

DOI: 10.11830/ISSN.1000-5013.202011014



面向无人机视频分析的 车辆目标检测方法

陶英杰^{1,2}, 张维纬^{1,2}, 马昕^{1,2}, 周密^{1,2}

(1. 华侨大学 工学院, 福建 泉州 362021;

2. 华侨大学 工业智能化与系统福建省高校工程研究中心, 福建 泉州 362021)

摘要: 提出一种将航拍车辆视频预处理与轻量化目标检测模型结合的级联方式。首先, 针对无人机拍摄的车辆视频数据大量冗余的问题, 在边缘设备设置一个两级过滤器, 通过帧的像素级及结构性差异过滤大量冗余帧, 从而大幅减少传输到后端的检测模型的帧数; 其次, 针对高精度目标检测模型时延高的问题, 采用通道剪枝与层剪枝结合的方法压缩 YOLOv3 模型并部署在 PC 端, 实现时延和精度的均衡。实验结果表明: 两级过滤器能够有效过滤 90% 以上的冗余帧数, 相较于原模型, 压缩模型在精度仅下降 2% 左右的情况下, 检测速度提高 78.3%, 达到 36.9 帧 \cdot s⁻¹。

关键词: 无人机; 视频预处理; 两级过滤器; YOLOv3; 模型压缩

中图分类号: TP 183; TP 391.41

文献标志码: A

文章编号: 1000-5013(2022)01-0111-08

Vehicle Target Detection Method for Unmanned Aerial Vehicle Video Analysis

TAO Yingjie^{1,2}, ZHANG Weiwei^{1,2}, MA Xin^{1,2}, ZHOU Mi^{1,2}

(1. College of Engineering, Huaqiao University, Quanzhou 362021, China;

2. Industrial Intelligence and System Fujian University Engineering Research Center,

Huaqiao University, Quanzhou 362021, China)

Abstract: A cascading method combining aerial vehicle video preprocessing with lightweight target detection model was proposed. Firstly, aiming at the problem of massive redundancy of vehicle video data shot by unmanned aerial vehicle, a two-stage filter was set in the edge device to filter a large number of redundant frames through the pixel level and structural difference of the frames, so as to greatly reduce the number of frames transmitted to the detection model at the back end. Secondly, to solve the problem of high delay of high-precision target detection model, a combination of channel pruning and layer pruning was used to compress the YOLOv3 model and deploy it on PC to achieve the balance of delay and precision. The experimental results show that the two-stage filter can effectively filter more than 90% of the redundant frames. Compared with the original model, the detection speed of the compression model is increased by 78.3%, reaching 36.9 frames per second, when the accuracy of the compression model is only decreased by about 2%.

收稿日期: 2020-11-04

通信作者: 张维纬(1981-), 男, 副教授, 博士, 主要从事大数据、物联网及边缘智能的研究. E-mail: 178483968@qq.com.

基金项目: 国家自然科学基金面上资助项目(61976098); 福建省泉州市科技计划项目(2020C067); 华侨大学研究生科研创新基金资助项目(18014084010)

Keywords: unmanned aerial vehicle; video preprocessing; two-stage filter; YOLOv3; model compression

随着交通基础设施的发展,中国许多城市在市区安装了成千上万个交通监控设备.这些视频要传输到监控中心,并进行人工分析,成本昂贵且工作实效低.目前,无人机辅助作业因成本低、体积小、灵活方便等优点,成为一项极具发展前景的技术.然而,无人机的主要作用是拍摄车辆特征明显的视频,所拍摄的视频还是需要进行人工分析.因此,无人机要真正做到“无人”,主要面临以下困境:航拍的视频帧数巨大,若直接处理视频中的每一帧,计算量大;完整的目标检测模型部署在嵌入式设备上,会导致能耗高、计算量高、时延高等问题.目前,基于运动的车辆检测主要分为非参数法和参数法两类^[1].非参数法主要以逐像素的方式分离前景和背景,如帧差法^[2]在固定监控、低帧率的运动车辆检测中运用广泛,但是当该方法用于高帧率的航拍视频去除大量冗余帧时,会丢失大量目标帧.参数法主要分为背景建模、光流及其他方法.背景建模^[3]主要针对视频流建立背景模型,应用前提是基于背景固定的假设.Zhan 等^[4]利用金字塔光流估计和形态学变换算子提取运动车辆目标,然而,光流法所需要的计算资源是嵌入式设备无法满足的.其他方法包括自适应运动直方图的方法^[5]、基于运动模型的方法^[6]和利用深度学习挖掘关键帧的方法^[7]等,但是此类方法存在时耗长、计算量大及模型大的问题,难以部署在嵌入式设备上.

