DOI:10.19651/j.cnki.emt.2107718

改进 YOLOv2 算法的道路摩托车头盔检测*

冉险生 陈 卓 张 禾
(重庆交通大学 机电与车辆工程学院 重庆 400074)

摘 要: 针对摩托车头盔的传统检测方法准确率低、泛化能力差和目标检测网络参数量大难以在嵌入式设备运行的问题,提出改进的 YOLOv2 的 MNXt-ECA-D-YOLOv2 目标检测算法模型。首先引入 MobileNeXt 网络替换 YOLOv2 原始骨干网络,其次在 MobileNeXt 的沙漏块中引入密集连接结构同时在网络中引入有效通道注意力机制,然后在不同深度网络层应用不同的激活函数,最后在网络输出卷积层之前增加 DropBlock 模块。采用 K-means 聚类算法重新设计了自制数据集的先验框尺寸。实验结果表明,改进后的模型相比原始 YOLOv2,在 AP_{50} 指标上提高了 3.53%,模型大小减少 77.44%,检测速度提高了近 4 倍。通过对比实验可知,改进后的 YOLOv2 模型在保持较高的精度下模型更小,在 CPU 中的推理速度更快,具有一定的应用价值。

关键词: 摩托车头盔检测;YOLOv2;MobileNetXt;有效通道注意力机制;激活函数;DropBlock
中图分类号: TP391.41;TP332 **文献标识码:** A **国家标准学科分类代码:** 520.6040

Improved YOLOv2 algorithm for road motorcycle helmet detection

Ran Xiansheng Chen Zhuo Zhang He
(School of Mechatronics and Vehicle Engineering, Chongqing Jiaotong University, Chongqing 400074, China)

Abstract: Aiming at the problems that the traditional detection methods of motorcycle helmet detection have low accuracy, poor generalization ability and large number of target detection network parameters, which are difficult to run on embedded devices, an improved MNXt-ECA-D-YOLOv2 target detection algorithm model of YOLOv2 is proposed. First, MobileNeXt network is introduced to replace original YOLOv2 backbone network, and a densely connected network structure is introduced into the sandglass block of MobileNeXt. At the same time, the effective channel attention mechanism is introduced into the network. And, different activation functions are applied at different depth network layers. Finally, DropBlock module is added before the network output convolutional layer. K-means clustering algorithm is adopted to redesign the anchor box size of self-made dataset. The experimental results show that compared with the original YOLOv2 under the same experimental conditions, the proposed method improves the AP_{50} metric by 3.53% and the model size reduced by 77.44%, and the detection speed increased by nearly 4 times. Comparison experiments demonstrate that the improved YOLOv2 has a higher average accuracy rate, a smaller model, and faster inference speed in CPU. Therefore, the proposed improved YOLOv2 model is valuable in practical applications.

Keywords: motorcycle helmet detection;YOLOv2;MobileNetXt;efficient channel attention;activation function;DropBlock

0 引 言

在全球机动车事故中摩托车驾乘人员伤亡率远大于汽车,相关研究表明其事故多以未佩戴摩托车头盔造成头部致命伤为主[1]。为此交通部门通过人工的方式来检查是否佩戴头盔,这不仅花费了大量的人力财力,而且驾驶员往往采取快速逃离的方式躲避处罚再次增加了事故危险。通过交通监控系统来自动检测摩托车驾乘人员是否佩戴摩托车头盔就显得十分必要。文献[2]根据监控摄像头采集的图像序列建立背景图像,采用背景差法以及设定颜色阈值来确定摩托车驾乘人员是否佩戴头盔。文献[3]首先使用最近邻分类器将运动对象分为摩托车以及其他运动对象,然后基于投影分析和分段分析来判断是否佩戴摩托车头盔。文献[4]使用 Haar 特征作为描述符和支持向量机分类器

收稿日期:2021-08-29
* 基金项目:重庆市科技局 2020 重庆市技术创新与应用发展专项面上项目(cstc2020jscx-msxmX0161)资助

完成对摩托车的检测，在运用霍夫变换提取头部区域，进而应用方向梯度直方图描述符提取图像特征，在使用多层神经网络分类器将目标分类从而判断是否佩戴摩托车头盔。上述是采用传统方式实现摩托车头盔检测，实践中存在设计有效特征的困难性、泛化能力差等问题。

目前基于深度学习的目标检测算法因具有较大的优势逐渐取代传统算法成为目标检测算法的主流[5]，不需人为特征设计，深度学习网络在足够的样本中自主提取有价值的特征从而完成相应的目标检测任务，同时有较强的泛化能力。基于深度学习的目标检测方法主要包括两类：一类是结合区域候选框（region proposal）和卷积神经网络（convolutional neural networks，CNN）的基于分类的 R-CNN 系列目标检测框架，称为两阶段（two stage）目标检测算法，特点是检测精度高但速度慢；另一类则是将目标检测转换为回归问题的算法，称为单阶段（single stage）目标检测算法，特点是检测速度快但检测精度低于两阶段目标检测算法。两阶段的主要研究有 R-CNN[6]、SPP-net[7]、Fast-RCNN[8]、Faster-RCNN[9] 等。单阶段主要研究有 YOLO[10]、SSD[11]。

因此，采用基于深度学习的目标检测算法实现对摩托车头盔佩戴情况的检测，以检测速度和精度较好的 YOLOv2 作为摩托车头盔佩戴检测的基础算法，并且为了解决该算法不能有效检测景深较大的小目标问题以及在移植于嵌入式设备中存在 CPU 下 Inference Time 较高和模型较大的问题，对原 YOLOv2 模型进行多种手段的改进，包括：1）使用 MobileNeXt 作为骨干网络以替换 Darknet-19，以较低模型大小实现网络的轻量化降低训练难度；2）为了使得特征信息在网络中得到充分的流动利用以实现网络特征信息的重复利用，在沙漏块中构建密集块；同时引入有效通道注意力机制（efficient channel attention），使得网络能够在不降维的条件下建立通道和其权重直接关系，以实现跨通道交互作用，从而增强网络特征的表达能力；3）激活函数有利于网络性能提升，根据激活函数的特性，在本文所构建的网络中浅层使用 ReLU6 激活函数，深层使用 h-swish 激活函；4）在网络输出结果层之前加入 DropBlock 模块，降低网络训练时过拟合情况，同时提高模型的泛化能力；5）自制摩托车头盔检测数据集，并使用 K-means 算法对所求目标聚类，得到预瞄框以提高网络训练收敛速度和网络精度，最后训练改进后的模型来实现摩托车头盔佩戴情况的检测。

