

基于多粒度剪枝的水下遗迹实时目标检测

张有波^{1,2}, 郭威^{2,3*}, 周悦¹, 徐高飞², 李广伟², 孙洪鸣^{2,3}

¹上海海洋大学工程学院, 上海 201306;

²中国科学院深海科学与工程研究所, 海南 三亚 572000;

³中国科学院大学, 北京 100049

摘要 针对水下机器人搭载平台进行深海遗迹探寻所面临的数据样本缺乏、采集图像模糊、嵌入式系统计算能力有限的问题, 分别在数据增强、视频增强和目标检测算法压缩等方面提出了有效的解决方案。首先, 利用数据增强提高小样本数据的泛化能力, 通过迁移学习进行知识迁移, 加速模型收敛; 然后, 基于结构相似性对采集视频进行关键帧图像选择, 采用限制对比度直方图拉伸对选择后的关键帧进行实时图像增强; 最后, 基于多粒度剪枝策略对 YOLOV4 进行通道和卷积层的双向压缩。实验结果表明, 压缩后的 YOLOV4 模型运算复杂度(BFLOPS)降为 10.588, 在 Jetson TX2 嵌入式图像处理器上, 对大小为 640 pixel×480 pixel 的输入图像的平均检测速度可以达到 18.2 frame/s。

关键词 图像处理; 视频图像增强; 模型压缩; 嵌入式图像处理器; 目标检测; 水下机器人

中图分类号 TP391.4 **文献标志码** A

doi: 10.3788/LOP202158.1410019

Real-Time Target Detection of Underwater Relics Based on Multigranularity Pruning

Zhang Youbo^{1,2}, Guo Wei^{2,3*}, Zhou Yue¹, Xu Gaoifei², Li Guangwei², Sun Hongming^{2,3}

¹College of Engineering Science and Technology, Shanghai Ocean University, Shanghai 201306, China;

²Institute of Deep-Sea Science and Engineering, Chinese Academy of Sciences, Sanya, Hainan 572000, China;

³University of Chinese Academy of Sciences, Beijing 100049, China

Abstract An effective solution for data enhancement, video enhancement, and target detection algorithm compression is proposed to address the data deficiency, blurred images, and limited computing power of an embedded system of the unmanned underwater vehicle for searching underwater relics. First, the data augmentation was used to improve the generalization ability of small sample, and knowledge transfer was then implemented through transfer learning to accelerate the convergence of models. Then, key frame images were selected from the captured video based on the structural similarity, and the selected key frames were enhanced using the contrast limited adaptive histogram equalization stretching in real time. Finally, based on the multigranularity pruning strategy, YOLOV4 performed bidirectional compression of the channel and convolutional layer. The experimental results show that the operation complexity (BFLOPS) of the compressed YOLOV4 model is reduced to 10.588. The average detection speed for images with a size of 640 pixel×480 pixel on the Jetson TX2 embedded image processor reaches 18.2 frame/s.

Key words image processing; video enhancement; model compression; embedded image processor; target detection; unmanned underwater vehicle

OCIS codes 100.2000; 100.4996; 100.2980; 100.3008

收稿日期: 2020-10-20; **修回日期:** 2020-11-05; **录用日期:** 2020-12-02

基金项目: 国家重点研发计划(2020YFC1521704)、海南省自然科学基金(2019RC260)、三亚市院地科技合作项目(2019YD01)

通信作者: *guow@idsse.ac.cn

1 引言

中国拥有辉煌的海上贸易历史,受当时航海技术限制和极端天气影响,海难事故时有发生,仅在中国南海海域就散落着大量的古代沉船遗迹。《国家文物事业发展“十三五”规划》强调重点研发水下考古机器人搭载平台、文物数字化装备和智能感知装备等终端设备。通过水下机器人(UUV)进行水下沉船遗迹探寻和打捞不仅可以重现历史,还原当时的社会形态,同时也可以为更好地研究古代海上丝绸之路之路填补资料空白。

沉船于海水的长期浸泡和海洋生物附着下的外貌会发生较大变化,仅通过探寻船体遗骸很难发现水下遗迹。陶瓷、丝绸是中国古代出口的主要商品,其中因独特的材料和加工特性,陶瓷在长时间的海水浸泡和泥土埋藏中可以保持与原始模样高度一致的形态,这给 UUV 通过探寻水下陶瓷进行海底遗迹定位提供了可能。与地面目标检测相比,水下物体检测存在更多挑战,具体有三个方面的限制:深海数据采集难度高,目标训练样本有限,通过小样本数据集直接进行模型训练时难以获得收敛的检测模型;水下环境复杂,海水对光的选择性吸收会造成水下物体颜色发生不同程度的变化,使得采集图像模糊,增加模型预测难度;UUV 受体积和功耗限制,只能搭载嵌入式设备,与实验室大功率计算设备相比,嵌入式设备的运算能力有较大限制,而快速识别是 UUV 实现实时目标检测的核心^[1]。

基于深度学习的目标检测算法目前主要分为两类:一类是通过生成目标推荐区域再进行预测的 two-stage 目标检测算法,代表算法以 R-CNN 系列为主,包括 Fast-RCNN、Faster-RCNN、R-FCN、Mask RCNN 等,检测准确度高,但检测速度慢^[2-4];另一类是通过神经网络直接进行预测的 one-stage 目标检测算法,典型代表有 You Only Look Once (YOLO) 系列、Single Shot MultiBox Detector (SSD)、RetinaNet 等。one-stage 目标检测算法直接通过卷积神经网络(CNN)进行物体检测,检测速度快,但想要提高检测精度往往需要特征提取能力更强的主干网络^[5-9]。为进一步提高算法检测精度,ResNet 采用“跳连”方式加深特征提取网络^[10],DenseNet 通过密集连接方式加宽特征提取网络^[11],Dual Path Network (DPN) 吸收 ResNet 和 DenseNet 的优点,在加深网络的同时加宽网络^[12]。

但是随着网络深度和宽度的增加,模型参数量也在逐渐增加,UUV 携带的硬件设备的计算能力往往达不到网络推理要求。为满足 UUV 水下目标检测需求,需要对目标检测网络进行压缩,裁剪掉网络中冗余和不重要的神经元及其网络连接,将压缩后的模型移植到 UUV 上的嵌入式平台,完成在线实时目标检测。