为了去除视频冗余帧,减少不必要的计算,研究者基于运动的车辆检测方法,结合其他方法进行了多种尝试.Zhang 等^[8]采用周期性固定间隔选取帧,再利用结构性相似的方案选取关键帧,但是固定间隔会导致目标帧丢失过多,检出率不高;Zhang 等^[9]以背景图像为参考帧,通过帧差图像判断并提取关键帧,但这种方法需要背景图像参考,对移动切换的场景不适用;Kang 等^[10]直接通过跳帧以减少视频数据量,这种方法简单,但会出现无目标帧剩余情况.另外,这些方法将预处理方法和检测模型均部署本地,边缘计算资源浪费.因此,本文提出一种面向无人机视频分析的车辆目标检测方法.通过在无人机设备上将实时拍摄的视频进行像素级及结构性差异过滤处理,得到关键帧,再通过部署在 PC 上压缩的检测模型进行检测.

1 无人机视频过滤机制

1.1 总体框架

文中提出的框架包括两级过滤器和目标检测器,其中,两级过滤器部署在嵌入式设备上,目标检测器部署在 PC 端,如图 1 所示.无人机拍摄的 M 帧视频通过灰度转换,输入像素差异检测器(PDD)判断相邻帧的相似程度,根据相似度选择丢弃或保留;一级筛选后将剩余的 $M-N$ 帧输入结构差异检测器(SDD)中,根据更符合人类视觉的结构性判断帧之间的相似度,剔除 P 帧;最后,将两级过滤后的 $M-N-P$ 帧送入压缩并添加计数模块的 YOLOv3 模型进行检测.

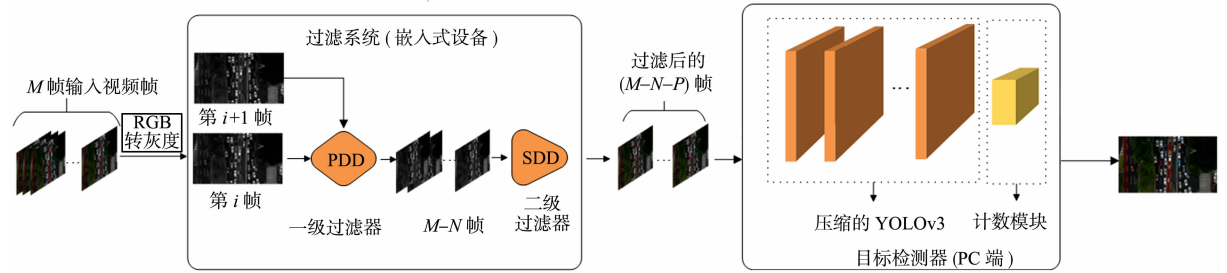


图 1 文中算法的结构框架

Fig. 1 Structural framework of this paper algorithm

1.2 两级过滤器的设计

1.2.1 像素差异检测器 像素差异检测器通过计算相邻帧之间的像素距离确定这两个帧是否相同.Himeur 等^[11]通过直方图的方法捕捉颜色信息的相似性,但是无人机拍摄的视频分辨率较高,一张像素为 $200\text{ px}\times 200\text{ px}$ 的图片就有 40 000 个像素点,每一个像素点都保存着一个 RGB 值,信息量相当庞大,而无人机拍摄的视频图片像素远不止 $200\text{ px}\times 200\text{ px}$. Gao 等^[12]的实验结果显示,如果将 RGB 图像转化为灰度图,会降低计算成本,且不影响效果.因此,首先将 M 个视频帧转换为灰度图;其次把图片

缩放到非常小, 根据对比显示缩放到 $9 \text{ px} \times 8 \text{ px}$ 是相对合理的, 因为每行 9 个像素值, 会产生 8 个差异值, 刚好构成一个字节, 可以转化为 2 个 16 进制值进行后续计算。

将 M 帧缩放过的灰度图输入差异检测器, 判断相邻帧的相似程度, 如果相似, 则保留相似帧中的一帧。第 i 帧图像相邻像素差值的判断方法为

$$a_i(x, y) = f_i(x, y) - f_i(x, y+1). \quad (1)$$

式(1)中: x, y 为图像像素的横、纵坐标; $f_i(x, y)$ 为第 i 帧图像的 x 行 y 列的像素值; $a_i(x, y)$ 表示 2 个像素差值。