1　YOLOv2 算法原理

YOLOv2 使用的是基于 Inception 的定制网络，该网络骨干模型称为 Darknet-19。YOLOv2 包括了卷积层、最大池化层、BN（batch normalization）层，整个网络主要采用 3×3 卷积核进行特征提取以及 1×1 卷积置于 3×3 卷积之间用于压缩特征，在向前传播过程中通过最大池化层实现特征图大小的缩小同时伴随着特征通道数成倍增加，经过 5 次最大池化层后特征图变为原图的 1/32 以及特征通道数由 32 变为 1 024。同时为了加速、稳定模型训练，每次卷积后使用 BN 层。YOLOv2 检测网络采用了聚类算法自动选取最佳的大小和数量的先验框（anchor boxes）。YOLOv2 还使用了一个转移层（passthrough layer），把第 13 层特征图 26×26×512 转化为 13×13×2 048 的特征图，并且与第 20 层特征图 13×13×1 024 融合成 13×13×3 072 的特征图，然后在此特征图上做卷积预测。最后该算法将图片处理为 13×13 的网格，每个网格包含了 6 大信息 t_x、t_y、t_w、t_h、conf（置信度）、classes（分类类别），本研究中类别为 2 类所以 classes 为 2。因此目标边界框如式（1）所示。

$$\begin{cases} b_x = \sigma(t_x) + c_x \\ b_y = \sigma(t_y) + c_y \\ b_w = p_w e^{t_w} \\ b_h = p_h e^{t_h} \end{cases} \tag{1}$$

式中：预测的坐标偏移值（t_x、t_y），预测框的宽高偏移值（t_w，t_h）每个网格单元的左上角坐标（c_x，c_y），先验框的宽高（p_w，p_h），σ 为 sigmoid 函数，（b_x，b_y）、（b_w，b_h）分别为边界框的中心坐标和宽高。

2　模型改进

2.1　骨干网络的改进

从轻量化模型的角度出发，引入 MobileNeXt[12] 作为新的特征提取骨干网络，替换 Darknet-19。MobileNeXt 基于沙漏型瓶颈结构，其使用模型压缩策略采用的是 Google 所提出的 MobileNets[13] 中的深度可分离卷积（depthwise separable convolution），它将标准卷积分解成深度卷积（depthwise，DW）和卷积核大小为 1×1 的标准卷积，也称为逐点卷积（pointwise，PW），使得模型的计算量和参数量大幅度的降低，具体示意如图 1 所示。其中 D_k 为卷积核的空间尺寸，M 为输入通道数，N 为输出通道数。设输入特征映射为 $D_F \times D_F \times M$，输出特征映射为 $D_G \times D_G \times N$，D_F、D_G 分别为输入和输出的特征图大小，假定 $D_F = D_G$。则可分离卷积与标准卷积的参数量以及计算量的比值分如式（2）所示。

$$\begin{cases} \dfrac{D_K \cdot D_K \cdot M + M \cdot N}{D_K \cdot D_K \cdot M \cdot N} = \dfrac{1}{N} + \dfrac{1}{D_K^2} \\ \dfrac{D_K \cdot D_K \cdot M \cdot D_F \cdot D_F + M \cdot N \cdot D_F \cdot D_F}{D_K \cdot D_K \cdot M \cdot N} = \dfrac{1}{N} + \dfrac{1}{D_K^2} \end{cases} \tag{2}$$

因此，在 D_F、N 较大的情况下，深度可分离卷积无论在计算速度还是在参数数目上都要比标准卷积有更大的优势。

MobileNetV2 的逆残差块（inverted residual block）与 MobileNeXt 的沙漏块（sandglass block）结构对比如图 2 所

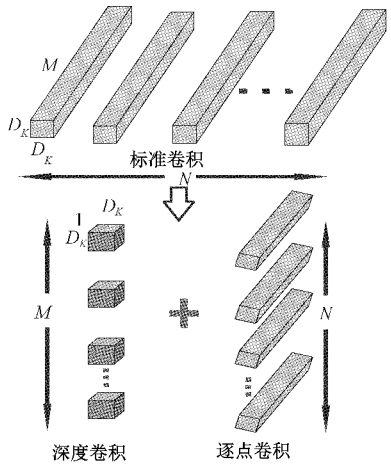


图 1　标准卷积分解过程

示。针对 MobileNetV2[14] 逆残差结构存在特征信息丢失和梯度混淆问题,调整了网络模块升维及降维的位置,原始的 MobileNetV2 采用逐点卷积升高维度再降低维度,而 MobileNeXt 中先降低维度再升高维度。假定,沙漏块输入张量为 $\boldsymbol{F}\in\boldsymbol{R}^{D_f\times D_f\times M}$,输出张量为 $\boldsymbol{G}\in\boldsymbol{R}^{D_f\times D_f\times M}$,只考虑张量经过逐点卷积的情况下,沙漏模块如式(3)所示。

$$\boldsymbol{G}=\varnothing_e(\varnothing_r(\boldsymbol{F}))+\boldsymbol{F} \tag{3}$$

式中:$\varnothing_e$ 和 $\varnothing_r$ 分别表示升维和降维的逐点卷积。

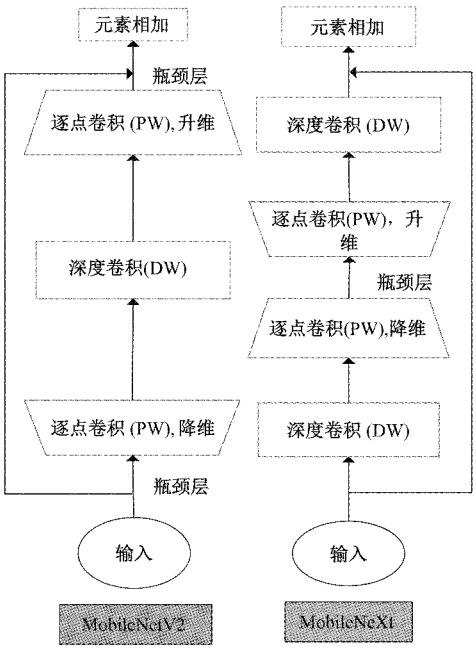


图 2　MobileNetV2 与 MobileNeXt 模块结构对比

整个 MobileNetXt 网络主体是由若干个基本的沙漏块级联组成,MobileNeXt 网络结构如表 1 所示。

表 1 中 t 表示每个沙漏块对维度的压缩率,b 表示具有相同输出维度的沙漏块重复个数。本文为了更好地运用 YOLOv2 算法,将 avgpool7×7 之前的结构作为特征提取网络以替换 Darknet-19。

表 1　MobileNeXt 网络结构

网络层数	输入维度	输出维度	t	s	b
conv2d 3×3	224×224×3	112×112×32	—	2	1
Sandglass blcok	112×112×32	56×56×96	2	2	1
Sandglass blcok	56×56×96	56×56×144	6	1	1
Sandglass blcok	56×56×144	28×28×192	6	2	3
Sandglass blcok	28×28×192	14×14×288	6	2	3
Sandglass blcok	14×14×288	14×14×384	6	1	4
Sandglass blcok	14×14×384	7×7×576	6	2	4
Sandglass blcok	7×7×576	7×7×960	6	1	3
Sandglass blcok	7×7×960	7×7×1 280	6	1	1
avgpool 7×7	7×7×1 280	1×1×1 280	—	—	1
conv2d 1×1	7×7×1 280	K	—	—	1