本文围绕 UUV 在线进行水下文物识别所需解决的关键问题开展研究。针对嵌入式设备计算能力和功耗有限的问题,基于多粒度剪枝策略对 YOLOV4 进行模型双向压缩,在保证算法精度的同时最大程度上提高模型检测速度;针对水下目标数据集缺少问题,通过在线和离线数据增强提高目标数据泛化能力;针对采集视频图像模糊的问题,基于结构相似性(SSIM)对采集视频进行关键帧选择,通过限制对比度直方图拉伸对关键帧图像进行实时在线增强,降低系统资源消耗的同时提高了模型预测准确度。压缩后的 YOLOV4 模型体积降为 14.3 MB,在 Nvidia Jetson TX2 嵌入式设备上可以达到 18.2 frame/s 的平均处理速度,为 UUV 在线实时目标检测提供了一种可行的方法。

2 模型压缩

2.1 神经网络压缩方法

深度学习目标检测模型是以神经元为基础由特定的网络连接结构组成的,利用误差反向传播更新每个神经元的权重。随着训练迭代次数增加,模型中的某些神经元对结果的调节作用越来越小,对目标识别基本没有贡献,但是在网络结构上的存在又会增加模型体积。模型压缩的本质是指在保证精度的前提下,对模型中不重要的神经元及其网络连接进行剔除,降低模型体积实现模型瘦身,从而达到快速检测的目的。目前用于目标检测神经网络的主要压缩方法有模型剪枝、模型量化、知识蒸馏和网络架构搜索^[13],如表 1 所示。

模型裁剪具体可分为权重更新限制和不进行权重更新限制两种。不进行权重更新限制方式通过参数评价函数,直接对不重要的参数所在的神经元或网络连接进行裁剪,压缩后的模型的精度受参数评价函数影响较大。权重更新限制主要通过通过对网络权重施加惩罚项,对参数更新进行限制,利用稀疏矩阵实现模型压缩。根据剪枝粒度级别的不同,模型剪枝可分为层间剪枝、特征剪枝、核剪枝以及核内剪

表 1 神经网络压缩方法概要
Table 1 Summary of neural network compression methods

Method	Principle	Advantage	Disadvantage
Pruning	Prune unimportant parameters and connections	Flexible operation and small loss of precision	Need fine-tuning, training time increase
Quantization	Reduce parameter bit width	Low computation cost	Accuracy drops too severely to restore
Knowledge distillation	Transfer knowledge from complex models to small models	A small number of parameters are used for calculation	Knowledge transfer standards are difficult to determine and only apply to classification problems
Network architecture search	Automatically search for a network model that meets the parameters and accuracy requirements	High accuracy	High training costs, search indicators are difficult to set

枝,分别从卷积层、特征图、卷积核通道以及卷积核内权重上进行对应的冗余结构剔除^[14]。在 MNIST、CIFAR 和 ImageNet 等数据集上对采用剪枝方法获得的压缩模型进行分类和识别测试,结果表明,相比其他压缩方法获得的压缩模型,拥有更低的错误率^[15]。模型量化不改变模型结构,主要通过降低模型参数位宽来减少模型计算量,即将原始模型中的 32-bit 参数转换为 16-bit、8-bit 或 4-bit 等较低位宽的参数,但是量化造成的模型精度损失往往很难恢复。知识蒸馏通过将复杂模型中的知识迁移到小模型以达到模型压缩要求,首先需要确定知识迁移的标准,标准选择不当会发生“老师”模型中的关键信息丢失,“学生”模型无法学习到知识的情况,且目前只适用于分类问题^[16]。网络架构搜索通过设定搜索策略对模型进行结构搜索,减少网络冗余实现自动模型压缩。网络架构搜索需要先验地进行搜索策略设定,网络搜索的好坏取决于搜索策略和性能评估指标的选择,搜索过程耗时久,训练成本高。除此,也有研究通过张量和矩阵分解消除冗余计算,但是由于缺乏有力的理论支撑,模型压缩稳定性往往得不到保证^[13]。

2.2 基础模型选择

进行神经网络压缩时,首先需要选择一个基础模型,为了让压缩后的模型在保证检测速度的同时最大程度上提高检测精度,选用 YOLOV4 作为基础模型。YOLOV4 是 YOLO 系列目前最优秀的目标检测算法之一,共由 162 个卷积层组成。主干网络采用 CSPDarknet53 结构和 mish 激活函数,用于特征提取,而特征预测部分采用 LeakyReLU 激活函数。实验结果表明,特征提取和预测阶段使用不

同的激活函数可以有效地提高目标检测准确率^[4]。为了进一步提高目标识别精度,YOLOV4 采用 PANet 网络替换 V3 中的 FPN 结构,通过上采样将高层特征传递至低层网络并与 PANet 网络融合,再对最底层特征进行上采样后与该层特征进行融合。同时 YOLOV4 的多尺度特征尺寸变为 76×76 、 38×38 、 19×19 ,多路特征复用策略的采用和预测尺度分辨率的提高,有效地保证了模型的检测能力和精度。

YOLOV4 是基于 anchor 的目标检测算法,通过将输入图像划分为 $S \times S$ 的网格,规定在每个网格中心对落在该网格中的物体进行目标检测。在每个网格上设定先验框(anchor box),通过预测 anchor box 和真实框(GT)之间的距离,进行最终的边界框(bbox)回归。但是 YOLOV4 的 anchor 是基于 Pascal VOC 数据集获得的,VOC2012 数据集包括 20 个类别图像,训练集规模为 11540,包含 27450 个对象实例,物体外形尺寸差异很大。YOLOV4 选用的 anchor box 为($[12 \ 16]$, $[19 \ 36]$, $[40 \ 28]$, $[36 \ 75]$, $[76 \ 55]$, $[72 \ 146]$, $[142 \ 110]$, $[192 \ 243]$, $[459 \ 401]$),直接使用默认的 anchor 进行水下目标检测会对模型收敛和检测精度产生影响。为了保证压缩后的模型精度,需要尽可能地提高基础模型检测精度,本文基于 K-means++ 算法对样本数据集标注后的 GT 进行聚类分析,确定属于目标数据自身的 anchor,通过设定起始聚类中心两两距离尽可能远的条件,可以有效地避免 K-means 必须手动设定 K 值的弊端。

$$\begin{cases} d(T_{\text{box}}, T_{\text{cen}}) = 1 - R_{\text{IoU}}(T_{\text{box}}, T_{\text{cen}}) \\ R_{\text{IoU}} = \frac{T_{\text{box}} \cap T_{\text{bbox, Pre}}}{T_{\text{box}} \cup T_{\text{bbox, Pre}}} \end{cases}, \quad (1)$$

