

机器学习大作业

水下目标检测

Underwater object detection

作者姓名: 杨题鸣

学科、专业: 电子信息工程

学号: 201883016

完成日期: 2021. 06. 02

大连理工大学

Dalian University of Technology

摘要

. 随着目标检测的发展，越来越多的领域开始运用目标检测。其中当海洋观测的快速发展，水下物体检测别在海军沿海防御任务以及渔业、水产养殖等海洋经济中发挥着越来越重要的作用。水下目标检测的目的是对水下场景中的目标进行定位和识别。该研究因其在海洋学、水下导航、养鱼等领域的广泛应用而不断受到人们的关注。然而，由于水下环境和照明条件的复杂性，这仍然是一项具有挑战性的任务。基于深度学习的目标检测系统在各种应用中都表现出了良好的性能，但在处理水下目标检测方面仍显不足。这是因为，首先，水下探测数据集是稀缺的，可用的水下数据集和实际应用中的对象通常较小。目前基于深度学习的检测器不能有效地检测小目标。其次，现有的水下数据集和实际应用中的图像都是杂乱的。在水下场景中，波长依赖性的吸收和散射会显著降低水下图像的质量。这导致了许多问题，如能见度损失，弱对比和颜色变化，这给检测任务提出了许多挑战。为了研究这些问题，我们采用 mmdetection 当中的 cascade rcnn 检测目标类别包括海参 “holothurian” ，海胆 “echinus” ，扇贝 “scallop” 和海星 “starfish” 四类。

1 绪论

机器学习包含了深度学习，在深度学习当中计算机视觉处于占主导地位。在计算机视觉当中，目标检测可谓是其一个标志性的成功。近几年来，目标检测算法取得了很大的突破。比较流行的算法可以分为两类，一类是基于 Region Proposal 的 R-CNN 系算法（R-CNN, Fast R-CNN, Faster R-CNN[3]等），它们是 two-stage 的，需要先算法产生目标候选框，也就是目标位置，然后再对候选框做分类与回归。而另一类是 Yolo[6]，SSD[7]这类 one-stage 算法，其仅仅使用一个卷积神经网络 CNN 直接预测不同目标的类别与位置。第一类方法是准确度高一些，但是速度慢，但是第二类算法是速度快，但是准确性要低一些。

目标检测推动了海洋观测的快速发展，目标检测成为了观测海洋的主力军。水下图像是海洋信息的重要载体，目标检测作为当前人工智能领域的学术热点，应用前景非常广泛。该技术对于军事活动、资源勘测、海洋噪声污染保护等方面有着巨大的作用。水下光学图像目标检测赛项代表了人工智能和水下机器人技术在未来深度融合的方向。基于深度学习的目标检测系统在各种应用中都表现出了良好的性能，但是在处理水下目标检测方面仍显不足。这是因为，首先，水下探测数据集是稀缺的，可用的水下数据集和实际应用中的对象通常较小。目前基于深度学习的检测器不能有效地检测小目标。其次，现有的水下数据集和实际应用中的图像都是杂乱的。为了让水下目标检测有一个更好的发展，本作业将会对目标识别进行展开实验。我们采用 mmdetection[1] 当中的

cascade rcnn 检测目标类别包括海参 “holothurian” , 海胆 “echinus” , 扇贝 “scallop” 和海星 “starfish” 四类。最终展示成 CSV 格式以及可视化展示目标检测结果

实验主要分为以下几个部分：1. 正文部分：介绍 baseline 模型 cascade rcnn; 介绍在实验当中使用的 trick。2. 实验部分：介绍水下目标检测数据集；介绍实验环境代码配置 ；介绍实验结果。

2 相关工作

A. mmdetection 框架[1]

. 在本次实验当中，我们采用 mmdetection 目标检测框架。此框架是由，商汤科技（2018 COCO 目标检测挑战赛冠军）和香港中文大学开源的一个基于 Pytorch 实现的深度学习目标检测工具箱 mmdetection，支持 Faster-RCNN, Mask-RCNN, Fast-RCNN, Cascade-RCNN 等主流的目标检测框架

B. Cascade-RCNN[2]

. Cascade R-CNN 算法是 CVPR2018 的文章，通过级联几个检测网络达到不断优化预测结果的目的，与普通级联不同的是，cascade R-CNN 的几个检测网络是基于不同 IOU 阈值确定的正负样本上训练得到的，这是该算法的一大亮点。Cascade R-CNN 经常用在一些比赛当中，并取得了出色的结果。