2.2　注意力机制与特征融合

为了提升 YOLOv2 对数据集中景深较大的小目标检测能力,多尺度特征融合对于 YOLO 目标检测算法的性能有较大的提升[15]。由此借鉴密集连接网络 DenseNet[16] 特征融合方式,密集块(dense block)和转换层(transition layers)是其基本组成单元。DenseNet 中密集块是主要的特征提取模块,其最大的特点是将每层的输出拼接起来作为下一层的输入,促使网络层级间的特征信息的最大流动,保证特征的重复利用,以实现多层特征的融合,如式(4)所示。

$$x_l=H_l([x_0,x_1,\cdots,x_{l-1}]),\ l=0,1,2,3,4 \tag{4}$$

式中:$H_l(*)$ 指的是 BN 层、激活函数以及卷积核大小为 3×3 的卷积级联组合;$[x_0,x_1,\cdots,x_{l-1}]$ 指代的是 0 层到 $l-1$ 层所分别产生的特征图的拼接,并将其作为 l 层的输入,然后通过 $H_l(*)$ 变换作为第 l 层的输出结果,通过转换层以实现特征图尺寸以及特征通道数的降低。

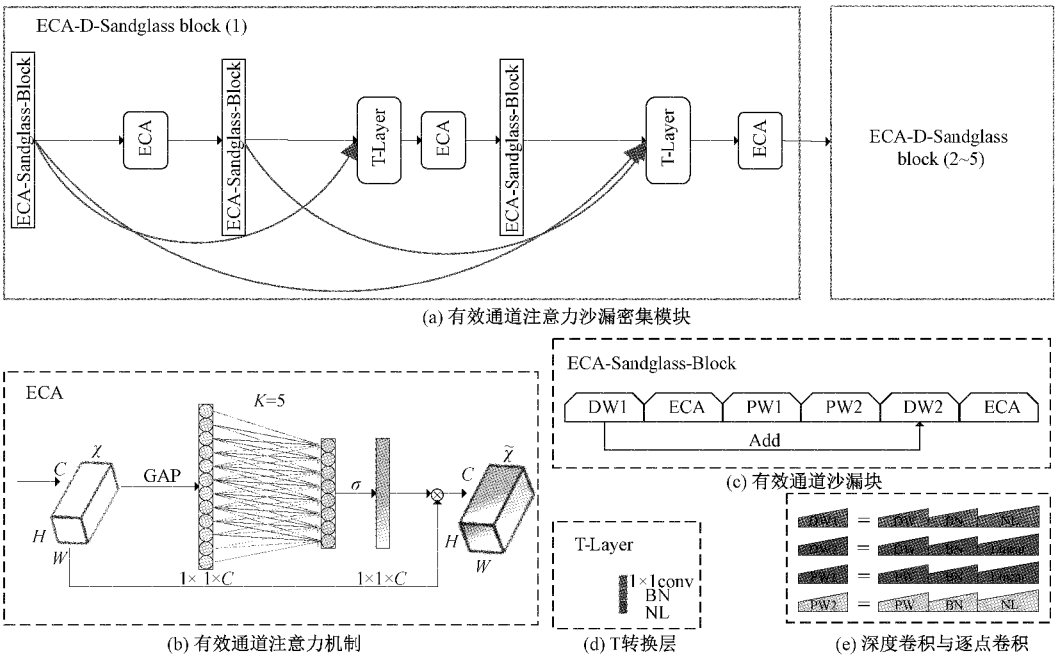
目前已证明注意力机制能极大的提升 CNN 网络性,在各类主流目标检测算法中得到了广泛应用[17-18]。考虑到本文中网络通道数变化较大,而特征提取网络对每个特征通道会采用相同的处理方式,这对于网络的检测性能带来了一定的限制。目前运用较多的 CBAM 注意力机制虽然同时基于通道和空间,但是它们计算过程是相互独立的,随之会使计算时间和计算量都大幅增加。因此在改进网络的特征提取网络后再引入有效通道注意力机制(efficient channel attention,ECA)。而有效通道注意力最突出的特点是避免降维以高效率的方式实现跨通道的交流,同时在增强网络特征表达能力时也能降低模型的复杂度[19-20]。ECA 模块如图 3(b)所示。首先输入特征图 χ,该特征图的所用通道经过全局平均池化层后,在通过一个可以共享权重的快速一维卷积进行特征学习,在特征学习过程中 ECA 注意到每个通道以及 k 个近邻来捕获局部跨通道交互,然后跟着一个 sigmoid 函数来获得对应通道的概率,再将其乘以原始的输入特征作为下一层的输入。超参数 K 为一维卷积的卷积核尺寸,其大小代表局部跨通道交互的覆盖

率。通过自适应的方式可以确定 K 值，其大小由其与通道维数 C 的正比关系得到，如式(5)所示。

$$K=\psi(C)=\left|\frac{\log_2(C)}{\gamma}+\frac{b}{\gamma}\right|_{\text{odd}} \tag{5}$$

式中：$\gamma=2$，$b=1$，$|*|_{\text{odd}}$ 表示最邻近的奇数，C 为通道维数。

考虑于此，本文再改进后的特征提取网络中同时引入密集网络和有效通道注意力机制。首先在骨干网络的沙漏模块引入 ECA 模块称为 ECA-Sandglass block，如图 3(c)所示，同时再普通卷后也引入 ECA 模块，以实现通道间充分的信息交流增强网络的性能。本文改进特征融合方式为，将表 1 中输出维度相同的沙漏块作为密集块，因此总共有 5 个密集块，在引入 ECA 模块后每个密集块中 ECA-Sandglass block 模块个数为 3、3、4、4、3。由于密集块中大量的特征通道的拼接使得网络的特征维度较大，将导致网络所需的计算量较高，对此本文借鉴 NIN[21] 的思想，引入 1×1 卷积层和 BN 层以及非线性的激活函数(NL)置于每次拼接特征通道后，称为 T-Layer，以此在不降低改进网络性能前提下，实现了特征维度的降低进而也降低了计算量。综上，本文基于有效通道注意力机制和密集网络构建了有效通道注意力机制的沙漏密集模块，称为 ECA-D-Sandglass block，如图 3(a)所示。



(a) 有效通道注意力沙漏密集模块　(b) 有效通道注意力机制　(c) 有效通道沙漏块　(d) T转换层　(e) 深度卷积与逐点卷积

图 3　基于有效通道注意力机制的沙漏密集模块

2.3　激活函数

激活函数是深度学习的核心单元，即使激活函数只有少量的提升，但它也会因为大量的使用而获得极大的收益。YOLOv2 中为了避免出现梯度弥散现象，使用的是 LeakyReLU 激活函数。为了避免低精度的移动端设备带来精度损失，MobileNeXt 采用的是对 ReLU 做最大输出值为 6 限制的 ReLU6 激活函数。考虑到非线性的瓶颈层会带来特征信息的损失，因此只在沙漏块的第 1 个深度卷积以及最后一个逐点卷积中添加 ReLU6，为了提高网络的分类性能在最后一个深度卷积采用的也是线性变换。本文为了进一步提高网络目标检测性能，引入了新的激活函数 Swish[22]，和 ReLU 一样，无上界有下界。与 ReLU 不同的是，Swish 是平滑且非单调的函数。Swish 函数定义如式(6)、(7)所示。

$$\text{Swish}(x)=x\cdot\sigma(x) \tag{6}$$

$$\sigma(x)=(1+\exp(-x))^{-1} \tag{7}$$

虽然 Swish 函数能够随着网络深度的增加有效地提高网络的精度，但是计算量较大，因此采用一个用近似函数逼近 Swish，称为 h-swish[23]，其定义如式(8)所示。

$$\text{h-swish}(x)=x\frac{\text{ReLU6}(x+3)}{6} \tag{8}$$

h-swish 与原始的 Swish 相比，性能相当并且降低了计算量。根据 h-swish 激活函数的特性，本文提出在网络层数较多时，浅层网络使用 ReLU6 激活函数，深层网络使用 h-swish 激活函数，具体每层网络使用 ReLU6 还是 h-swish，本文在网络改进了骨干网络以及引入了密集块和 ECA 后，通过实验确定激活函数在网络中分布情况，如图 4 所示。

