算法的检测精度和收敛性，为了对数据集中的目标框重新聚类，还采用了 K-means 聚类算法；岳诗琴等人[9]将原本 SSD 中

的 backbone VGG-16 替换为 ResNet50 来提取特征并在附加层中引入 BN 层,其实验结果在保证检测准确率的同时加快了检测速度;庞殊杨等人[10]对 MTCNN 网络进行改进,将网络中的池化层替换为卷积层构成全卷积网络,然后引入 MobileNet 轻量级网络结构减少计算量,实现安全帽的多尺度识别。

目前安全帽佩戴情况的要求是在保证检测正确的情况下同时实现实时检测目标。本文提出的改进的 YOLOv5 算法是基于 YOLOv5s 实现的,其网络是 YOLOv5 系列中深度最小,特征图宽度最小的网络,因此 YOLOv5s 凭借较快的速度与较小的计算量在处理实时检测任务中被广泛应用。文中使用的安全帽数据集是通过网络收集安全帽和行人的图片建立而成的;第二部分详述了检测安全帽是否佩戴算法的过程包括 YOLOv5s 算法的具体流程和网络结构,并在此基础上对算法做出了优化改进;第三部分进行了改进前后的实验结果对比分析,证明了优化方法的可行性;论文最后一部分对全文进行了总结。

## 2. 安全帽实时检测算法

本文采用的方法与之前方法的区别在于不再将整个身体作为检测的目标,而是将工人的头部区域作为检测的重点,让神经网络更注重学习工人的头部特征信息。如图 1 所示,该流程具体操作如下:首先利用 YOLOv5 的预训练目标检测权重对工人进行区域定位,使用人脸检测模块对人脸进行标记,然后利用安全帽识别模块提取工人头部子区域,最后对提取到的图像采用二分类方法,判断工人佩戴安全帽的情况。

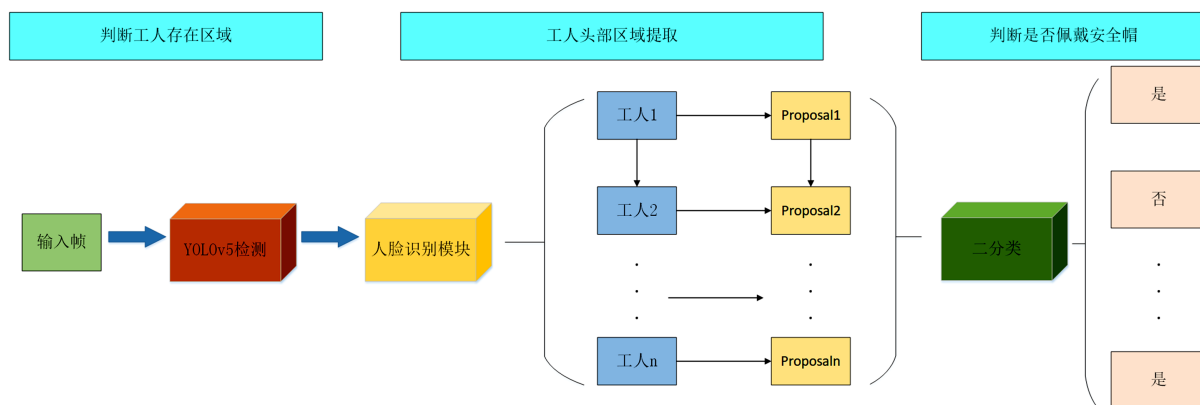


Figure 1. Flow chart of helmet wearing identification

图 1. 安全帽佩戴识别流程图

### 2.1. 判断工人存在区域

采用 YOLOv5 算法,其目标检测框架可以实现对多类目标物体的检测,训练标签使用标注工具(via 等)画出 ground truth,生成 json 文件,再通过代码完成训练过程。在本文的安全帽检测任务中,要求模型只关注工人这一目标,所以训练标签只采用人的标签,以此可以减少训练时间并提升模型效率。如图 2 所描述的就是 YOLOv5 的整体框架。网络由以下三个部分组成:

- 1) Backbone 骨干网络: 卷积神经对不同分辨率的图像进行下采样的特征提取;
- 2) Neck: 对 Backbone 提取到的特征进行上采样融合,并传递到预测层;
- 3) Output: 基于提取的特征生成边界框并预测类别。