3 正文

本节当中，我们首先解释我们的 baseline 模型，然后介绍我们在水下目标检测当中使用的方法，用于提升检测效果的一些 trick, trick 可以列为如下所示 1、Mixup 2、DCN 与多尺度训练测试 3、global context ROI 4、旋转数据增强 5、韵达模糊 6、attention 模块。

3.1 Baseline 模型

3.1.1 Object Detection

. 我们使用的 baseline 为 mmdetection 目标检测框架当中的 cascade rcnn 模型。再解释 cascade rcnn 之前，可以先由 Figure 1 得知几种网络结构。

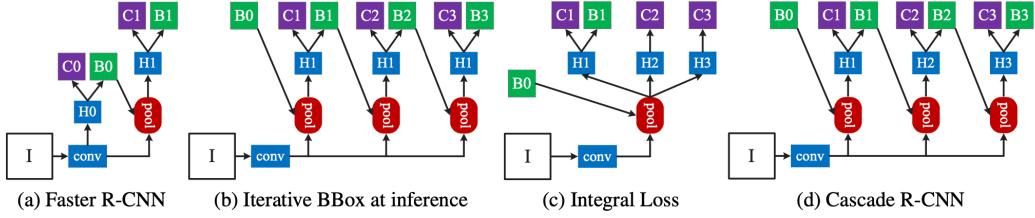


Fig. 1. 不同框架的架构。“I”为输入图像，“conv”为骨干卷积，“pool”为区域特征提取，“H”为网络头，“B”为边界框，“C”为分类。“B0”是所有架构中的建议。

A. Faster R-CNN

. 目前经典的 two-stage 架构如图 Figure 1(a)。第一阶段是一个提框的子网 H_0 ，用于生成初步的 bndbox。第二阶段为特定区域处理的检测子网 H_1 ，给定 bndbox 最终的分类分数 C 和 bndbox 坐标 B

B. Iterative BBox at inference

. 有的研究者认为单次的 box regress 是不足以产生准确的位置信息的，因此需要进行多次迭代来精调 bndbox，这就是 iterative bounding box regression：

$$f'(x, \mathbf{b}) = f \circ f \circ \cdots \circ f(x, \mathbf{b}) \quad (1)$$

实现如图 Figure 1 (b) 所示，所有的 head 都是一样的，但是这个方法忽略了两个问题：

Figure 2 所示，detector ($u=0.5$) 对于所有的高质量的 bndbox 是次优解，甚至降低了 IoU 大于 0.85 的 bndbox 的准确度

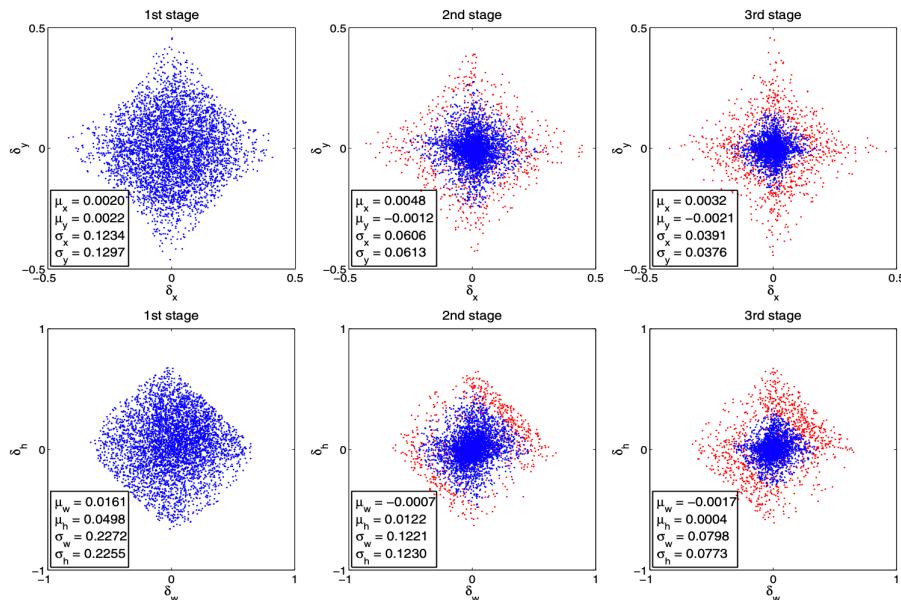


Fig. 2. 不同级联阶段的顺序 Δ 分布(未归一化)。红点为增加欠条阈值时的异常值，剔除异常值后得到统计数据。

Figure 2 为 bndbox 的 (x, y, w, h) 与 GT 间的差值分布，从图中可以看出，不同阶段的 bndbox 分布是显著不同的。若 regressor 对于初始化的分布是最优的，那对于在后面的阶段肯定是次优的