第 i 帧图像的相邻像素强度的判断方法为

$$d_i(x, y) = \begin{cases} 1, & a_i(x, y) \geq 0, \\ 0, & a_i(x, y) < 0. \end{cases} \quad (2)$$

式(2)中: 产生全为 0 和 1 的 8×8 矩阵, 而每 8 位可以组成 2 个 16 进制值, 连接起来转换为字符串, 得到哈希值。

第 i 帧和第 j 帧图像的差异值的判断方法为

$$D_{i,j}(k) = H_i(k) \oplus H_j(k). \quad (3)$$

式(3)中: $H_i(k)$ 和 $H_j(k)$ 分别为第 i 帧和第 j 帧图像的第 k 个哈希值转换的二进制值; \oplus 表示异或运算; $D_{i,j}(k)$ 为第 i, j 帧哈希值的运算结果。

两帧图像的相似判断图为

$$\Delta_{i,j} = \sum_k D_{i,j}(k). \quad (4)$$

式(4)中: $\Delta_{i,j}$ 为两帧之间的相似度量参数, 当 $\Delta_{i,j} = 5$ 时, 能有效度量两帧航拍图像之间的相似度。

1.2.2 结构差异检测器 通过 PDD 的过滤, 可以初步过滤掉 70% 以上的冗余帧, 但是交通拥堵事件发生的平均时间不超过 5%^[13], 而且无人机的应用场景和固定摄像头不同, 场景随时切换, 图像结构会发生变化, 还应进一步过滤。而结构相似性倾向于通过统计指标(如熵、灰度平均值、协方差)和图像质量评价指标(如峰值信噪比(PSNR)和结构相似性指数(SSIM))对图像进行全局比较。Hore 等^[14]的实验结果表明, SSIM 是一个完整的参考图像质量评价指标, 它结合了亮度、对比度和结构衡量图像的相似性, 在图像相似性评价上优于 PSNR。因此, 结构差异检测器采用 SSIM 算法度量图像的结构相似性。

给定 2 幅图片, SSIM 计算式为

$$M_{\text{SSI}}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{x,y} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}. \quad (5)$$

式(5)中: μ_x, μ_y 表示图像的均值; σ_x^2, σ_y^2 表示图像的方差; $\sigma_{x,y}$ 表示图像的协方差; C_1, C_2 为常数, 为了避免分母为零而维持稳定, 通常取 $C_1 = (k_1 L)^2, C_2 = (k_2 L)^2, L$ 为像素值的动态范围, 一般取 255, 根据文献^[15]给出的 k_1, k_2 默认值, 取 $k_1 = 0.01, k_2 = 0.03$ 。SSIM 的相似度判定阈值 β 取 0.8 时, 能够有效度量两帧图像之间的相似度。

根据实验发现, 利用 SSIM 算法将无人机拍摄的视频进行相邻帧的相似度量并做保留或丢弃处理, 会产生相似度传递, 并累积错误, 如图 2 所示。

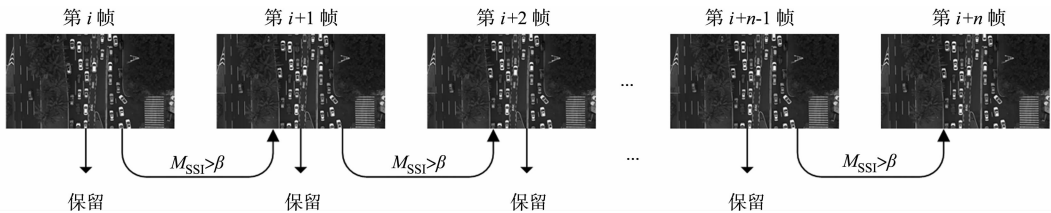


图 2 SSIM 算法产生相似度传递

Fig. 2 SSIM algorithm produces similarity transfer

由图 2 可知: 对于 M 帧的视频, 连续的帧与帧之间的相似度非常高, 最坏的可能是最后保留 $M-1$ 帧图像, 只进行相似度量, 没有过滤帧。因此, 对应用航拍视频的 SSIM 算法进行改进。