具体地, YOLOv5 的 Backbone 区别于之前版本的地方是加入了 Focus 层如图 3,从图中可以看出其对输入图像进行了切片操作,目的是可以从高分辨率的图像中提取像素点再融合到低分辨率的图像中,扩大每个像素点的感受野,增加模型对细节特征的学习,以此可以减少计算量并加快速度;

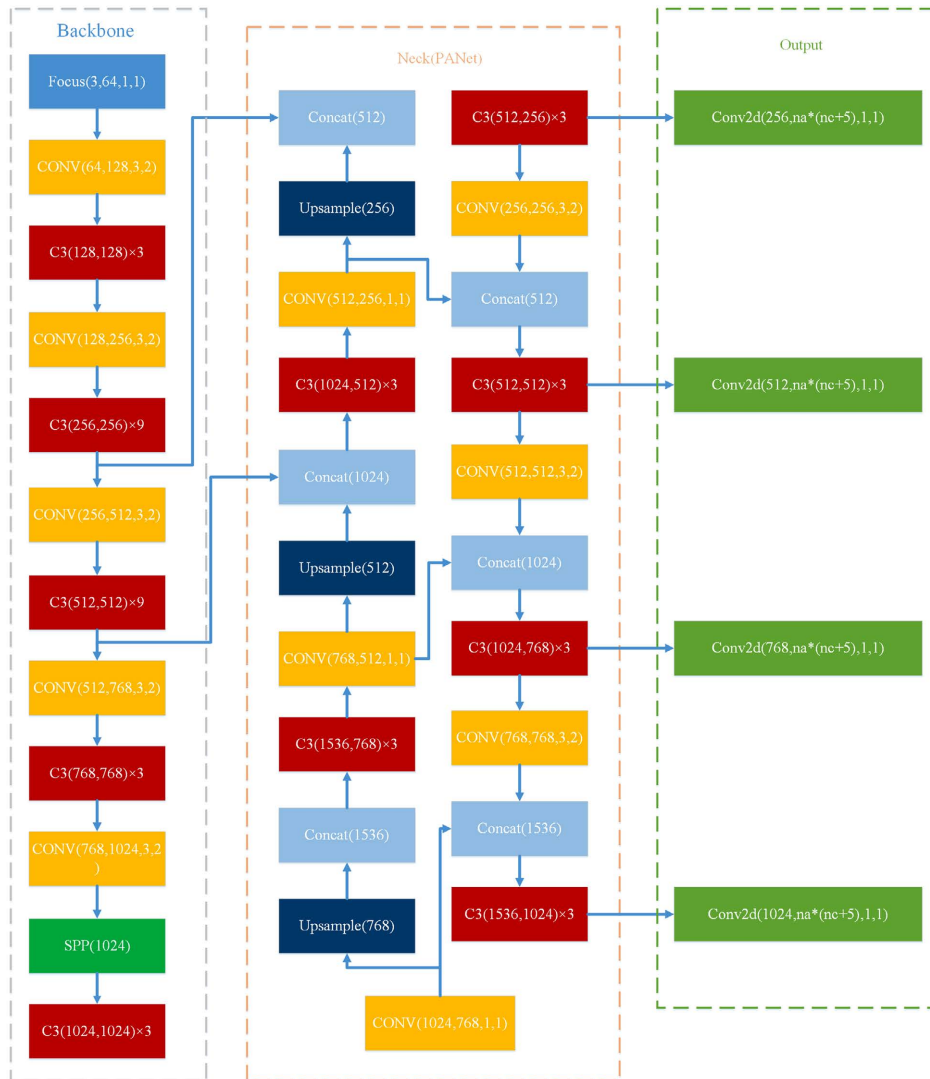


Figure 2. Overall network structure of YOLOv5  
图 2. YOLOv5 整体网络结构

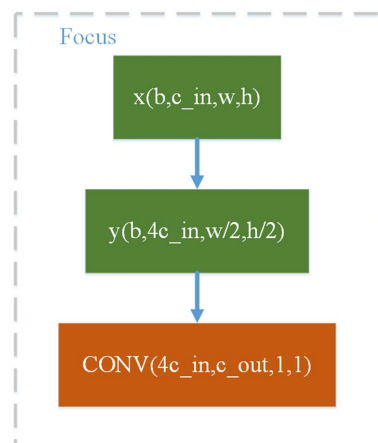


Figure 3. Focus module structure  
图 3. Focus 模块结构

Backbone 中的 C3 层结构如图 4 所示, YOLOv5 在这个结构上对比之前的版本也做出了改动, 其中 Bottleneck 采用了残差连接的思想, 以免发生网络加深后梯度消失的问题;