本实验根据 h-swish 激活函数特性，以降低验证所需花费时间以及实验难度的基础上，从网络最后一个 ECA-Sandglass block 中由深层到浅层逐渐添加 h-swish。图 4(a)、(b)中横坐标 1～8 分别表示从网络深层到浅层添加 h-swish，即 ECA-Sandglass block、ECA-D-Sandglass block (2～6，共 5 个)、ECA-Sandglass block、ECA-Sandglass block，总共 8 处。每处为分界点，其前面网络层都使用

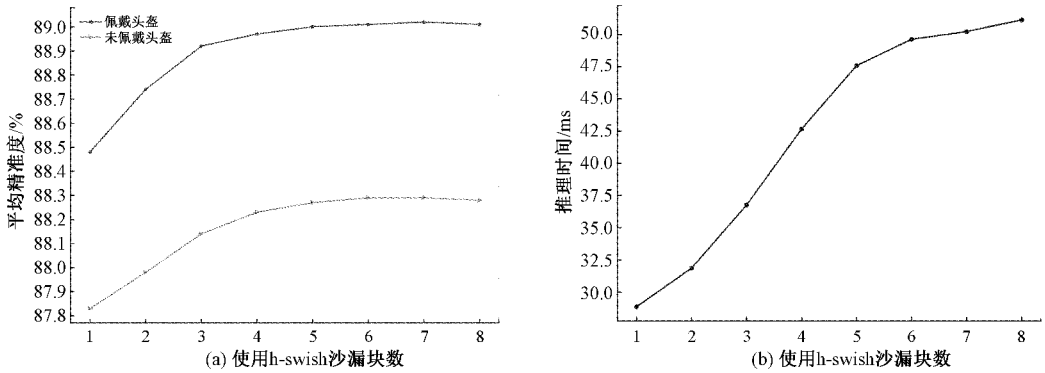


图 4　网络性能与 h-swish 的应用层数关系

ReLU6,其后则都为 h-swish。由图 4(a)可知,随着 h-swish 激活函数引入网络的平均检测精度不断增加,4 处之前增幅明显,随后 4～6 处增幅降低,7～8 处相比之前平均检测精度有下降的趋势。图 4(b)反映了网络的推理时间随着应用 h-swish 层数的增加而增加。考虑检测精度与推理速度之间的平衡关系,提出在将第 4 处作为分界点,即从第 3 个沙漏密集块处都使用 h-swish。

2.4　DropBlock

通常,将 Dropout 被广泛的运用到全连接层中,但是对于卷积层几乎不起作用。因为卷积层的特征图中相邻位置的空间语义信息是共享的,而 Dropout 主要通过对某个神经元进行中断,所以仅仅中断部分神经元信息仍然可以传递到下层网络中。所以笔者引入 DropBlock[24]正则化方法,不同于 Dropout 中断某一个神经元,DropBlock 是随机的将特征图中相邻区域单元一起中断。通过该方式,将保留下来的神经元能够集中的学习特征图中的其他信息,提高网络的泛化能力和识别被遮挡目标的能力,也在一定程度上降低了计算量。DropBlock 有两个重要的参数:γ 和 b_s。b_s 表示特征图中进行归零的区域大小,此处设置为 5,表示大小 5×5 的区域置零;γ 表示中断神经单元的个数,该大小控制着每个特征图中有多少通道进行 DropBlock,该参数由式(9)中计算确定。

$$\gamma = \frac{1-k_p}{b_s}\frac{f_s^2}{(f_s-b_s+1)^2} \tag{9}$$

式中:f_s^2 为输入特征图的大小;$(f_s-b_s+1)^2$ 为经过 DropBlock 后的特征图大小;k_p 为特征图中每个神经单元被保留的概率,该值大小会影响网络的精度。在训练过程中,本文发现固定大小的 k_p 值对网络几乎不起作用,因此采用与训练步数成线性负相关方式将 k_p 从 1 降到 0.80。

3　MNXt-D-YOLOv2 网络结构

结合上述对 YOLOv2 所提出的该改进方法,本文构建了如表 2 所示的 MobileNeXt-Efficient Channel Attention-Dense-YOLOv2(MNXt-ECA-D-YOLOv2)整体网络结构。

表 2　MNXt-ECA-D-YOLOv2 网络结构

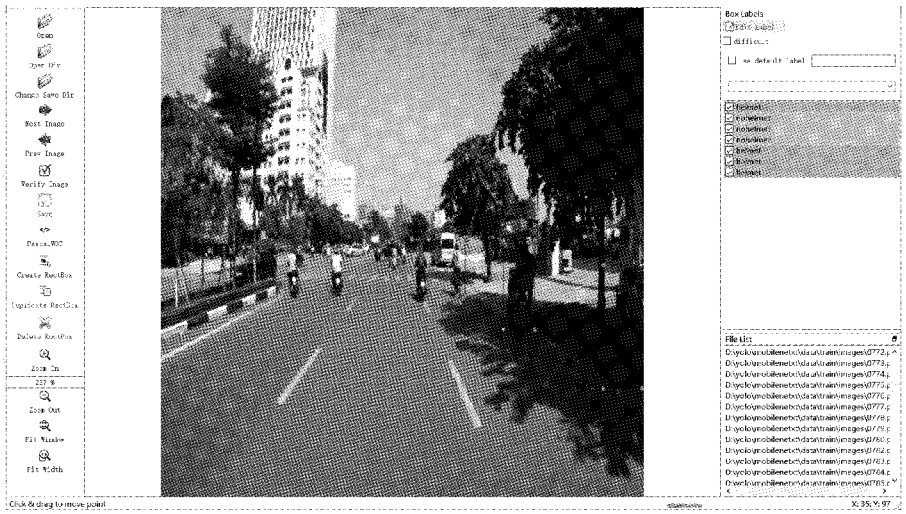
网络层数		输入维度	输出维度	t	步长	NL
conv2d+ECA	—	416×416×3	208×208×32	—	2	RE
ECA-Sandglass block	—	208×208×32	104×104×96	2	2	RE
ECA-Sandglass block	—	104×104×96	104×104×144	6	1	RE
ECA-D-Sandglass block 1	ECA-Sandglass block (1)	104×104×144	52×52×192	6	2	RE
	ECA-Sandglass block (2～3)	52×52×192	52×52×192	6	1	RE
ECA-D-Sandglass block 2	ECA-Sandglass block (1)	52×52×192	26×26×288	6	2	RE
	ECA-Sandglass block (2～3)	26×26×288	26×26×288	6	1	RE
ECA-D-Sandglass block 3	ECA-Sandglass block (1)	26×26×288	26×26×384	6	1	HS
	ECA-Sandglass block (2～4)	26×26×384	26×26×384	6	1	HS
ECA-D-Sandglass block 4	ECA-Sandglass block (1)	26×26×384	13×13×576	6	2	HS
	ECA-Sandglass block (2～4)	13×13×576	13×13×576	6	1	HS
ECA-D-Sandglass block 5	ECA-Sandglass block (1)	13×13×576	13×13×960	6	1	HS
	ECA-Sandglass block (2～3)	13×13×960	13×13×960	6	1	HS
ECA-Sandglass block	—	13×13×960	13×13×1 280	6	1	HS
conv2d+ECA	—	13×13×1 280	13×13×1 280	—	1	HS
Dsparableconv2d+ECA	—	13×13×1 024	13×13×1 024	—	1	HS
Dsparableconv2d+ECA	—	13×13×1 024	13×13×1 024	—	1	HS
DropBlock	—	13×13×1 024	13×13×1 024	—	—	—
conv2d 1×1	—	13×13×1 024	13×13×35	—	1	—