伪代码描述如下。

算法:基于航拍视频的帧结构差异过滤算法
输入:Aerial Video,SSIM,相似度判定阈值 β
输出:相似度小于 β 的帧

```
1: i ← 0
2: j ← 0
3: read Aerial Video
4: save first frame
5: frame numbers save to array A
6: B ← A
7: while length[A]>i and length[B]>j
8: do 取出 A[i],B[j]
9: if SSIM(A[i],B[j])>β
10: then j ← j+1
11: End if
12: if SSIM(A[i],B[j])<β
13: then save B[j]
14: i ← j
15: end
```

1.3 YOLOv3 模型的压缩

为实现在线实时高精度检测,选取精度较高的 YOLOv3 作为基础网络,其速度和精度在相同情况下均优于 SSD^[16],fast R-CNN^[17]等主流算法,在工业界也有广泛的应用^[18-21]. YOLOv3 模型能避免车辆大量漏检情况的发生,具有较高的精度和可优化的空间.考虑到目前的压缩方法^[22-23]对精度和速度单方面的需求,没有将速度与精度进行权衡.故将通道剪枝和层剪枝方法结合,保证精度需求的情况下对 YOLOv3 模型进行加速.

受到 SlimYOLOv3^[24]的启发,为了便于通道修剪,为每个通道分配一个比例因子,其中,比例因子的绝对值表示通道的重要性.在 YOLOv3 中的每个卷积层之后都有一个 BN 层,用以加速收敛和提高泛化能力,BN 层使用小批量归一化卷积特征,即

$$f=\gamma\times\frac{x-\mu}{\sigma}+e.$$
(6)

式(6)中: μ 和 σ 分别为输入特征的均值和标准差; γ 和 e 分别为比例因子和偏差.

直接采用比例因子作为通道重要性的指标,为了有效区分重要通道和不重要通道,对 γ 增加一个正则项^[25],即

$$L=l_{\text{yolo_spp}}+\lambda\sum_{\gamma\in\Gamma}\varphi(\gamma).$$
(7)

式(7)中: L 为损失函数; $l_{\text{yolo_spp}}$ 为模型预测产生的损失; $\lambda\sum_{\gamma\in\Gamma}\varphi(\gamma)$ 用来约束 γ ,其中, λ 是权衡两项的超参数, $\varphi(\gamma)=|\lambda|$,就是 L1 范式,可达到稀疏的作用.

通道剪枝示意图,如图 3 所示.

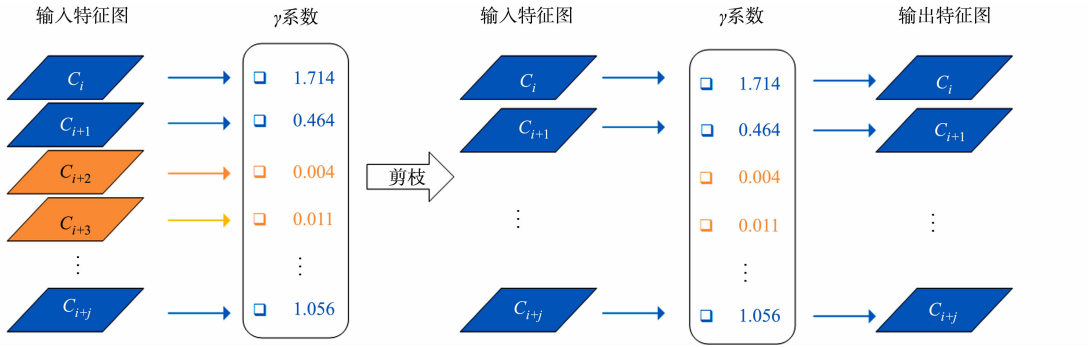


图 3 通道剪枝示意图

Fig. 3 Schematic diagram of channel pruning

在通道剪枝的基础上,对每一个 shortcut 层前面一个 CBL(YOLOv3 网络结构中的最小组件,由 conv+BN+Leaky_relu 激活函数三者组成)进行评价,对各层的 γ 均值进行排序,最小的 γ 均值进行层剪枝.为了保证结构完整性,每剪掉一个 shortcut 结构,会同时剪掉一个 shortcut 层和前面 2 个 conv 层.对于 YOLOv3,有 23 处 shortcut,共有 69 个层剪层空间.

层剪枝示意图,如图 4 所示.图 4 中: $\gamma_i < \gamma_{i+1}$. 根据选取交并比(IoU) ≥ 0.5 的预测框,进行计数,统计预测车辆的数量.