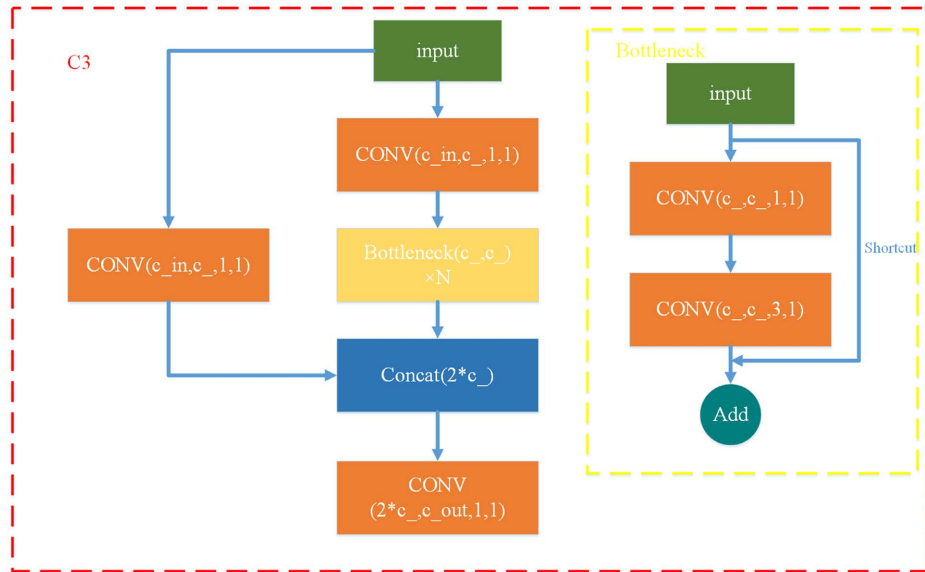


Figure 4. C3 module structure  
图 4. C3 模块结构

SPP 空间金字塔池化模块是比较经典的模块, 其结构如图 5, 分别采用了  $3 \times 3$ 、 $5 \times 5$ 、 $7 \times 7$  的最大池化, 再进行 concat 融合以此提高每个像素点的感受野;

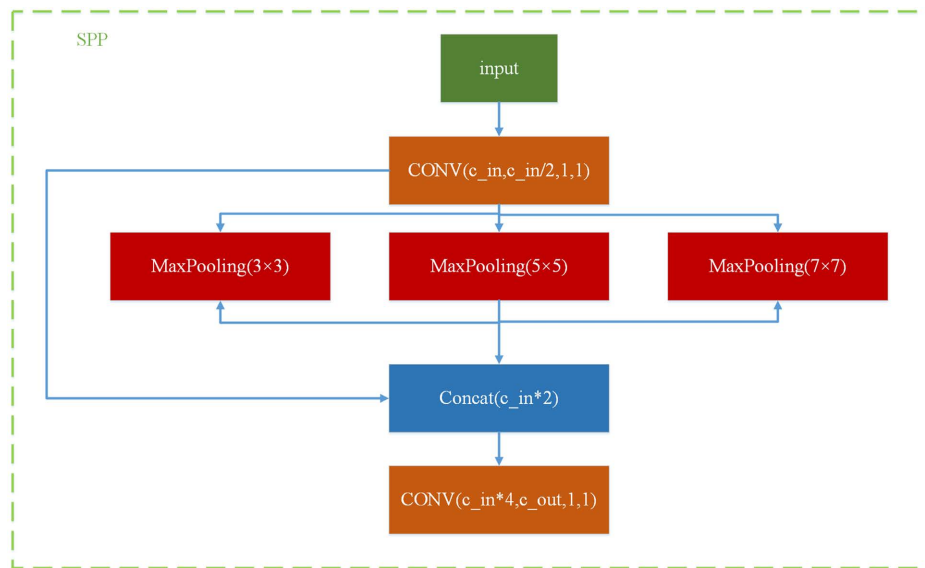


Figure 5. SPP module structure  
图 5. SPP 模块结构

Neck 部分采用了 PANet [11], 此网络是在 FPN [12] 的原有基础上继续改进, 对原本 FPN 结构存在的顶层与底层特征图因距离过远而无法有效地融合特征的问题进行了改善, 增加了从底层到顶层的信息传输路径, 再通过自适应池化, 从而实现多尺度信息的融合, 最后得到采样输出结果。