式中: R_{IoU} 为 GT 与预测框的交并比; d 为类别距离; T_{cen} 为 K-means++ 算法计算出的 bbox; T_{box} 为图像上标注的 GT; $T_{bbox,Pre}$ 为预测的真实框。

通过(1)式,可以实现对数据集中的 GT 进行聚类分析,为了更加准确地判断出目标数据集中的 anchor 个数,对聚类结果进行性能分析,表达式为

$$A = \operatorname{argmax} \frac{\sum_{k=1}^K R_{IoU}(T_{box}, T_{cen})}{K}, \quad (2)$$

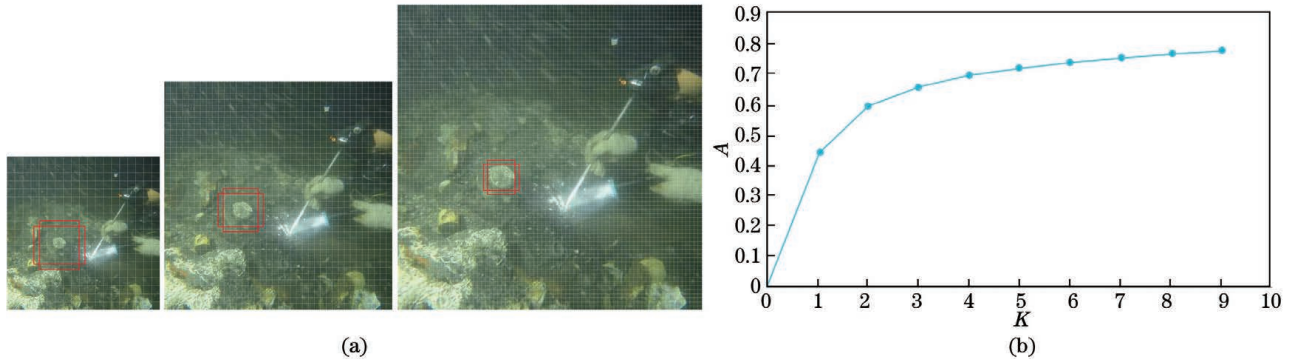


图 1 YOLOV4 分析。(a)三种尺寸特征图与其对应的 anchor 原理图;(b)聚类分析结果

Fig. 1 YOLOV4 analysis. (a) Three kinds of dimensional feature diagrams and their corresponding anchor schematic diagrams; (b) cluster analysis result

2.3 压缩算法设计

目前用于神经网络压缩的算法各有优缺点,在一定条件和范围内可以达到压缩效果,但是没有一种算法可以同时兼顾各种应用场景^[16]。本文在保证压缩模型检测精度的同时为了可以方便灵活地进行压缩比例设定,结合基础网络结构和嵌入式设备应用场景,选用剪枝方法对基础模型进行压缩。当下主流目标检测算法广泛采用 Batch Normalization (BN)结构对卷积输出进行归一化处理,保证每个卷积层的输入输出呈相同分布,进而加快网络收敛。BN 结构在网络结构中位于卷积层之后,可以在不增加任何参数的情况下利用 BN 层上的缩放因子 γ 判断通道和卷积层的重要性^[17]。目前基于 γ 进行剪枝主要有设定全局卷积层通道裁剪比例和裁剪阈值两种。裁剪比例指的是对稀疏模型中每个卷积层按照一定的比例进行通道剔除,裁剪阈值是指将卷积层上低于裁剪阈值的通道全部剔除。按照比例进行裁剪会发生对重要卷积层“过剪”,对不重要的卷积层“少剪”的现象。按照阈值进行裁剪,需要逐层分析模型同时进行先验阈值设定,基础模型一改变,剪枝策略就需要随之改变,且阈值设定往往没有标准。为获得最佳的压缩模型,选择基于稀疏约束的模型剪枝方法对所选的基础

式中: A 为准确度。

根据图 1(a)可知,随着特征图分辨率的增加,网络对小目标的识别精度逐渐提高。由图 1(b)聚类分析结果可知,当 K 取值为 8 时,聚类准确度接近 80%,随着 K 值继续增加,准确度变化不大。因此本文仍选择 anchor 数量为 9, anchor box 重定义为 $[[38 \ 36], [60 \ 77], [112 \ 84], [118 \ 136], [233 \ 137], [145 \ 221], [270 \ 204], [214 \ 274], [289 \ 287]]$ 。

网络进行核剪枝和层间剪枝的双向压缩。压缩过程分为基础训练、稀疏训练、模型剪枝和微调 4 个步骤^[18]。

Step 1: 基础训练,选择 YOLOV4 作为 baseline, L_{base} 为其损失函数,对目标数据集进行基础训练,当 $0 < L_{base}(\text{baseline}) \leq 1$ 且损失变化稳定时停止基础训练。

Step 2: 选用 BN 层上缩放因子 γ 衡量通道重要性,对 γ 引入 L_1 正则化,

$$L_1(\gamma^i) = |\gamma^{i1}| + \dots + |\gamma^{i\eta}| + \dots + |\gamma^{ic}|, \quad (3)$$