因此，iterative BBox 需要大量的手工操作，如 box voting，而其结果不是稳定提升的。通常来说，对 bndbox 进行多于两次相同的 regressor 是几乎没有收益的

C. Integral Loss

. 由于 bndbox 经常包含目标和一定的背景，因此很难去判定当前 bndbox 是否正样本

$$y = \begin{cases} g_y, & \text{IoU}(x, g) \geq u \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

常用的方法是判断其与 GT 的 IoU，当 IoU 大于阈值时，则赋予其对应 GT 的 label。但是阈值的设定是十分苦难的，当阈值过高时，正样本包含很少的背景，但是会导致难以生成足够多的正样本进行训练，反之，则会导致 detector 容易产生 close false positives。因此，很难找到一个单独的 classifier 能一致地对所有 IoU 的 bndbox 是最优的

$$L_{cls}(h(x), y) = \sum_{u \in U} L_{cls}(h_u(x), y_u) \quad (3)$$

一种尝试的方法是使用一个分类器集合，Figure 1(c) 所示，优化针对各种质量的 bndbox 的 loss。 $U = 0.5, 0.55, \dots, 0.75$ 是根据 COCO 设定 IoU 阈值合集，按照定义，分类器在推理时再进行组装

这种解决方法存在两个问题：1. 不同的 classifier 的正样本数量是不一样的，如图 4 所示，正样本的数量随着 u 的提高显著下降，这意味着高质量的 classifiers 容易过拟合 2. 在推理时，高质量的 classifiers 需要处理相对低质量的 bndbox，而他们对这些 bndbox 并没有优化因此，Integral loss 在很多 IoU 水平难以表现出高的准确率。相对于原始的 two-stage 架构，Integral loss 的架构收益相对较小

3. 1. 2 Cascade R-CNN[2]

. A. Cascaded Bounding Box Regression

由于很难训练一个能应付所有 IoU 水平的 regressor，可以把回归任务分解成一个级联的 regression 问题，架构 Figure 1(d) 所示

$$f(x, \mathbf{b}) = f_T \circ f_{T-1} \circ \dots \circ f_1(x, \mathbf{b}) \quad (4)$$

其中， T 是级联阶段数，每个 regressor f_t 对于当前的级联输入都是最优的，随着阶段的深入，bndbox 在不断的提升。

cascade regression 与 iterative BBox 有以下区别：1. iterative BBox 是后处理的方法，而 cascaded regression 是能够改变 bndbox 分布的重采样过程
2. cascaded regression 在训练和推理时是一致的，不存在区别 3. cascaded regression 的多个 regressor 对于对应阶段的输入分布是最优的，而 iterative BBox 仅对初始分布是最优的

$$\begin{aligned}\delta_x &= (g_x - b_x)/b_w, \quad \delta_y = (g_y - b_y)/b_h \\ \delta_w &= \log(g_w/b_w), \quad \delta_h = \log(g_h/b_h)\end{aligned}\quad (5)$$

Bndbox 在回归时，为了对 scale 和 location 有不变性，将对坐标的学习转移到对坐标差值的学习。由于坐标插值通常较小，因此将其进行归一化 $\delta' = (\delta_x - \mu_x)/\sigma$ ，以权衡定位与分类的 loss。Cascade R-CNN 在每一个 stage 结束后，都会马上进行计算这些均值/方差

B. Cascaded Detection

. 产生 Cascade R-CNN 的启发点主要有两个：

1. Figure 2 的 1st stage 图所示，初始的 bndbox 分布大多落在低质量的区域，这对于高质量 classifiers 来说是无效的学习 2. 在图 Figure 1(c) 实验中可以看到，所有的曲线都高于对角线，即 regressor 都倾向于能够提升 bndbox 的 IoU。