卷积模块如图 6，使用的是最简单的 2d 卷积层通过归一化再进行一次 relu 激活函数的模式。

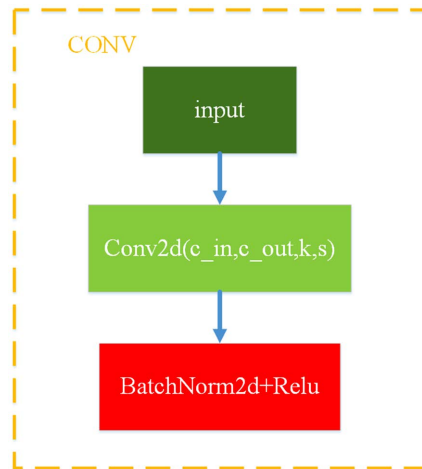


Figure 6. Convolution module structure  
图 6. 卷积模块结构

## 2.2. 工人头部区域提取

与人的动作检测不同，当检测工人是否佩戴安全帽时，头部区域是更为关键的，因此在 YOLOv5 检测出工人存在的区域后，本文要求模型对此区域进行进一步的细化提取，将分离出来的头部区域再输入到之后的网络中作后续分析，最终能够判断出是否佩戴好了安全帽。但是，在提取工人存在区域的图像时，往往提取到的区域存在大小不一的情况，为此，本文统一在各区域的中部靠上方部分提取，将提取区域固定为正方形(选定大小为  $60 \times 60$ )，其区域面积不超过工人整体区域面积，但不小于安全帽区域面积，这样可以保证提取到的区域包含住工人的头部区域，提升容错率，也不会增加网络计算量。

## 2.3. 判断是否佩戴安全帽

在提取到工人头部区域之后，接下来用二分类法完成对是否佩戴了安全帽的情况进行判断。这部分功能使用的是经典的卷积神经网络实现的。具体网络结构如图 7 所示，图像经过两次重复的  $3 \times 3$  卷积和最大池化操作后，再通过两层全连接层，最终得到分类结果。

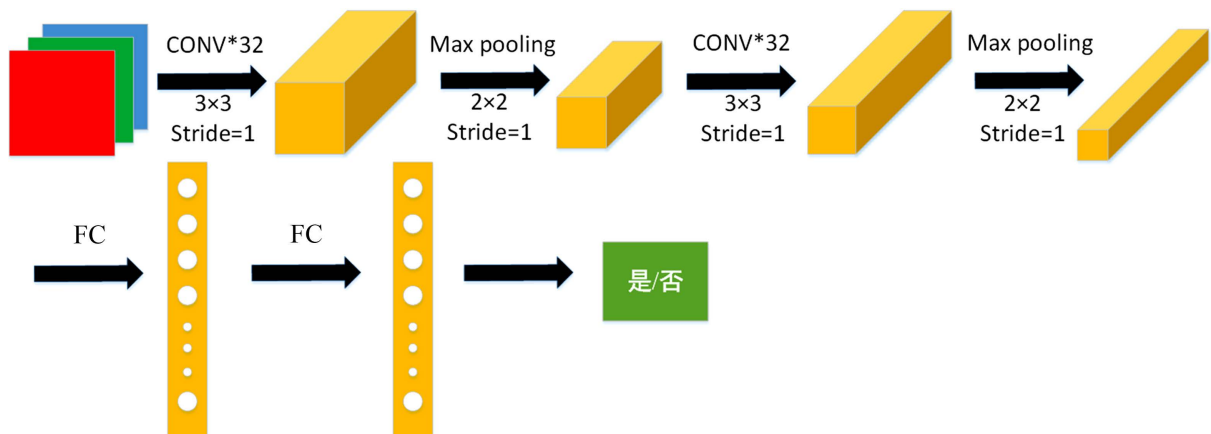


Figure 7. Helmet wearing identification network structure  
图 7. 安全帽佩戴识别网络结构

## 2.4. 算法损失函数的改进

在目标检测任务中损失函数包含了分类损失函数 Classification Loss 和 Bounding Box Loss, 对于分类损失函数采用了传统的交叉熵损失函数, 即:

$$L(\hat{y}_i, y_i) = -[y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)] \quad (1)$$