表 2 中 RE 和 HS 分别表示非线性激活函数 ReLU6、h-swish；conv2d+CAE 表示标准卷积后级联 ECA 模块，深度可分离卷积+CAE 的指的是的深度可分离卷积级联 ECA 模块。

4　制作数据集与确定先验框

样本的获取：通过网络收集正常天气条件下实际道路中白天摩托车行驶视频流，再将其处理为图片格式。

样本的标注：结合实际情况只考虑摩托车驾驶员是否佩戴头盔，将摩托车以及摩托车驾驶员视为一个整体，因此，只需标注佩戴摩托车头盔和不佩戴摩托车头盔这两类别，使用 labelImg 标注出需检测目标的位置以及所属类别并生成 VOC 格式的文件，如图 5 所示。



(a) labelImg标准界面

```
<size>
        <width>416</width>
        <height>416</height>
        <depth>3</depth>
</size>
<segmented>0</segmented>
<object>
    <name>helmet</name>
    <pose>Unspecified</pose>
    <truncated>0</truncated>
    <difficult>0</difficult>
    <bndbox>
        <xmin>259</xmin>
        <ymin>154</ymin>
        <xmax>312</xmax>
        <ymax>252</ymax>
    </bndbox>
</object>
...
```

(b) VOC文件

图 5　样本标注

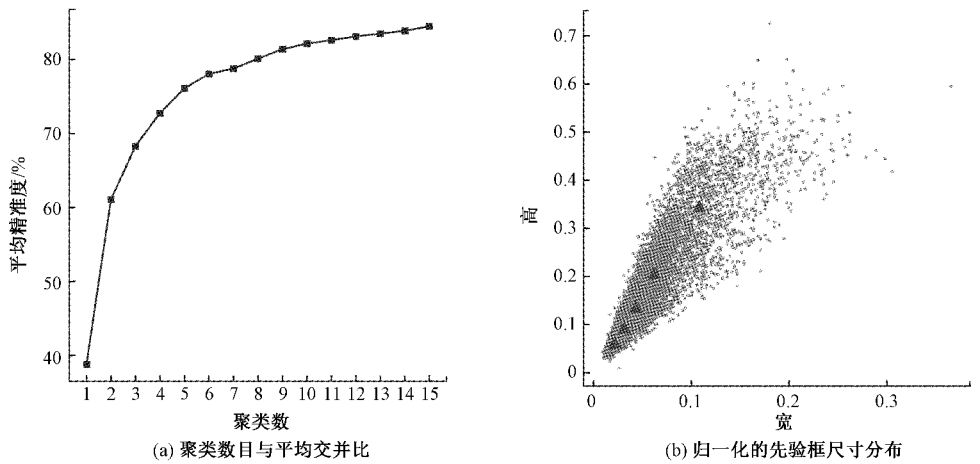
总共标注了 3 411 张图片，其中训练集有 2 921 张，测试集有 490 张。在整个数据集中，标注出的目标总数为 13 136 个，该数据集称为 motorcycle helmet[25] 数据集，其具体分配如表 3 所示。

表 3　数据集分配结果

类别	总数	训练总数	验证总数
佩戴摩托车头盔	7 004	6 139	865
未佩戴摩托车头盔	6 132	5 483	649

合理的先验框尺寸有利于目标框回归和准确率的提高。YOLOv2 根据 COCOC 数据集和 PASTCAL VOC 分别设定了 5 个先验框，但是考虑到 motorcycle helmet 数据集与两者数据集都存在一定的差异，为了提高特定检测任务的精度，使用 K-means[26] 聚类算法重新聚类求解出先验框尺寸。不同聚类数目对应的平均 IOU 和归一化后先验框的尺寸分布如图 6 所示。

由图 6(a)可知，聚类数目与 IOU 值成正相关的，当聚类数目为 5 时，平均交并比为 76.63%，但随后的涨幅不断



(a) 聚类数目与平均交并比　　(b) 归一化的先验框尺寸分布

图 6　先验框的确定

降低。为了计算效率和检测精度的平衡,选择产生的 5 个先验框。在图 6(b)中,三角形状所处位置为 5 个先验框归一化后的聚类尺寸,将归一化后的先验框还原为最终特征图相对应的大小,最终确定先验框尺寸由如下 5 组:(0.812 5,2.687 5)、(0.562 5,1.781 25)、(0.281 25,0.812 5)、(1.406 25,4.5)、(0.406 25,1.218 75)。

5　实验分析与可视化

5.1　模型训练环境

实验平台主要由两部分组成,其中硬件平台主要包括:Intel-Core i7-8750H CPU @ 2.21 GHZ; NVIDIA GeForce GTX 1060 显卡;16 G 内存。软件环境包括:64 位 Windows10 操作系统;Tensorflow2.0 深度学习框架构建网络模型;Pycharm Community IDE;采用 CUDA10.2、CUDNN8.0 对 GPU 加速;Python3.7.6。

训练过程使用 Nadam[27] 作为优化器,初始学习率为 0.01,权重衰减设置为 0.000 1,动量为 0.9,batch size 设置为 8。本文中所有实验均使用单尺度训练,图像的输入大小为 416 pixel×416 pixel。

5.2　Loss 可视化分析

使用 motorcycle helmet 数据集从零开始训练 MNXt-D-YOLOv2 网络,最终得到能够自动检测摩托车头盔的网络模型。如图 7 所示为 MNXt-ECA-D-YOLOv2 网络训练时的损失值变化曲线。