因此，以集合 (x_i, b_i) 作为开始，通过级联 regress 来产生高 IoU 的集合 (x'_i, b'_i) 。如图 4 所示，这种方法能在提升样本整体 IoU 水平的同时，使得样本的总数大致维持在一个水平，这会带来两个好处：1. 不会存在某个阈值的 regressor 过拟合。2. 高阶段的 detector 对于高 IoU 阈值是最优的

$$L(x^t, g) = L_{cls}(h_t(x^t), y^t) + \lambda[y^t \geq 1]L_{loc}(f_t(x^t, b^t), \mathbf{g}) \quad (6)$$

在每一个阶段 t，都独立一个对阈值 $u_t (u_t > u_{t-1})$ 最优的 classifier h_t 和 regressor f_t ， $b^t = f_{t-1}(x^{t-1}, b^{t-1})$ 是上一阶段的输出， $\lambda = 1$ 是权重因子， $[y^t \geq 1]$ 是指示函数，表示背景的 L_{loc} 不加入计算。与 integral loss 不同，公式 8 保证了顺序地训练 detectors 来逐步提高 bndbox 质量。在推理时，bndbox 的质量

是顺序提高的，高质量的 detectors 只需要面对高质量的 bndbox。

3.2 实验当中使用的水下目标检测 trick

A. Mixup[4]

. mixup 是一种运用在计算机视觉中的对图像进行混类增强的算法，它可以将不同类之间的图像进行混合，从而扩充训练数据集。

mixup 原理可以解释为：假设 batch x_1 是一个 batch 样本，batch y_1 是该 batch 样本对应的标签；batch x_2 是另一个 batch 样本，batch y_2 是该 batch 样

本对应的标签， λ 是由参数为 α, β 的贝塔分布计算出来的混合系数，由此我们可以得到 mixup 原理公式为

$$\lambda = \text{Beta}(\alpha, \beta)$$

$$\text{mixed_batch}_x = \lambda * \text{batch}_{x1} + (1 - \lambda) * \text{batch}_{x2} \quad (7)$$

$$\text{mixed_batch}_y = \lambda * \text{batch}_{y1} + (1 - \lambda) * \text{batch}_{y2}$$

其中 Beta 指的是贝塔分布， mixed_batch_x 是混合后的 batch 样本， mixed_batch_y 是混合后的 batch 样本对应的标签。

在本实验当中，实验效果图可以展示为 Figure 3

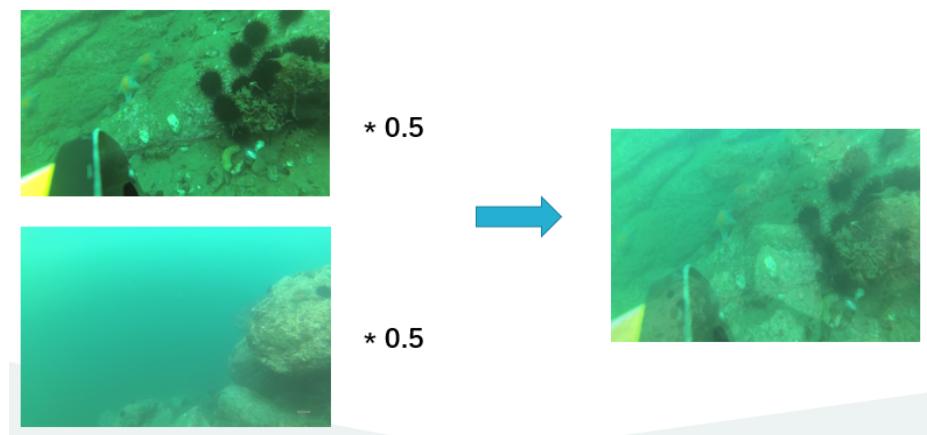


Fig. 3. 水下目标检测 Mixup 效果图

B. Deformable Convolutional Networks

. 核心思想：不管是 deformable convolution（可变性卷积）还是 deformable RoI pooling（可变形 RoI 池化），主要通过引入 offset（偏移，指的是位置的偏移量），使得特征提取过程能够更加集中于有效信息区域（类似于视觉注意力机制将网络的注意力集中在感兴趣区域）。