其中,  $y$  为样本的期望输出,  $\hat{y}$  为样本的实际输出。

对于回归框损失函数, 在 YOLOv5 检测算法中采用了 GIoU\_Loss [13] 损失函数, 其是对 2016 年提出的 IoU\_Loss [14] 损失函数进行了改进, IoU\_Loss 损失函数计算的是预测框与真实框的交集与并集的比, 但此种计算损失的方式存在两个问题: 一个是, 当预测框与真实框的交集为 0 时, 损失函数就不可导, 那么 IoU\_Loss 损失函数无法对预测框进行优化; 二是其无法准确地反映预测框与真实框的重合度, 如图 8 所示的三种情况, 可以看出它们的 IoU 值是相同的, 但是重合度明显不同。

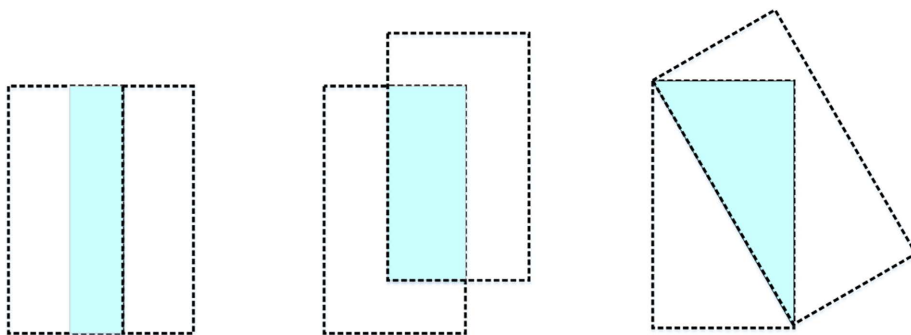


Figure 8. Three cases with different coincidence degrees, but IoU is the same  
图 8. 三种重合度不同的情形但 IoU 相同

针对这两点, 2019 年提出的 GIoU\_Loss 做出了相应改善:

定义真实框面积为  $A$ , 预测框面积为  $B$ , 那么

$$IoU = \frac{|A \cap B|}{|A \cup B|} \quad (2)$$

$$GIoU = IoU - \frac{|C - (A \cup B)|}{|C|} \quad (3)$$

其中  $C$  代表了包围  $A$ 、 $B$  的最小面积。如此, 当  $IoU$  为 1 时,  $GIoU$  为 1, 当  $IoU$  为 0 时,  $GIoU$  小于等于 0。但  $GIoU$  依然存在弊端, 如图 9 所示:

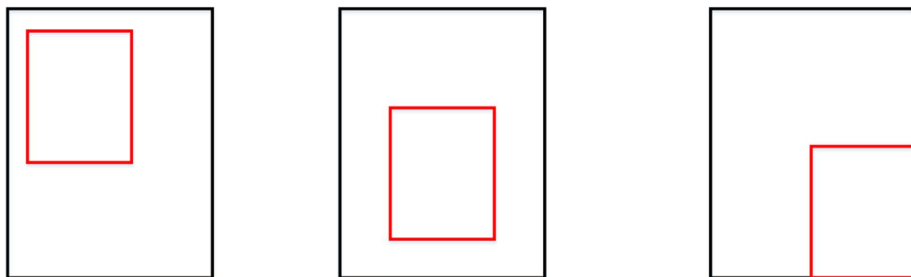


Figure 9. When the prediction box is inside the real box  
图 9. 当预测框在真实框内部

如果预测框在真实框的内部，那么预测框与真实框的差集相同，根据公式计算出的  $GIoU$  值都是相同的，那还是无法判断两个框之间的相对位置关系。

因此，本文提出改进 YOLOv5 所使用的回归框损失函数，用  $CIoU\_Loss$  [15] 代替原来的  $GIoU\_Loss$ 。 $CIoU\_Loss$  更加符合目标框的回归机制，既提出了对重叠部分的关注，主要计算真实框和预测框的重叠率和两框之间的中心点距离，又考虑到了目标框与锚框之间的尺度、距离和 bounding box 的长宽比。如图 10，黑色框代表真实框，蓝色框代表预测框，A 和 B 两点分别为其中心点，最外侧虚线红框是同时包围两框的最小矩形区域，d 表示两个中心点的欧式距离，c 表示最小矩形区域的对角线距离。

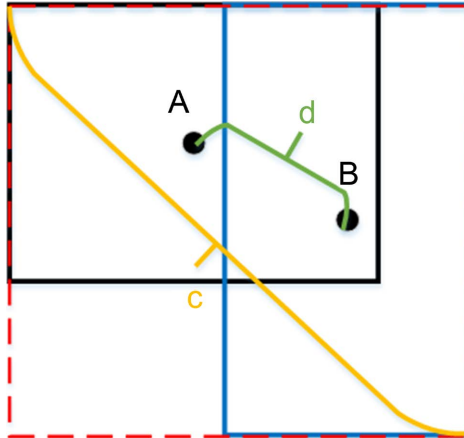


Figure 10. CIoU diagram

图 10. CIoU 图示

定义  $CIoU$  公式为：

$$CIoU = IoU - \left( \frac{d^2}{c^2} + \alpha v \right) \quad (4)$$

$$v = \frac{4}{\Pi^2} \left( \arctan \frac{w^{gt}}{h^{gt}} - \arctan \frac{w}{h} \right)^2 \quad (5)$$

$$\alpha = \frac{v}{(1 - IoU) + v} \quad (6)$$

其中， $v$  计算的是锚框和目标框之间的相似程度， $\alpha$  是权重函数，从  $\alpha$  给定的公式可以看出，损失函数的优化方式是增加真实框与目标框之间的重叠区域。

定义  $CIoU\_Loss$  公式为：

$$L_{CIoU} = 1 - IoU + \left( \frac{d^2}{c^2} + \alpha v \right) \quad (7)$$

因此， $CIoU\_Loss$  的优势就是可以直接最小化目标框与真实框之间的距离，这使得损失函数的收敛速度更快，效果也更好。

### 3. 安全帽检测算法实验

#### 3.1. 数据处理

本文使用的数据集图片均来源于网络收集，数据集包含工人头部佩戴安全帽和普通行人的图片，由



于数据集数量有限，因此在工人存在区域判断阶段，采取以下两种对数据的预处理方式：

1) Mosaic 数据增强：将四张图片进行随机缩放、裁剪、排版并拼接起来的的操作，如图 11，将四张图片合并成一张，丰富了数据集也加快了网络训练速度，并降低了模型内存需求；



Figure 11. Data enhancement process  
图 11. 数据增强过程

2) 自适应图片缩放：将原始图像固定为统一尺寸再输入到网络中去。由于先前的缩放方法存在着原图长宽不同而导致缩放后产生大量信息冗余的问题，YOLOv5 为了使缩放操作不影响算法的推理速度，先计算原始图像与输入到网络图像的缩放比例，再根据此比例缩放原始图像，最后使用黑边填充数值，以此来提升算法的推理速度。

在工人头部区域提取阶段，神经网络 CNN 可以单独训练，网络的输入端数据集是裁剪过的即进行对工人的头部区域的选取，将头部佩戴好安全帽的图片作为正样本保存，如图 12；反之为负样本，如图 13。训练时由于数据集的有限性，对数据再简单地在线处理，例如翻转、缩放、旋转等数据增强方式。



Figure 12. Partial positive sample  
图 12. 部分正样本



Figure 13. Partial negative sample  
图 13. 部分负样本

### 3.2. 参数设定

在做实验的过程中，参数的选择对实验的结果往往有着决定性的作用，因此调整超参数，并找到使模型达到最优解的参数需要不断的尝试，最终各项超参数的具体数值如表 1 所示。