式中: L_1 范数空间由线段组成,可以产生稀疏值; $\gamma^{i\eta}$ 为网络中第 i 个卷积层上的第 η 通道的缩放因子; c 为通道总数。

Step 3: 稀疏训练,在 baseline 的损失函数后加上 γ 的 L_1 正则化项,对模型参数更新进行约束,构成模型稀疏训练过程中新的损失函数 L 。

$$L = L_{base}(\text{baseline}) + \alpha \sum g(\gamma), \quad (4)$$

式中: $g(\gamma)$ 为缩放因子 γ 的 L_1 正则化; α 为用于控制正则化程度的惩罚系数。

Step 4: 核剪枝,稀疏训练后网络中的某些通道上的 γ 会趋于 0,对各通道上的 γ 进行排序,设定初始通道裁剪阈值为 τ_c 。

$$\begin{cases} \tau_c = \gamma^i[\text{floor}(0.8c)], & \gamma^i[\text{floor}(0.8c)] < \tau_c, \\ \tau_c, & \gamma^i[\text{floor}(0.8c)] \geq \tau_c, \end{cases} \quad (5)$$

式中: $\text{floor}(\cdot)$ 为向下取整操作, 将 $\gamma^{ij} \leq \tau_c$ 的通道及其连接移除。

Step 5: 层间剪枝, 计算稀疏模型中每个卷积层上的 γ 平均值, 即 $\bar{\gamma}^i = \frac{1}{c} \sum_{j=0}^c \gamma^{ij}$ 。

$$\bar{\gamma}^i = 0, \min_{e=0}^E \{\bar{\gamma}^i\} + \phi < \tau_{\text{lay}}, \quad (6)$$

式中: E 为 baseline 的层数; ϕ 为阈值调节系数。设定初始层间裁剪阈值为 τ_{lay} , 对 γ 全为 0 的层及其连接进行剔除。

Step 6: 模型微调, 对核剪枝和层间剪枝后得到的压缩模型重新进行超参数设定, 在原数据集上对压缩模型进行再次训练, 微调后得到的模型作为最终的压缩模型 Comp_YOLOV4。

由于每层网络训练结果不同, 在核剪枝阶段采用分段剪枝率可以有效避免剪掉卷积层下所有通道, 造成网络结构破坏, 同时利用通道裁剪阈值对不满足要求的通道进行裁剪。对核剪枝后的模型中每个卷积层的重要性进行排序, 对不重要的层及其层间连接再次进行层间剪枝。相比较直接比较权重幅值策略, 所提剪枝策略能够更有效地鉴别网络中的重要部分, 既考虑层间通道比例又考虑其重要性, 进一步提高了模型压缩率。

3 数据准备与采集图像预处理

3.1 目标数据集建立

古代日用瓷器包括碗、碟、盘、瓶、壶、盆、罐等, 种类繁多, 形态各异。为提高水下遗迹目标检测模型的泛化能力, 将瓷器中高度矮、底座小、开口直径大的碗、碟、盘和盆归为一类, 命名“china”; 将瓷器中形态细长、开口直径小、中部突出的瓶、壶和罐归为另一类, 命名“chinavase”。为建立可供模型训练和测试的水下陶瓷图像数据集, 通过 3 种方式收集原始图像: 1) 对水下机器人采集的水下遗迹视频进行目标图像截取; 2) 从水下遗迹发掘和考古纪实视频中截取目标图像; 3) 利用数据挖掘从网络上下载相关图像。

采用 3 种方式共收集 3328 张“chinavase”类别图像和 7247 张“china”类别图像, 按照 85%:15% 的分配方式进行图像训练集和测试集划分。在模型训练阶段, 采用 Mosaic 数据增强方式在线增加数据集目标丰富度, 让有限的样本发挥最大的模型训练价值。Mosaic 数据增强一次后读入 4 张图像, 随机对原始图像进行裁剪, 然后再按随机顺序将 4 张图像拼接成一张完整的图像送入检测模型。与其他在线数据增强方式相比, Mosaic 增强方式使得一次输入到网络中的图像由原来的 1 张变成为 4 张, 有效地增加了图像中的目标样本数量。除进行在线数据增强外, 从原始数据集中随机选取一半的图片, 对其进行旋转、裁剪、色调调整、平移和缩放, 进一步增加训练样本的丰富度。数据集图像如图 2 所示。

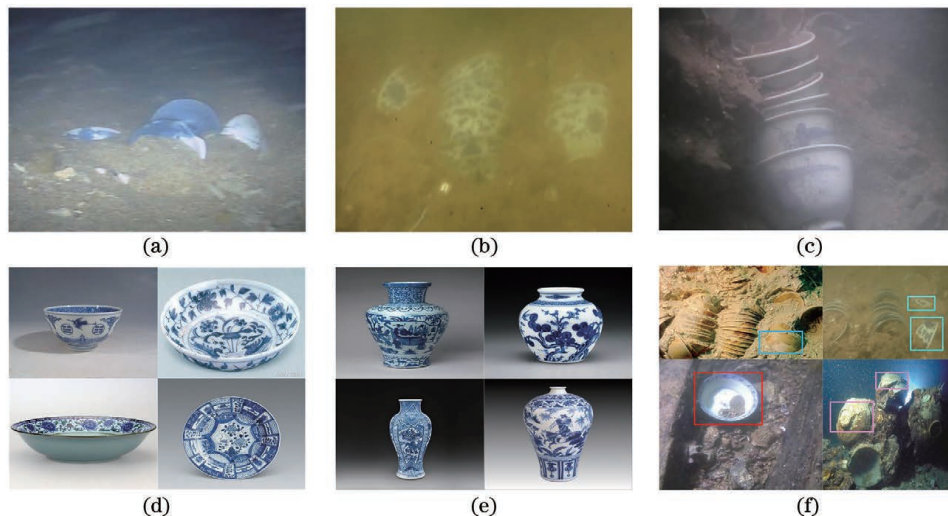


图 2 数据集图像。(a)水下机器人采集的图像;(b)“小白礁”遗迹发掘图像;(c)数据挖掘图像;(d)“china”类别;(e)“chinavase”类别;(f) Mosaic 在线数据增强

Fig. 2 Images in the dataset. (a) Image collected by UUV; (b) image of “xiaobaijiao” relic; (c) data mining image; (d) “china” category; (e) “chinavase” category; (f) Mosaic online data augmentation