C. 多尺度训练测试

. 输入图片的尺寸对检测模型的性能影响相当明显，事实上，多尺度是提升精度最明显的技巧之一。在基础网络部分常常会生成比原图小数十倍的特征图，导致小物体的特征描述不容易被检测网络捕捉。通过输入更大、更多尺寸的图片进行训练，能够在一定程度上提高检测模型对物体大小的鲁棒性，仅在测试阶段引入多尺度，也可享受大尺寸和多尺寸带来的增益。训练时，预先定义几个固定的尺度，每个 epoch 随机选择一个尺度进行训练。测试时，生成几个不同尺度的 feature map，对每个 Region Proposal，在不同的 feature map 上也有不同的尺度，我们选择最接近某一固定尺寸（即检测头部的输入尺寸）的 Region Proposal 作为后续的输入。多尺度训练测试原理框图如 Figure 4 所示。

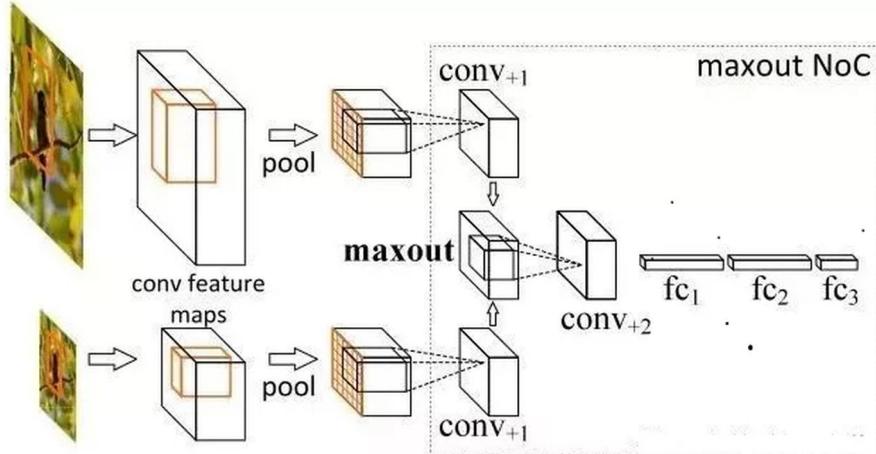


Fig. 4. 多尺度训练测试原理框图

D. global context ROI[8]

- . 全局上下文利用场景配置作为对象检测的额外信息源。对于早期的对象检测器，集成全局上下文的一种常见方法是集成组成场景的元素的统计摘要，如 Gist。对于现代的基于深度学习的检测器，有两种方法来集成全局上下文。第一种方法是利用大的接受域（甚至大于输入图像）或 CNN feature 的全局池化操作。第二种方法是将全局上下文看作一种序列信息，并使用递归神经网络学习它。

E. 旋转数据增强

- . 旋转数据增强是一个无损的方法进行数据增强。在实验当中，我们采取随机旋转 90 度的方法，增加数据的鲁棒性。

F. 运动模糊

- . 移动模糊也就是物体移动时会产生模糊的现象，可以说是相对观察相机快速变动的物体（与其说是物体，还不如说是相应的片元 fragment）产生的残影与物体本体叠加在一个范围，产生的模糊现象。在实验当中，由于实际拍摄当中存在运动，增加运动模糊模块极大的提升了数据的鲁棒性。

G. attention 机制[6]

- . attention 机制是模仿人类注意力而提出的一种解决问题的办法，简单地说就是从大量信息中快速筛选出高价值信息。主要用于解决 LSTM/RNN 模型输入序列较长的时候很难获得最终合理的向量表示问题，做法是保留 LSTM 的中间结果，用新的模型对其进行学习，并将其与输出进行关联，从而达到信息筛选的

目。计算机视觉的相关应用中大概可以分为两种：1. 学习权重分布：输入数据或特征图上的不同部分对应的专注度不同 2. 任务聚焦：通过将任务分解，设计不同的网络结构（或分支）专注于不同的子任务，重新分配网络的学习能力，从而降低原始任务的难度，使网络更加容易训练。

4 实验

本部分，将首先介绍实验使用的数据集，其次对实验环境进行介绍，以及最终实验结果的展示。

4.1 水下目标检测数据集

随着海洋观测的快速发展，水下物体检测在海军沿海防御任务以及渔业、水产养殖等海洋经济中发挥着越来越重要的作用，而水下图像是海洋信息的重要载体，本次比赛希望参赛者在真实海底图片数据中通过算法检测出不同海产品（海参、海胆、扇贝、海星）的位置。数据由鹏城实验室提供。 7600 幅训练图像（含人工标注真值数据），数据集结构如下：