**Table 1.** Hyper parameter configuration  
**表 1.** 超参数配置

超参数	数值
Learning rate	0.01
SGD momentum	0.937
optimizer weight decay	0.0005
ciou loss gain	0.05
cls loss gain	0.5
cls BCELoss positive weight	1.0
obj loss gain	1.0
obj BCELoss positive weight	1.0
iou training threshold	0.80
anchor-multiple threshold	4.0
image HSV-Hue augmentation	0.015
image HSV-Saturation augmentation	0.7
image HSV-Value augmentation	0.4
image scale	0.5
batch	16
epoch	100

学习率在深度学习中决定着模型算法能否将误差收敛到局部最小值，并且判断算法何时收敛到最小值，在实验中我们采用的学习率为 0.01，在此学习率下模型能较快收敛；优化器使用的是随机梯度下降 SGDM (Stochastic Gradient Descent with Momentum)，其每次随机选择一小部分样本而不是全部样本，用梯度下降的方法来更新模型参数，在此基础上加入了动量机制，使得优化器不会卡在梯度较小的点而影响参数的更新，实验中将动量设置为 0.937，权重衰减系数设定为 0.0005；损失函数包含三个：Classification Loss、Object Loss 和 CIoU Loss，其可调参数分别设定为 0.05、0.5 和 1.0；训练时，IoU 交并比即真实框与预测框交并比的阈值设定为 0.8，锚框倍数的阈值设定为 4；处理图像时，图像明亮度、饱和度和色相分别设置为 0.015、0.7 和 0.4，尺寸设置为原尺寸的 0.5 倍；将整个训练样本划分为 16 个批次，并对所有样本进行了 100 次迭代的训练。

### 3.3. 实验结果

本文为评估模型的效果，采取了两类实验进行对比。

实验一，使用原始的 YOLOv5 算法和改进损失函数的 YOLOv5 算法在数据集上进行对比实验，如表 2 所示，从实验结果可以看出，三个检测的评价指标都是后者性能更佳，并且后者是在比前者快一

倍的速度上达到了当前的检测水平，由此看出改进了损失函数后的 YOLOv5 算法检测效果在准确率和速度上都有明显的提升。

**Table 2.** Comparison of evaluation indexes before and after model improvement  
**表 2.** 模型改进前后的评价指标对比

方法	mAP (IoU = 0.5)	Precision	Recall	F1	epoch
YOLOv5 (GIoU)	0.8934	0.8893	0.8782	0.5600	88
YOLOv5 (CIoU)	0.9053	0.8988	0.8865	0.5603	40

其中，mAP 指模型将所有类别分类正确的平均准确率， $Precision = TP/(TP + FP)$ ， $Recall = TP/(TP + FN)$ ，(TP 为真正例，FP 假正例，FN 假反例，TN 真反例)，F1 是 Precision 和 Recall 的调和平均数。

在现实视频中的检测效果对比如图 14 所示，由检测结果可知，在同一视频的一时刻，改进后的算法对安全帽和头部的检测效果分别有了 2% 和 6% 的增长。



**Figure 14.** The two pictures show the same time (left: before improvement, right: after improvement)  
**图 14.** 两图为同一时刻(左: 改进前, 右: 改进后)

实验二，将本文提出的模型算法与其他方法进行了对比，具体结果如表 3 所示。

**Table 3.** Comparison results with various methods  
**表 3.** 与各方法的比较结果

	YOLOv4 [16]	YOLOv3 [17]	SSD	Faster R-CNN	Ours (CIoU)
Helmet mAP	92.97%	91.94%	87.09%	92.93%	93.87%
Head mAP	88.99%	78.22%	67.12%	85.09%	91.59%
mAP	90.98%	85.08%	77.11%	88.66%	92.73%

由对比结果可以看出本文提出的检测方法不论是在检测安全帽还是头部区域时，检测准确率相较于其他方法都有了显著的提升，因此本文的模型能够满足对工人安全帽实时检测的速度和精度要求。

## 4. 结束语

为检测流水线生产中工人佩戴安全帽的情况, 本文提出对安全帽的检测方案通过对 YOLO 算法的损失函数进行改进, 加快了检测速度, 提升了检测精度, 在此算法的基础上再进行多阶段的检测模式, 既融合了安全帽与人之间的特征信息又能更好地让模型专注于对安全帽佩戴的判断, 通过各组对比实验, 证明了本文提出的方法在实际操作中的可行性, 实验结果也表明本文的方法无论是在速度还是准确率上都有很大的提升。