3.2 关键帧图像在线增强

受水中悬浮颗粒及水对光的选择性吸收影响,水下图像往往存在模糊、色偏等问题^[19]。在相同情况下,采集的图像越清晰,目标检测精度越高。为提高 UUV 在线目标检测精度,对 UUV 采集的视频图像进行在线增强,将增强后的图像送入压缩后的模型中进行在线目标检测。UUV 进行水下探测时速度往往较慢,所采集的视频中相邻帧图像之间会存在很高的相似性,若对每一帧图像都进行增强会增加硬件计算负荷和系统消耗。为降低 UUV 目标识别过程中的资源消耗同时提高检测精度,基于 SSIM 对 UUV 采集的视频进行关键帧图像选择,剔除掉重复度高的图像,对剩余关键帧图像进行在线增强,以提高水下目标检测精度。关键帧图像选择算法流程如图 3 所示。

算法中 S 为采集的视频, Ψ 为视频总帧数, S_0 为采集视频的第一帧图像, K' 为参考帧, G 为关键帧图像集合。结构相似性 $S_{SIM}(u, v)$ 表示为

$$S_{SIM}(u, v) = \frac{(2\mu_u\mu_v + C_1)(2\sigma_{uv} + C_2)}{(\mu_u^2 + \mu_v^2 + C_1)(\sigma_u^2 + \sigma_v^2 + C_2)}, \quad (7)$$

式中: μ_u, μ_v 分别为 S 中第 u 帧和第 v 帧图像均值; σ_u, σ_v 分别为 u 和 v 的方差; σ_{uv} 为 u, v 之间的相关系数; C_1, C_2 为防止等式分母为 0 的常数。

Algorithm: key frame image selection

Input: $S, \xi=0.9, \omega=1, j=1, S_0=K'$

Output: G

$G_0=S_0$

for iteration from 1 to Ψ

if $S_{SIM}(S_i, K) \in [\xi, \omega]$

$K'=S_i$

else

$G_j=S_i$

$j++$

end for

return $G = \{G_0, G_1, \dots, G_j\}$

图 3 关键帧图像选择算法

Fig. 3 Key frame image selection algorithm

从集合 G 中随机选择 m 张图像,将图像无损地预处理到 $640 \text{ pixel} \times 480 \text{ pixel}$ 大小,选择限制对比度自适应直方图均衡化(CLAHE)、直方图均衡化(HE)、非监督性水下图像增强(UCM)^[20]和水下暗通道(UDCP)^[21]4 种水下图像增强算法对集合 G 中的图像进行增强。选取 $m=100$,在同一设备上使用 CLAHE、HE、UCM 和 UDCP 增强 100 张图像的平均时间消耗分别为 26.78 ms、21.92 ms、2190 ms 和 2391 ms,增强结果如图 4 所示。

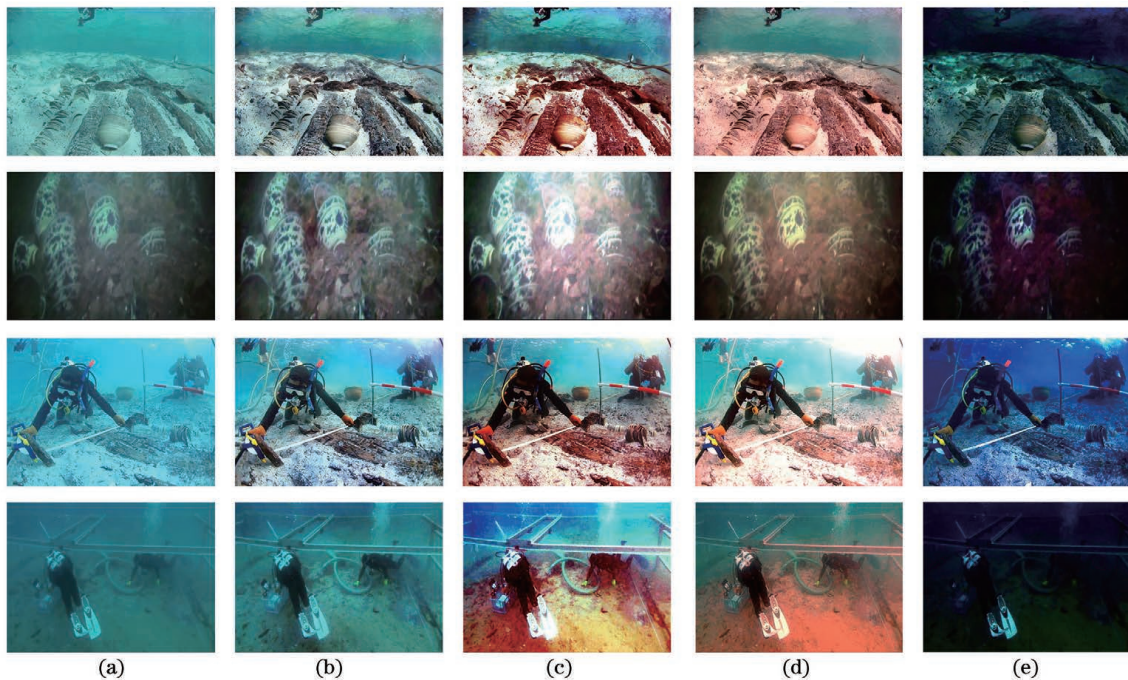


图 4 图像增强结果。(a)原图;(b) CLAHE 增强;(c) HE 增强;(d) UCM 增强;(e) UDCP 增强

Fig. 4 Enhanced results. (a) Original images; (b) CLAHE enhancement; (c) HE enhancement; (d) UCM enhancement; (e) UDCP enhancement

由图 4 可见:采用 HE 算法,增强后的图像整体亮度提高,但是颜色失真明显;UCM 在直方图滑动拉伸的基础上通过颜色修正进行图像增强,但是对通道补偿的误差过大,造成图像过度增强;UDCP 利用蓝绿色通道进行水下图像复原建模,对红色通道的补偿存在误差,导致增强后的图像亮度较低;CLAHE 算法通过设定剪切限制,对超过限制的直方图进行截取,将截取的直方图平均到全局像素灰度级上,增加对比度,提高了图像的清晰度。基于以上分析,选用 CLAHE 增强,时间短、效果好,能够满足 UUV 实时在线关键帧图像增强要求。