2 实验结果与分析

2.1 实验环境配置

系统为 ubuntu18.04,搭建 Pytorch 框架进行训练,通过 Python 进行测试.训练硬件配置为 Intel(R) Xeon (R) Gold 5118 CPU@2.3 GHz,NVIDIA GeForce TI-TAN Xp;测试硬件配置为 PC 机 Intel(R) Core(TM) i5-8300H CPU@2.30 GHz,NVIDIA GeForce GTX 1050;jetson nano 为 64 位 4 核 ARM A57@1.43 GHz,128 核 NVIDIA Maxwell@921 MHz.实验方案如下:首先,使用部署在嵌入式设备 jetson nano 上的两级过滤器对采集的无人机视频进行预处理并进行效果对比;再将处理后的剩余帧传输到部署在 PC 端的轻量化目标检测器进行检测;最后,展示效果.

2.2 性能评价标准

实验训练数据为 UA-DETRAC,VisDrone2018 数据集,采集一段无人机视角的交通视频,分辨率为 1 280 px \times 720 px,帧数为 3 634 帧.其中,目标帧数为 185 帧,由于目标帧之间也存在冗余,故每 5 帧取 1 帧作为实际目标帧,即认为实际目标帧为 37 帧.

为了能够评估无人机视频过滤系统性能,对时延、帧总过滤率、目标帧保留率、目标帧占比、冗余帧过滤率等参数进行评估.

帧总过滤率(R_{TFF})表达式为

$$R_{\text{TFF}} = \frac{F_{\text{fil}}}{F_{\text{tot}}}. \quad (8)$$

目标帧保留率(R_{TFR})表达式为

$$R_{\text{TFR}} = \frac{F_{\text{tot}_t} - F_{\text{fil}_t}}{F_{\text{tot}_t}}. \quad (9)$$

目标帧占比(P_{TF})表示被过滤的剩余帧数中含有多少目标帧,其表达式为

$$P_{\text{TF}} = \frac{F_{\text{tot}_t} - F_{\text{fil}_t}}{F_{\text{tot}} - F_{\text{fil}}}. \quad (10)$$

冗余帧过滤率(R_{RFF})表达式为

$$R_{\text{RFF}} = \frac{F_{\text{fil}} - F_{\text{fil}_t}}{F_{\text{tot}} - F_{\text{tot}_t}}. \quad (11)$$

式(8)~(11)中: F_{tot} 为测试视频的总帧数; F_{tot_t} 为测试视频包含的目标帧帧数; F_{fil} 为测试视频被过滤的帧数; F_{fil_t} 为测试视频被过滤的目标帧帧数.单一的目标帧保留率或冗余帧过滤率无法验证过滤器的性能,因此,将二者结合起来评估过滤器性能,帧总过滤率作为参考.

为了能够评估加速的目标检测器的性能,采用平均精确率均值(P_{mA})、检测速度及参数总量作为评价准则,其中,平均精确率均值是目标检测任务中最重要的指标,决定了检测效果.为了在计算机中更快速精确地求出 P_{mA} 的大小,通常使用以下方法,即

$$P_{\text{mA}} = \frac{1}{n} \sum_{i=1}^n \int_0^1 P(r) dr. \quad (12)$$

2.3 结果分析

为了验证过滤系统的效果并评估其性能,在 jetson nano 平台将文中方法与文献[8]、文献[9]、文献[10]中的方法进行对比实验.不同方法下的过滤时间和过滤效果的对比,如图 5,6 所示.图 5 中: t 为过滤时间.

文中方法通过 PDD 将高分辨率帧进行像素级压缩,再进行哈希值相似度判断,大大减少了计算量,时耗仅为 11.7 s.过滤系统不仅要提高速度,还要提高目标帧的保留率,因此,使用 SDD 根据人类视觉,