在本文的研究中, 检测的速度与精度都达到了很好的效果, 能够取代人工监控, 因此在今后的工作中会将此算法应用到实际中, 将开发一套完善的安全帽检测系统作为工作重点。

## 参考文献

- [1] 张栋. 一个小动作、一份大责任——安全帽与责任[C]. 2020 年智慧建造与设计学术云论坛(成都)论文集. 成都: 重庆市鼎耘文化传播有限公司, 2020: 125-129.
- [2] Girshick, R., Donahue, J., Darrell, T., *et al.* (2017) Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation Tech Report (v5).
- [3] Girshick, R. (2015) Fast R-CNN. *Proceedings of 2015 IEEE International Conference on Computer Vision (ICCV)*, Santiago, 7-13 December 2015, 1440-1448. <https://doi.org/10.1109/ICCV.2015.169>
- [4] He, K., Gkioxari, G., Dollar, P., *et al.* (2017) Mask R-CNN. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, **42**, 386-397. <https://doi.org/10.1109/TPAMI.2018.2844175>
- [5] Liu, W., Anguelov, D., Erhan, D., *et al.* (2016) SSD: Single Shot MultiBox Detector. In: *Proceedings of the 14th European Conference on Computer Vision*, Springer, Amsterdam, 21-37. [https://doi.org/10.1007/978-3-319-46448-0\\_2](https://doi.org/10.1007/978-3-319-46448-0_2)
- [6] Sermanet, P., Eigen, D., Zhang, X., *et al.* (2013) OverFeat: Integrated Recognition, Localization and Detection Using Convolutional Networks.
- [7] Lin, T.Y., Maire, M., Belongie, S., *et al.* (2014) Microsoft COCO: Common Objects in Context. Springer International Publishing, Berlin. [https://doi.org/10.1007/978-3-319-10602-1\\_48](https://doi.org/10.1007/978-3-319-10602-1_48)
- [8] 张勇, 吴孔平, 高凯, 等. 基于改进型 YOLOV3 安全帽检测方法的研究[J]. 计算机仿真, 2021, 38(5): 413-417.
- [9] 岳诗琴, 张乾, 邵定琴, 等. 基于 ResNet50-SSD 的安全帽佩戴状态检测研究[J]. 长江信息通信, 2021(3): 86-89.
- [10] 庞殊杨, 毛尚伟, 贾鸿盛, 等. 基于深度学习的安全帽定位与颜色识别的方法及系统[P]. 中国, CN110188724A. 2019.
- [11] Liu, S., Qi, L., Qin, H., *et al.* (2018) Path Aggregation Network for Instance Segmentation. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, 18-22 June 2018, 8759-8768. <https://doi.org/10.1109/CVPR.2018.00913>
- [12] Lin, T.Y., Dollar, P., Girshick, R., *et al.* (2017) Feature Pyramid Networks for Object Detection. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, 21-26 July 2017, 2999-3007. <https://doi.org/10.1109/CVPR.2017.106>
- [13] Rezaatfighi, H., Tsoi, N., Gwak, J.Y., *et al.* (2019) Generalized Intersection over Union: A Metric and a Loss for Bounding Box Regression. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, 15-20 June 2019, 658-666. <https://doi.org/10.1109/CVPR.2019.00075>
- [14] Yu, J., Jiang, Y., Wang, Z., *et al.* (2016) UnitBox: An Advanced Object Detection Network. *ACM on Multimedia Conference*, Amsterdam, 15-19 October 2016, 516-520. <https://doi.org/10.1145/2964284.2967274>
- [15] Zheng, Z., Wang, P., Liu, W., *et al.* (2019) Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression. *Proceedings of the AAAI Conference on Artificial Intelligence*, **34**, 12993-13000. <https://doi.org/10.1609/aaai.v34i07.6999>
- [16] Bochkovskiy, A., Wang, C.Y. and Liao, H.Y.M. (2020) YOLOv4: Optimal Speed and Accuracy of Object Detection.
- [17] Redmon, J. and Farhadi, A. (2018) YOLOv3: An Incremental Improvement.