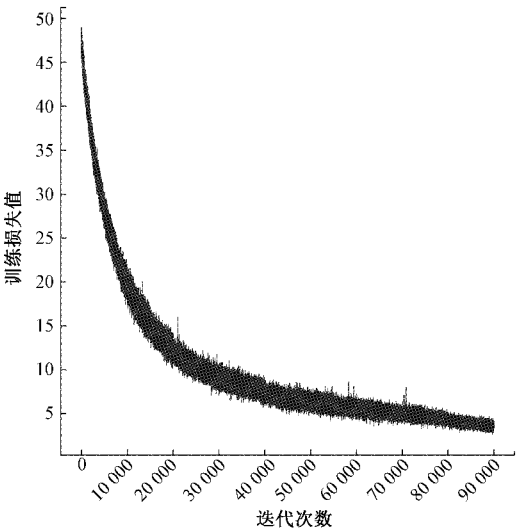


图 7　训练过程损失曲线

模型训练初期的损失值以较快的方式下降,一方面得益于 Nadam 优化器加速梯度更新,另一方面 ECA-Sandglass block 以及密集网络组合加强了梯度反方向的传播。随着迭代次数的增加,模型参数更新频率不断降低,损失值逐渐趋于平稳。为了防止训练过程中模型出现过拟合现象引入早停(early stopping)机制,迭代 90 000 次之后,停止训练,得到训练好的模型。

5.3　评价指标

混淆矩阵(confusion matrix)作为传统机器学习算法中的重要评价标准,目标检测中的精确率(Precision)和召回率(Recall)都可以通过混淆矩阵求得。如图 8 所示的混淆矩阵,其纵轴为模型预测类别,横轴为真实标签类别。P(Positives)和 N(Negatives)为模型预测的类别,T(True)和 F(False)用于评价模型的判断结果是否正确。

	T	F
P	True Positives (TP)	False Positives (FP)
N	False Negatives (FN)	True Negatives (TN)

图 8　混淆矩阵

由此可得 4 个重要的评价因子分别为:1)TP:模型正确识别为正样本;2)FP:模型将负样本识别为正样本;3)FN:模型将正样本识别为负样本;4)模型正确的将负样本识别为负样本。

引入公式精确率、召回率以及 AP_{50},如式(10)~(12)所示。

$$\text{Precision} = \frac{TP}{TP + FP} \tag{10}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{11}$$

$$AP_{50} = \int_0^1 P(R)\mathrm{d}R, \text{IOU} > 0.5 \tag{12}$$

式中:Precision 衡量模型在目标检测中的准确度;Recall 体现了模型的识别查全能力;IOU 为模型预测的目标区域与真实标定区域的交并面积比,其衡量模型的目标定位能力。由于本文中存在戴头盔和未佩戴头盔两种类别,因此在求得模型对每个类别的预测 AP_{50} 精度后,在对各类别 AP_{50} 求和再求其平均即得 mAP_{50}。目前 AP_{50} 是目前评价主流目标检测算法性能的最重要指标之一。

5.4　本文提出改进模型的消融实验

本文提出的改进 YOLOv2 摩托车头盔检测算法,包含了 4 个改进,分别是:将去除分类输出层的 MobileNeXt 作为 YOLOv2 的骨干网络,引入密集网络和有效通道注意力机制 ECA,不同深度的网络层采用不同激活函数,引入 DropBlock 模块。为了验证本文对 YOLOv2 算法改进的有效性,本文在自制的 motorcycle helmet 数据集上,对每个改进方法进行的消融实验。在本实验配置的条件下测试模型的在 CPU 中推理时间(inference time),主要有两点原因:1)本文所改进的模型主要考虑将其运用到资源有限的嵌入式设备中;2)GPU 与 CPU 对于数据处理方式不同,GPU 能够并行处理大规模的矩阵内积数据运算,而 CPU 则倾向于对数据串行运算。当 GPU 的显存足够大时,那么网络每层的计算都可以一次处理,此时总的运行时间主

要由网络层数决定，相反 CPU 对于数据处理所需的时间主要来自模型总的计算量。本文所提出的模型使用了大量深度可分离卷积，减少了总的参数量和计算量，但是相应层数增多，因此使用 CPU 运行优化后的模型更加合理。本文对改进方法的消融实验结果如表 4 所示。表中 mAP_{50} 表示两类检测结果 AP_{50} 的平均精度均值。

表 4　改进 YOLOv2 的消融实验结果

模型	YOLOv2 及其改进模型	AP_{50}/%		mAP_{50}/%	ΔmAP_{50}/%	推理时间/(ms,CPU)	模型大小/MB
		佩戴头盔	未佩戴头盔				
A	YOLOv2	87.22	86.59	86.91	—	170.80	194.48
B	A+MobileNeXt	87.06	86.34	86.70	−0.21	19.74	28.47
C	B+DenseNet&ECA	89.65	88.92	89.29	2.59	22.86	43.94
D	C+h-swish	90.55	89.71	88.83	0.84	42.26	43.94
E	D+DropBlock	90.87	90.00	90.44	0.31	41.32	43.94

由表 4 可知，YOLOv2 引入 MobileNeXt 作为主干网络的模型 B 相比原始的 YOLOv2 大大提高了检测速度和模型大小大幅降低；在引入密集网络和有效通道注意力机制后模型 C 虽然较模型 B 检测速度虽有降低以及模型大小有所增加，但是模型 mAP_{50} 提高了 2.59%；模型 D、E 在模型 C 的基础上依次增加的 h-swish 激活函数和 DropBlock 模块，在检测速度和模型大小基本保持的条件下，检测精度不断提高。

综上，本文所提出的 MNXt-ECA-D-YOLOv2 网络，在自制的数据集上 mAP_{50} 达到了 90.44%，在 CPU 条件下的推理时间为 41.31 ms 达到了实时性的要求，模型大小为 43.94 MB；与 YOLOv2 相比，mAP_{50} 提高了 3.53%，检测速度提高了近 4 倍，模型大小减少 77.44%。

5.5　本文提出算法与现有算法的对比

为了更好地验证本文算法的有效性，将具有代表性的二阶段和一阶段目标检测算法与本文提出的算法作对比。二阶段算法选择了精度较高的 Faster R-CNN，一阶段算法选择了 YOLOv3、Tiny-YOLOv3、YOLOv4。在相同的实验配置下，在自制的数据集上实现了所选用的算法，实验结果如表 5 所示。

表 5　各类算法对比结果

算法	AP_{50}/%		mAP_{50}/%	推理时间/(ms,CPU)	模型大小/MB
	佩戴头盔	未佩戴头盔			
Faster R-CNN	91.60	91.32	91.46	962.35	146.30
YOLOv4	92.13	91.64	91.89	264.27	265.58
Tiny-YOLOv3	74.67	74.28	74.47	86.15	34.56
YOLOv3	88.47	87.86	88.16	238.21	245.20
MNXt-ECA-D-YOLOv2(本文)	**90.87**	**90.00**	**90.44**	**41.32**	**43.94**

由表 5 可知，本文所提出的算法在模型检测精度、模型大小以及检测速度上均优于 YOLOv3；相比 Tiny-YOLOv3，虽然模型大小稍高，但平均精度均值远高于 Tiny-YOLOv3，检测速度也快一倍多。相比 YOLOv4 和 Faster R-CNN，本文的算法在平均检测精度均值上有些许差距，但是在模型大小仅为 YOLOv4 的 1/6、Faster R-CNN 的 1/3，检测速度为 YOLOv4 的 6 倍、Faster R-CNN 的 23 倍。由于本文所提出的算法主要考虑应用到道路监控等嵌入式边缘计算设备中，所以在检测精度较高的条件下，更小的模型尺寸、更高检测速度的模型更能满足实际要求。