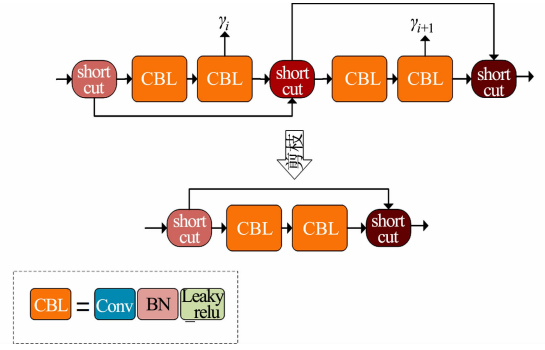


图 4 层剪枝示意图

Fig. 4 Schematic diagram of layer pruning

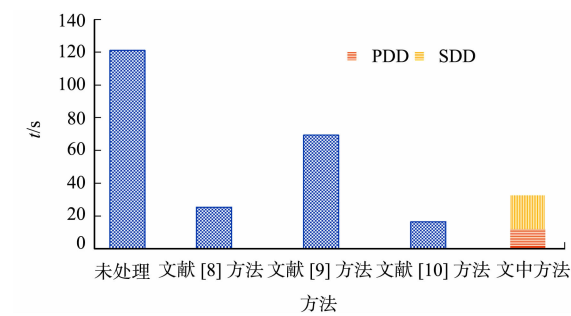


图 5 不同方法下的过滤时间对比
Fig. 5 Comparison of filtering time
in different methods

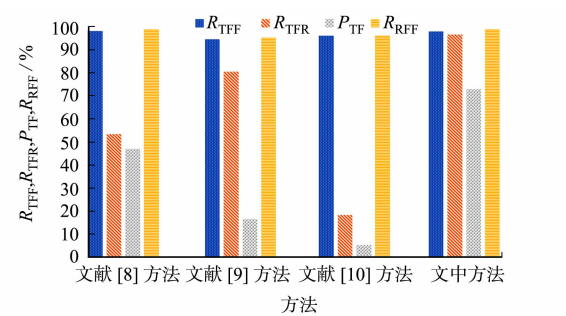


图 6 不同方法下的过滤效果对比
Fig. 6 Comparison of filtering effects
in different methods

通过结构化相似度判断,进一步去除冗余帧.由于已经通过 PDD 初步筛选,因此,SDD 需要判断的帧数相对减少,耗时仅为 20.77 s.由图 5 可知:文献[8]和文献[10]方法的过滤时间分别为 25.30,16.34 s,这是由于文献[10]直接进行了跳帧处理,文献[8]在选择跳帧后进行了相似度检测,因此,两种方法的速度都较快.

由图 6 可以看出:文中方法的目标帧保留率 R_{TFR} 明显优于文献[8]、文献[9]、文献[10]的方法,达到 97.30%,几乎可以保留所有目标帧;文中方法、文献[8]、文献[9]和文献[10]方法的冗余帧过滤率 R_{RFF} 都非常高,分别是 99.38%,95.96%,96.83%和 99.64%.但单一指标无法评估性能,由综合目标帧占比率 P_{TF} 可以看出,在剩余的帧数中,使用两级过滤器的文中方法含有目标帧的帧数分别是文献[8]、文献[9]、文献[10]方法的 1.54,4.29,5.14 倍.

YOLOv3 通道剪枝压缩情况,如表 1 所示.采用层剪枝的方式将 YOLOv3 中的 shortcut 层进行修剪,为了保证结构的完整性,同时将 shortcut 层对应的 2 个 CBL 层也剪掉. YOLOv3 层剪枝压缩情况,如表 2 所示.

表 1 YOLOv3 通道剪枝压缩情况
Tab.1 YOLOv3 channel pruning compression situation

参数	层索引													
	0	1	2	3	5	...	60	62	63	64	...	109	110	111
所有通道数	32	64	32	64	128	...	512	1 024	512	1 024	...	256	128	256
剩余通道数	21	35	16	27	65	...	79	278	63	206	...	68	32	57

表 2 YOLOv3 层剪枝压缩情况
Tab.2 YOLOv3 layer pruning compression situation

剪掉的 shortcut 层	15	18	27	43	46	49	55	61	65	68	71	74
剪掉的 CBL 层	13	16	25	41	44	47	53	59	63	66	69	72
	14	17	26	42	45	48	54	60	64	67	70	73

压缩后的 YOLOv3 模型的各项指标,如表 3 所示.为了评估压缩 YOLOv3 的效果与性能,在 PC 端将压缩模型与 YOLOv3 模型和轻量化 YOLOv3 模型的各项指标进行对比,结果如表 4 所示.表 3,4 中: V 为模型参数量; t_f 为前向推断耗时; F 为检测速度.

通过通道剪枝,模型大小为 33.2 MB,参数压缩了 86.8%,再通过层剪枝,剪了 12 个 shoutcut,相当于剪了 36 层,模型大小为 22 MB.