4 实验与分析

4.1 模型基础训练

为进一步分析基础模型性能,选用准确率 P 、召回率 R 、交并比 R_{IoU} 、 F_1 分数、平均准确度均值 m_{AP} 、模型损失 L_{loss} 和模型体积 V 进行模型性能评价。

$$\begin{cases} P = \frac{N_{TP}}{N_{TP} + N_{FP}} \\ R = \frac{N_{TP}}{N_{TP} + N_{FN}} \\ F_1 = \frac{2PR}{P + R} \end{cases}, \quad (8)$$

表 2 模型基础训练性能统计

Table 2 Statistics of the basic training performance of models

Model	N_{TP}	N_{FP}	N_{FN}	$P / \%$	$R / \%$	$R_{IoU} / \%$	$F_1 / \%$	$m_{AP} / \%$	L_{loss}	V / MB
YOLOV3	829	83	321	91	72	79.92	80	77.51	0.4514	246.3
YOLOV3-SPP	652	31	498	95	57	82.27	71	68.34	0.4270	250.5
YOLOV3-tiny	356	319	794	53	31	35.99	39	26.65	1.0500	34.7
YOLOV4	1292	68	210	95	86	86.74	90	88.34	0.4047	256.0

由表 2 可见,在 4 种模型中,YOLOV4 的 m_{AP} 远高于其他模型,在相同训练次数下,模型收敛效果最好、模型综合性能最佳。因此选用 YOLOV4 作为基础模型可以更好地进行水下遗迹目标检测。

4.2 模型多粒度剪枝

由表 2 可知,基础训练后的 YOLOV4 模型体积较大,为实现 UUV 实时在线目标检测,对基础训练后的 YOLOV4 模型进行多粒度剪枝压缩。设定模型稀疏训练的迭代次数为 500,正则化程度的惩罚系数、剪枝阈值设定如表 3 所示,对稀疏训练后的稀疏模型进行微调,微调初始学习率为 0.001,当迭代次数达到总次数的 80% 时,学习率降低 1/10,其他参数设置与基础训练保持一样。微调后的模型性

式中: N_{TP} 为将目标识别为正样本; N_{FP} 为将非目标识别为正样本; N_{TN} 为将非目标识别为负样本; N_{FN} 为将目标识别为负样本。准确率 P 反映模型对目标预测的准确性,召回率 R 反映模型对目标检测的完整性。

$$m_{AP} = \frac{1}{C} \sum_{h=1}^C E(P, R, h), \quad (9)$$

式中: m_{AP} 是衡量模型对所有类别检测结果优良的指标; C 为检测类别数量,取 2; E 为准确度均值函数,为 P - R 曲线与坐标轴所围面积。

选用 YOLOV3、YOLOV3-SPP、YOLOV3-tiny 和 YOLOV4 进行模型基础训练。由于目标数据集样本较少,直接进行基础训练难以获得具有泛化性的训练结果,采用迁移学习策略,将各模型在 ImageNet 数据集上训练后的权重作为模型基础训练的初始化权重。调整输入图像的分辨率为 608×608 ,设定 Batch size 为 16,动量为 0.949,初始学习率为 0.001,权值衰减率为 0.0005,总迭代次数为 8000。训练硬件平台配置为 64 G Intel i7-9700K 处理器,显存为 8 G 的 Nvidia RTX2080 独立显卡。模型基础训练性能如表 2 所示。

能统计如表 3 所示,表中 model-m-n 表示使用压缩后的模型和权重进行微调,model-pr-n 为采用压缩后的模型和基础训练后的权重进行微调, n 为微调迭代次数。

表 3 中, $\alpha = 10^{-3}/10^{-4}$ 表示正则化局部衰减,在稀疏训练前 249 次迭代中正则化惩罚系数为 10^{-3} ,而后迭代中惩罚系数取值 10^{-4} 。由表 3 可得,在三种不同剪枝阈值下,模型压缩比分别为 13.98、17.90 和 22.65。在相同微调策略和剪枝阈值设定下,正则化局部衰减可以加速模型收敛,提高模型精度。但随着通道剪枝阈值变小,层间剪枝阈值变大,模型收敛变慢,模型精度开始降低,表明模型裁剪过多,微调后很难提高模型精度。在正则化

表 3 压缩模型性能统计

Table 3 Statistics of the performance for compression models

Model	α	τ_c	τ_{lay}	ϕ	$F_1 / \%$	$m_{AP} / \%$	L_{loss}	V / MB
YOLOV4-m-8000	10^{-3}	0.10	0.5	0.05	78	75.50	0.7100	18.5
YOLOV4-pr-8000	10^{-3}	0.10	0.5	0.05	77	77.90	0.4418	18.5
YOLOV4-m-8000	$10^{-3}/10^{-4}$	0.10	0.5	0.05	85	77.98	0.6123	14.3
YOLOV4-pr-8000	$10^{-3}/10^{-4}$	0.10	0.5	0.05	86	80.47	0.5606	14.3
YOLOV4-m-8000	$10^{-3}/10^{-4}$	0.08	0.5	0.1	73	71.32	1.0513	11.3
YOLOV4-pr-8000	$10^{-3}/10^{-4}$	0.08	0.5	0.1	74	73.82	1.1832	11.3

局部衰减压缩策略下,采用 model-pr 模式得到的 Comp_YOLOV4 模型的综合性能最优。在该模式下模型收敛效果好, L_{loss} 和 F_1 与基础模型的基本一