5.6　本文方案与现有摩托车头盔检测方案对比

目前对于摩托车头盔检测的实现方式主要分为两大类，分别是机器学习和深度学习。文献[28-29]采用机器学习方式实现摩托车头盔佩戴检测，其检测步骤主要分为：1)对视频或者图片中摩托车的识别；2)在确定为摩托车区域中设置兴趣区域(ROI)，在对该区域进一步特征分析以确定驾乘者是否佩戴摩托车头盔。文献[30]采用与本文相同的深度学习的方式。如图 9 所示代表了目前摩托车头盔检测方案的流程。

在相同实验条件下，分别采用了与文献[28-30]相同的方式进行摩托车头盔佩戴检测，得到了检测速度与检测精度的关系图，如图 10 所示。就检测精度上，本文稍逊于基于传统机器学习方式实现的摩托车头盔检测精度，但在检测速度上完全超过了文献[28-29]所使用的检测方式。一方面由于采用机器学习的方式实现目标检测，往往需要确定检测区域然后在进行相应的特征提取最后在通过分类器，由此带来了推理时间消耗。另一方面，本文对原始的

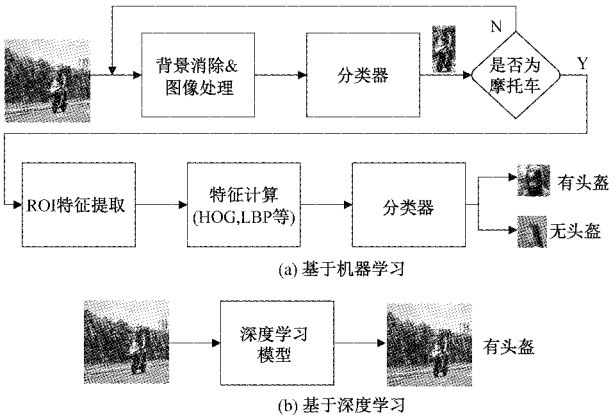


图 9 不同方案的摩托车头盔检测

YOLOv2 算法进行的模型上的轻量化,也是使得检测时间得到一定的降低。本文与文献[30]都以深度学习作为摩托车头盔检测模型,但是所提出的 MNXt-ECA-D-YOLOv2 模型在检测精度、检测速度以及模型大小都超过了文献[30]所使用的 RetinaNet-50 模型。

综上所述,MNXt-ECA-D-YOLOv2 模型在摩托车检测方案上相比机器学习,省去了手工特征设计的复杂,通过基于端到端的 YOLO 算法的深度学习极大地简化了摩托车头盔检测方案的设计,在不失去检测精度的条件下也极大地提高了检测速度。同样,MNXt-ECA-D-YOLOv2 模型的各方面的检测性能都优于基于 RetinaNet-50 的检测方案。考虑于此,本文所构建的改进 YOLOv2 模型,在检测方案设计的复杂性、检测精度以及速度上,都得到了很好的权衡。

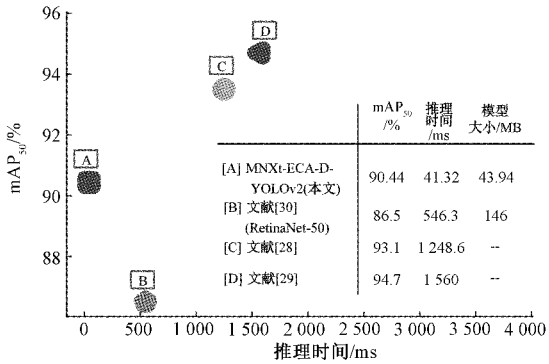


图 10 不同摩托车头盔检测方案速度与精度的对比

5.7 摩托车头盔佩戴检测效果可视化

随机选择 4 张测试集样本检测的可视化结果如图 11 所示,第 1 列为原始样本,第 2 列和第 3 列分别为 YOLOv2 和 MNXt-ECA-D-YOLOv2 的可视化检测结果。两模型对检测出的物体都能正确识别,但是通过对比可以直观地看出,对于图中景深较大的小目标 YOLOv2 不能有效的检测出来,而本文算法能够很好地检测出来并识别正确,因此可以定性得出本文提出的算法在检测精度上有更好的效果。

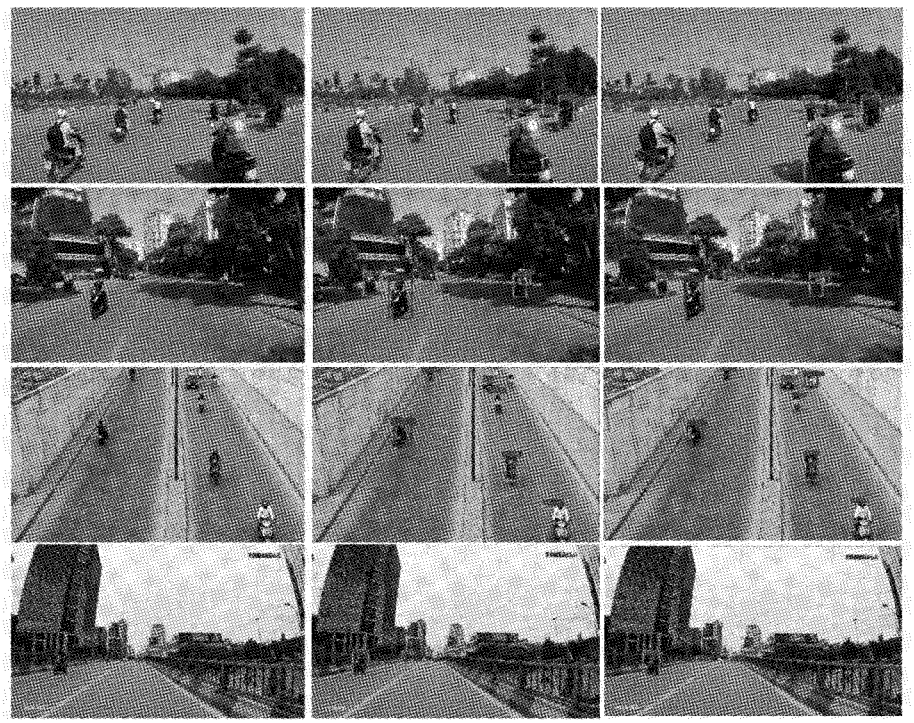


图 11 YOLOv2 与 MNXt-D-YOLOv2 检测结果

6　结　　论

本文提出了一种可用于嵌入式设备中的道路摩托车头盔佩戴检测算法：MNXt-ECA-D-YOLOv2，该算法从骨干网络的轻量化、有效通道注意力与特征融合、不同深度网络应用不同激活函数、网络的正则化等 4 个方面对原始的 YOLOv2 进行了改进。为了降低模型大小和计算量，引入轻量化的骨干网络以替换原始骨干网络，但是带来了网络检测精度的降低，对此在网络中构建密集网络以实现特征信息在网络中的最大流动，提高对小目标检测的敏感性，再引入有效通道注意力机制实现跨通道交互作用增强特征表带能力，通过密集网络和 ECA 促使网络能够充分学习并能利用更有用的特征信息极大的提高了网络的检测精度；不同激活函数在不同深度层的应用也进一步提高了网络模型的表达能力，应用正则化模块 DropBlock，也为网络带来了微小的增益。在同样的实验条件使用自制的数据集进行训练测试，实验得出了在模型大小、检测速度以及检测精度上都优于原始 YOLOv2 模型。同时笔者将所提出的模型与 YOLOv3、Tiny-YOLOv3、Faster R-CNN 以及 YOLOv4 目前较先进的模型算法对比可知，在保持较高的检测精度条件下，模型大小以及检测速度都优于所对比的模型；将该模型运用到嵌入式设备中成为可能，为不需人为道路摩托车头盔佩戴检测等应用提供一定参考。未来将对摩托车上驾乘人数在 2 人及以上的摩托车头盔佩戴情况进行研究。