表 3 压缩后的 YOLOv3 模型的各项指标
Tab.3 Various indicators of
compressed YOLOv3 model

指标	压缩方式		
	稀疏训练	剪通道	剪通道+剪层
P_{mA}	0.89	0.59	0.63
$V/\times 10^6$	62.57	8.28	5.49
t_f/ms	11.8	8.8	8.4

表 4 压缩模型与 YOLO 模型对比
Tab.4 Comparison between compression
model and YOLOv3 model

指标	YOLOv3 模型	轻量化 YOLOv3 模型	压缩模型
P_{mA}	0.86	0.31	0.63
$F/\text{帧}\cdot\text{s}^{-1}$	20.7	40.3	36.9
$V/\times 10^6$	62.10	8.80	5.49
t_f/ms	29.2	7.1	8.4

通过层剪枝和通道剪枝的结合, 去除对网络输入结果影响小的卷积核和卷积层, 压缩了模型的深度和宽度, 从而加快了模型检测速度; YOLOv3 模型的参数量压缩了 91.23%, 大大减少了计算量. 由表 4 可知: 与 YOLOv3 模型相比, 压缩模型虽然在精度方面降低了 20% 左右, 但是检测速度提高了 78.3%, 推断速度提升了 71.2%; 相较于轻量化 YOLOv3 模型, 压缩模型的精度提高了 103%, 而检测速度差别不大. 3 种模型各截取一帧, 检测效果如图 7 所示.



(a) YOLOv3 模型

(b) 轻量化 YOLOv3 模型

(c) 压缩模型

图 7 3 种模型检测效果

Fig. 7 Detection effects in three models

3 结束语

针对视频帧大量冗余的问题, 进行了大规模且有效的过滤; 另外, 对全功能模型检测速度慢的问题, 进行了压缩加速, 在保证精度的情况下, 减少模型的参数总量和模型体积, 与现有的模型相比, 实现了精度与速度的均衡. 然而, 研究的最终目的是将过滤器及检测模型全部部署在无人机上, 充分利用边缘设备计算资源. 因此, 在今后工作中, 将会进一步研究模型的压缩和改进, 从而实现无人机实时在线全功能过滤交通视频检测异常的情况.

参考文献:

- [1] YANG Zi, PUN-CHENG L S C. Vehicle detection in intelligent transportation systems and its applications under varying environments: A review[J]. *Image and Vision Computing*, 2018, 69: 143-154. DOI: 10.1016/j.imavis.2017.09.008.
- [2] XIA Yingjie, WANG Chunhui, SHI Xingmin, *et al.* Vehicles overtaking detection using RGB-D data[J]. *Signal Processing*, 2015, 112: 98-109. DOI: 10.1016/j.sigpro.2014.07.025.
- [3] ELGAMMAL A, HARWOOD D, DAVIS L. Non-parametric model for background subtraction[C]// *European Conference on Computer Vision*. Dublin: Springer, 2000: 751-767.
- [4] ZHAN Wei, JI Xiaolong. Algorithm research on moving vehicles detection[J]. *Procedia Engineering*, 2011, 15: 5483-5487. DOI: 10.1016/j.proeng.2011.08.1017.
- [5] ZHANG Wei, WU Q M J, YIN Haibing. Moving vehicles detection based on adaptive motion histogram[J]. *Digital Signal Processing*, 2010, 20(3): 793-805. DOI: 10.1016/j.dsp.2009.10.006.
- [6] JAZAYERI A, CAI Hongyuan, ZHENG Jiangyu, *et al.* Vehicle detection and tracking in car video based on motion model[J]. *IEEE Transactions on Intelligent Transportation Systems*, 2011, 12(2): 583-595. DOI: 10.1109/TITS.2011.2113340.
- [7] ZHU Wangjiang, HU Jie, SUN Gang, *et al.* A key volume mining deep framework for action recognition[C]// *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Las Vegas: IEEE Press, 2016: 1991-1999. DOI: 10.1109/CVPR.2016.219.
- [8] ZHANG Jing, LIANG Xi, WANG Meng, *et al.* Coarse-to-fine object detection in unmanned aerial vehicle imagery using lightweight convolutional neural network and deep motion saliency[J]. *Neurocomputing*, 2020, 398: 555-565. DOI: 10.1016/j.neucom.2019.03.102.
- [9] ZHANG Chen, CAO Qiang, JIANG Hong, *et al.* FFS-VA: A fast filtering system for large-scale video analytics [C]// *Proceedings of the 47th International Conference on Parallel Processing*. Eugene: ACM, 2018: 1-10. DOI: 10.1145/3225058.3225103.
- [10] KANG D, EMMONS J, ABUZAID F, *et al.* Noscope: Optimizing neural network queries over video at scale[J].