致,在测试集上的检测精度可达 80.47%。Comp_YOLOV4 模型结构如图 5 所示。

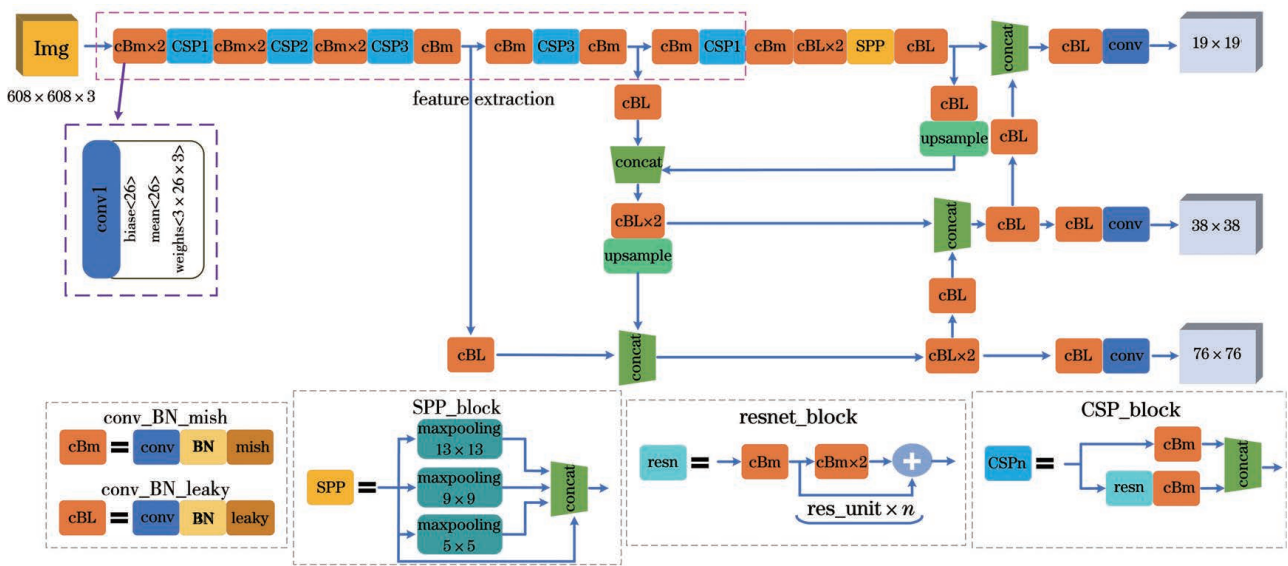


图 5 Comp_YOLOV4 结构

Fig. 5 Structure of Comp_YOLOV4

由图 5 可知,Comp_YOLOV4 共有 104 个卷积层,层间剪枝主要对特征提取部分进行修剪,去除了 CSPnet 中的冗余结构,模型预测尺度与基础模型保持一致,预测部分卷积层数量变少。核剪枝对卷积层上的通道进行裁剪,如图 5 所示,基础模型第一个卷积层原来有 32 个通道,压缩后变为 26 个,从卷积层和卷积层通道两个层面上实现了对基础模型的压缩,模型层数和通道数量的减少是实现检测速度提升的重要保证。

4.3 嵌入式平台目标检测

选用 Nvidia Jetson TX2 嵌入式图像处理器作为实验平台,采用 Python 语言进行程序编写和脚步运行。TX2 核心板尺寸为 50 mm×87 mm,典型负载下的功耗仅为 7.5 W,CPU 采用 ARM Cortex-

A57,GPU 为 Nvidia Pascal GPU,满足 UUV 对处理器的功耗和体积要求。

如图 6 所示,关键帧选择算法与视频采集放在一起进行,在水下相机内部完成。图像增强是目标检测算法中的预处理部分,即属于压缩模型中的一部分,集成在检测网络中。当视频采集开始后首先进行关键帧挑选,将关键帧无损地预处理到大小为 640×480 分辨率。然后将固定大小的图片送入到压缩模型,使用 CLAHE 对其进行在线图像增强,

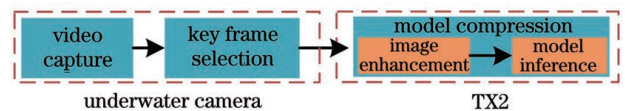


图 6 检测过程

Fig. 6 Process of detection

使用 Comp_YOLOV4 算法对增强后的图像进行目标检测。压缩后的模型在 TX2 上的整体推理速度如表 4 所示,其中 Billion Floating Operations (BFLOPS)表示模型运算复杂度,avg_FPS 表示平均检测速度,采用平均检测速度而非视频中某一

表 4 目标检测速度对比

Table 4 Speed comparison of the target detection

Model	avg_FPS / (frame·s ⁻¹)	BFLOPS
YOLOV4	3.2	106.752
Comp_YOLOV4	18.2	10.588

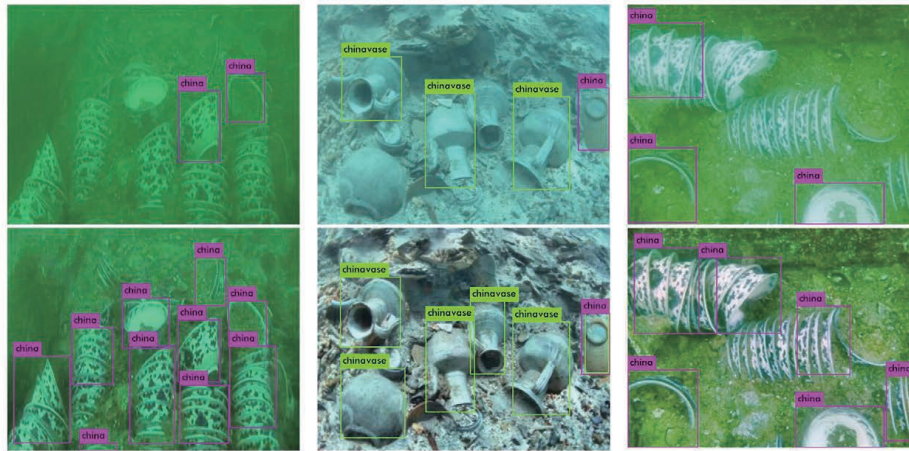


图 7 模型检测结果

Fig. 7 Target detection results of the models

5 结 论

针对 UUV 水下遗迹探寻实际需求,提出一种基于核剪枝和层间剪枝的压缩模型。对于水下遗迹检测中的目标选择问题,选择水下瓷器作为目标检测对象,利用数据增强提高数据泛化能力,通过迁移学习加速基础模型收敛。为保证压缩模型的精度,选用 YOLOV4 作为基础模型,对采集视频进行关键帧选择并对其采用 CLAHE 进行实时图像增强,降低 UUV 系统资源消耗的同时提高了目标检测精度。针对现有目标检测算法难以部署到嵌入式设备上的问题,通过通道分段剪枝阈值设定和全局卷积层剪枝阈值设定,在模型保证精度和结构完整性的前提下实现对 YOLOV4 的双向模型剪枝。通过对比基础模型与其压缩后的模型,可以得出,模型速度的提高是以降低模型体积和复杂度为基础的,进而去除模型中不重要的卷积层和通道,但是当模型裁剪幅度过大时,模型微调后很难提高精度。实验结果表明,剪枝策略压缩后的 YOLOV4 模型能够以较高的准确度和检测速度在嵌入式设备上水下