参考文献

[1] 俞春俊，王长君. 摩托车头盔与摩托车交通事故的相关研究[J]. 中国安全生产科学技术，2009，5(2)：76-80.

[2] 邱敦国，王茂宁. 一种基于图像分析的摩托车不戴头盔违章事件检测方法：104200668A[P]. 2017-07-11.

[3] WARANUSAST R，BUNDON N，TIMTONG V，et al. Machine vision techniques for motorcycle safety helmet detection[J]. International Conference Image and Vision Computing New Zealand，2013：35-40.

[4] 王海，李诚，蔡英凤，等. 一种对摩托车驾驶员头盔佩戴情况的视觉检测方法：105760847A[P]. 2016-07-13.

[5] 孙伟，潘蓉，卞磊，等. 基于预分割和回归的深度学习目标检测[J]. 光学 精密工程，2017，25(10s)：221-227.

[6] GIRSHICKI R，DONAHUE J，DARRELL T，et al. Region-based convolutional networks for accurate object detection and segmentation [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence，2016，38(1)：142-158.

[7] HE K，ZHANG X，REN S，et al. Spatial pyramid pooling in deep convolutional networks for visualrecognition [J]. Lecture Notes in Computer Science(Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)，2014，8691 LNCS(PART 3)：346-361.

[8] GIRSHICK R. Fast R-CNN[J]. Proceedings of the IEEE International Conference on Computer Vision，2015，2015 Inter：1440-1448.

[9] REN S，HE K，GIRSHICK R，et al. Faster R-CNN：Towards real-time object detection with region proposal networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence，2017，39(6)：1137-1149.

[10] REDMON J，DIVVALA S，GIRSHICK R，et al. You only look once：Unified，real-time object detection[J]. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition，2016，2016-Decem：779-788.

[11] LIU W，ANGUELOV D，ERHAN D，et al. SSD：Single shot multibox detector[J]. Lecture Notes in Computer Science(Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)，2016，9905 LNCS：21-37.

[12] ZHOU D，HOU Q，CHEN Y，et al. Rethinking bottleneck structure for efficient mobile network design [J]. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)，2020，12348 LNCS(Xx)：680-697.

[13] HOWARD A G，ZHU M，CHEN B，et al. MobileNets：Efficient convolutional neural networks for mobile vision applications[J]. ArXiv Preprint，2017，ArXiv：1704.04861.

[14] SANDLER M，HOWARD A，ZHU M，et al. MobileNetV2：Inverted residuals and linear bottlenecks[J]. Proceedings of the IEEE Computer Society Conference on Computer Vision and Patern Recognition，2018：4510-4520.

[15] 徐晓光，李海. 多尺度特征在 YOLO 算法中的应用研究[J]. 电子测量与仪器学报，2021，35(6)：96-101.

[16] HUANG G，LIU Z，VAN D，et al. Densely connected convolutional networks [J]. Proceedings-30th IEEE Conference on Computer Vision and Pattern Recognition，CVPR 2017，2017，2017-Janua：2261-2269.

[17] 邓杰，万旺根. 基于改进 YOLOv3 的密集行人检测[J]. 电子测量技术，2021，44(11)：90-95.

[18] 朱江，杜瑞，李建奇，等. 基于注意力机制的曲轴瓦盖上料机器人视觉定位和检测方法[J]. 仪器仪表学报，2021，42(5)：140-150.

[19] WANG Q，WU B，ZHU P，et al. ECA-Net：Efficient channel attention for deep convolutional neural

networks[J]. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2020: 11531-11539.

[20] 崔江波,侯兴松. 基于注意力机制的 YOLOv4 输电线路故障检测算法[J]. 国外电子测量技术,2021,40(7): 24-29.

[21] LIN M, CHEN Q, YAN S. Network in network[J]. 2nd International Conference on Learning Representations, ICLR 2014-Conference Track Proceedings, 2014: 1-10.

[22] RAMACHANDRAN P, ZOPH B, LE Q V. Searching for activation functions[J]. 6th International Conference on Learning Representations, ICLR 2018-Workshop Track Proceedings, 2018: 1-13.

[23] HOWARD A, SANDLER M, CHEN B, et al. Searching for mobileNetV3[J]. Proceedings of the IEE International Conference on Computer Vision, 2019, 2019-Octob: 1314-1324.

[24] GHIASI G, LIN T Y, LE Q V. Dropblock: A regularization method for convolutional networks[J]. ArXiv Preprint,2018, ArXiv:1810.12890.

[25] 陈卓. Motorcycle helmet dataset[EB/OL]. (2021-5-28)[2021-8-14]. https://pan.baidu.com/s/1CIGQ17b6mhA8qsQdz7_Zw.

[26] VINCENT O R, MAKINDE A S,SALAKO O, et al. A self-adaptive K-means classifier for business incentive in a fashion design environment[J]. Applied Computing and Informatics, 2018, 14(1):88-97.

[27] RUDER S. An overview of gradient descent optimization[J]. Eprint ArXiv, 2016: 1-14.

[28] PADMINI V L, KISHORE G K, DURGAMALLESWARAO P, et al. Real time automatic detection of motorcyclists with and without a safety helmet [J]. Proceedings-International Conference on Smart Electronics and Communication, ICOSEC 2020, 2020(Icosec): 1251-1256.

[29] SHINE L, C. V J. Automated detection of helmet on motorcyclists from traffic surveillance videos: A comparative analysis using hand-crafted features and CNN[J]. Multimedia Tools and Applications, 2020, 79(19-20): 14179-14199.

[30] SIEBERT F W,LIN H. Detecting motorcycle helmet use with deep learning[J]. Accident Analysis and Prevention, 2020, DOI:10.1016/j.aap.2019.105319.

作者简介

冉险生,副教授,硕士生导师,主要研究方向为智能车辆技术、车辆动力学。

E-mail:cqrxs@qq.com

陈卓,硕士,主要研究方向为计算机视觉、深度学习、图像处理。

E-mail:729882383@qq.com

张禾,硕士,主要研究方向为图像处理、深度学习、计算机视觉。

E-mail: 1976286120@qq.com