- Proceedings of the VLDB Endowment, 2017, 10(11): 1586-1597. DOI: 10.14778/3137628.3137664.
- [11] HIMEUR Y, AIT-SAADI K, OUAMANE A. A fast and robust key-frames based video copy detection using BSIF-RMI[C]// International Conference on Signal Processing and Multimedia Applications. Vienna: IEEE Press, 2016: 16156288. DOI: 10.5220/0005060000400047.
- [12] GAO Zhen, LU Guoliang, LYU Chen, *et al.* Key-frame selection for automatic summarization of surveillance videos: A method of multiple change-point detection[J]. Machine Vision and Applications, 2018, 29(7): 1101-1117. DOI: 10.1007/s00138-018-0954-7.
- [13] WANG Haizhong, RUDY K, LI Jia, *et al.* Calculation of traffic flow breakdown probability to optimize link throughput[J]. Applied Mathematical Modelling, 2010, 34: 3376-3389. DOI: 10.1016/j.apm.2010.02.027.
- [14] HORE A, ZIOU D. Image quality metrics: PSNR vs. SSIM[C]// 20th International Conference on Pattern Recognition. Istanbul: IEEE Press, 2010: 2366-2369. DOI: 10.1109/ICPR.2010.579.
- [15] WANG Zhou, BOVIK A C, SHEIKH H R, *et al.* Image quality assessment: From error visibility to structural similarity[J]. IEEE Transactions on Image Processing, 2004, 13(4): 600-612. DOI: 10.1109/TIP.2003.819861.
- [16] LIU Wei, ANGUELOV D, ERHAN D, *et al.* SSD: Single shot multibox detector[C]// European Conference on Computer Vision. Amsterdam: Springer, 2016: 21-37. DOI: 10.1007/978-3-319-46448-0_2.
- [17] GIRSHICK R. Fast R-CNN[C]// Proceedings of the IEEE International Conference on Computer Vision. Santiago: IEEE Press, 2015: 1440-1448. DOI: 10.1109/ICCV.2015.169.
- [18] 姚巍巍, 张洁. 基于模型剪枝和半精度加速改进 YOLOv3-tiny 算法的实时司机违章行为检测[J]. 计算机系统应用, 2020, 29(4): 41-47. DOI: 10.15888/j.cnki.csa.007348.
- [19] 张富凯, 杨峰, 李策. 基于改进 YOLOv3 的快速车辆检测方法[J]. 计算机工程与应用, 2019, 55(2): 12-20. DOI: 10.3778/j.issn.1002-8331.1810-0333.
- [20] 陈宏彩, 任亚恒, 郝存明, 等. 基于 YOLOv3 的医药玻璃瓶缺陷检测方法[J]. 包装工程, 2020, 41(7): 241-246. DOI: 10.19554/j.cnki.1001-3563.2020.07.034.
- [21] 隋靓, 党建武. 基于运动目标轨迹的高速公路异常事件检测算法研究[J]. 计算机应用与软件, 2018, 35(1): 246-252. DOI: 10.3969/j.issn.1000-386x.2018.01.043.
- [22] HAN Song, MAO Huizi, DALLY W J. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding[C]// 4th International Conference on Learning Representations. San Juan: [s. n.], 2016: 1-13.
- [23] YAO Shuochao, ZHAO Yiran, ZHANG A, *et al.* Deepiot: Compressing deep neural network structures for sensing systems with a compressor-critic framework[C]// Proceedings of the 15th ACM Conference on Embedded Network Sensor Systems. Delft: ACM, 2017: 1-14. DOI: 10.1145/3131672.3131675.
- [24] ZHANG Pengyi, ZHONG Yunxin, LI Xiaoqiong. SlimYOLOv3: Narrower, faster and better for real-time UAV applications[C]// Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops. Seoul: IEEE Press, 2019: 37-45. DOI: 10.1109/ICCVW.2019.00011.
- [25] LIU Zhuang, LI Jianguo, SHEN Zhiqiang, *et al.* Learning efficient convolutional networks through network slimming[C]// Proceedings of the IEEE International Conference on Computer Vision. Venice: IEEE Press, 2017: 2736-2744. DOI: 10.1109/ICCV.2017.298.

(责任编辑: 黄晓楠 英文审校: 吴逢铁)