时刻检测速度,可以更加全面地描述算法的实时检测性能^[22]。

由表 4 可知,未经压缩的基础模型在嵌入式设备上的处理速度只有 3.2 frame/s,过高的模型复杂度降低了检测速度。压缩后的模型复杂度显著降低,在 TX2 上对于输入为 640×480 分辨率的图像检测速度是基础模型的 5.7 倍,达到 18.2 frame/s,可基本实现 UUV 通过嵌入式平台进行实时目标检测的需求。检测结果如图 7 所示,第一行是未经图像增强的检测结果,第二行为图像增强后的检测结果。

遗迹目标探测。

参 考 文 献

- [1] Lin S, Zhao Y. Review on key technologies of target exploration in underwater optical images[J]. Laser & Optoelectronics Progress, 2020, 57(6): 060002. 林森, 赵颖. 水下光学图像中目标探测关键技术研究综述[J]. 激光与光电子学进展, 2020, 57(6): 060002.
- [2] Ren S Q, He K M, 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.
- [3] He K M, Gkioxari G, Dollár P, et al. Mask R-CNN [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, 42(2): 386-397.
- [4] Dai J F, Li Y, He K M, et al. R-FCN: object detection via region-based fully convolutional networks [EB/OL]. (2016-05-20) [2020-10-15]. <https://arxiv.org/abs/1605.06409v1>.
- [5] Bochkovskiy A, Wang C Y, Liao H Y. YOLOv4: optimal speed and accuracy of object detection [EB/

- OL]. (2020-04-23)[2020-07-01]. <https://arxiv.org/abs/2004.10934>.
- [6] Redmon J, Farhadi A. YOLO9000: better, faster, stronger[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 21-26, 2017, Honolulu, HI, USA. New York: IEEE Press, 2017: 6517-6525.
- [7] Redmon J, Farhadi A. YOLOv3: an incremental improvement[EB/OL]. (2018-04-08)[2020-07-01]. <https://arxiv.org/abs/1804.02767>.
- [8] Liu W, Anguelov D, Erhan D, et al. SSD: single shot MultiBox detector[M]//Leibe B, Matas J, Sebe N, et al. Computer vision-ECCV 2016. Lecture notes in computer science. Cham: Springer, 2016, 9905: 21-37.
- [9] Lin T Y, Goyal P, Girshick R, et al. Focal loss for dense object detection[C]//2017 IEEE International Conference on Computer Vision (ICCV), October 22-29, 2017, Venice, Italy. New York: IEEE Press, 2017: 2999-3007.
- [10] He K M, Zhang X Y, Ren S Q, et al. Deep residual learning for image recognition[C]//2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 27-30, 2016, Las Vegas, NV, USA. New York: IEEE Press, 2016: 770-778.
- [11] Huang G, Liu Z, van der Maaten L, et al. Densely connected convolutional networks [C] // 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 21-26, 2017, Honolulu, HI, USA. New York: IEEE Press, 2017: 2261-2269.
- [12] Chen Y, Li J, Xiao H, et al. Dual path networks [EB/OL]. (2017-08-01) [2020-07-01]. <https://arxiv.org/abs/1707.01629>.
- [13] Goel A, Tung C, Lu Y H, et al. A survey of methods for low-power deep learning and computer vision[C]//2020 IEEE 6th World Forum on Internet of Things (WF-IoT), June 2-16, 2020, New Orleans, LA, USA. New York: IEEE Press, 2020: 1-6.
- [14] Anwar S, Sung W Y. Coarse pruning of convolutional neural networks with random masks [EB/OL]. (2016-10-30) [2020-07-01]. <https://arxiv.org/abs/1610.09639v1>.
- [15] Yin W F, Liang L Y, Peng H M, et al. Research progress on convolutional neural network compression and acceleration technology[J]. Computer Systems & Applications, 2020, 29(9): 16-25.
尹文枫, 梁玲燕, 彭慧民, 等. 卷积神经网络压缩与加速技术研究进展[J]. 计算机系统应用, 2020, 29(9): 16-25.
- [16] Hinton G, Vinyals O, Dean J. Distilling the knowledge in a neural network[EB/OL]. (2015-03-09) [2020-07-01]. <https://arxiv.org/abs/1503.02531>.
- [17] Liu Z, Li J G, Shen Z Q, et al. Learning efficient convolutional networks through network slimming [C]//2017 IEEE International Conference on Computer Vision (ICCV), October 22-29, 2017, Venice, Italy. New York: IEEE Press, 2017: 2755-2763.
- [18] Lin Y, Tu Y, Dou Z. An improved neural network pruning technology for automatic modulation classification in edge devices[J]. IEEE Transactions on Vehicular Technology, 2020, 69(5): 5703-5706.
- [19] Liu K, Li X J. De-hazing and enhancement methods for underwater and low-light images[J]. Acta Optica Sinica, 2020, 40(19): 1910003.
刘柯, 李旭健. 水下和微光图像的去雾及增强方法[J]. 光学学报, 2020, 40(19): 1910003.
- [20] Iqbal K, Odetayo M, James A, et al. Enhancing the low quality images using unsupervised colour correction method[C]//2010 IEEE International Conference on Systems, Man and Cybernetics, October 10-13, 2010, Istanbul, Turkey. New York: IEEE Press, 2010: 1703-1709.
- [21] Drews P, do Nascimento E, Moraes F, et al. Transmission estimation in underwater single images [C]//2013 IEEE International Conference on Computer Vision Workshops, December 2-8, 2013, Sydney, NSW, Australia. New York: IEEE Press, 2013: 825-830.
- [22] Wang X Q, Wang X J. Real-time target detection method applied to embedded graphic processing unit [J]. Acta Optica Sinica, 2019, 39(3): 0315005.
王晓青, 王向军. 应用于嵌入式图形处理器的实时目标检测方法[J]. 光学学报, 2019, 39(3): 0315005.