Table 1. 数据集结构表

一级目录	二级目录	解释说明	格式
train/	image/	所有训练图片	jpg
	box/	同名图像文件的对应标注结果	xml

其中 train/image 文件夹中包含所有训练数据，这些图片之间不存在帧间连续性。图片路径示例如下：train/image/000001.jpg，其对应的目标检测标注真值位于路径 train/box/000001.xml 文件中，该文件包含了对应图像中所有物体的类别以及目标框参数（位置和尺寸）。本届比赛需检测的目标类别包括海参“holothurian”，海胆“echinus”，扇贝“scallop”和海星“starfish”四类。

如 Figure 5 所示，为水下目标检测数据集展示图：

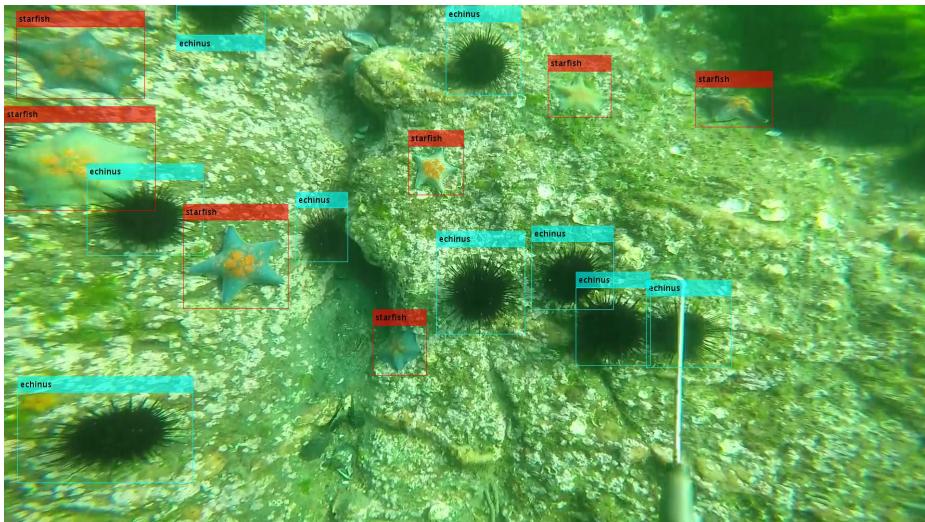


Fig. 5. 水下目标检测数据集数据样例

4.2 实验环境以及代码配置

实验环境

.OS:Ubuntu

GPU: 2080Ti * 8

python: python3.7

nvidia 依赖:

cuda: 10.0

cudnn: 7.5.1

nvidia driver version: 430.14

deeplearning 框架: pytorch1.1.0

代码配置

训练框架 mmdetection 以及模型 cascade rcnn

Backbone: mmdetection 官方开源 htc resnext64×4d 预训练模型

(具体见 github 代码: <https://github.com/yangtiming/underwater>)

4.3 实验结果

实验最终按照生成 CSV 文件，csv 文件中的每一行对应一个检测结果。

生成的结果格式可以展示为如下 Table 2:

Table 2. 生成 CSV 结果格式

字段名称	字段含义	字段类型
name	检测类别	string
image_id	图像编号 (.jpg 之前的编号)	string
confidence	置信度	float
xmin	检测结果左上角 x 值	int
ymin	检测结果左上角 y 值	int
xmax	检测结果右下角 x 值	int
ymax	检测结果右下角 y 值	int

部分结果可以展示为如下 Table 3:

Table 3. 部分数据展示

name	image_id	confidence	xmin	ymin	xmax	ymax
echinus	212	0.000115	3662	928	3785	1032
echinus	212	0.00010551	3365	1699	3487	1745
starfish	212	0.00022828	2697	1726	3060	1959
holothurian	41	0.00391553	2198	384	2419	508
holothurian	41	0.00103047	2218	427	2406	516
scallop	41	0.00524434	772	1093	859	1152
scallop	41	0.00496623	1529	878	1604	929

随机选取测试集进行数据可视化展示如图 Figure 6:

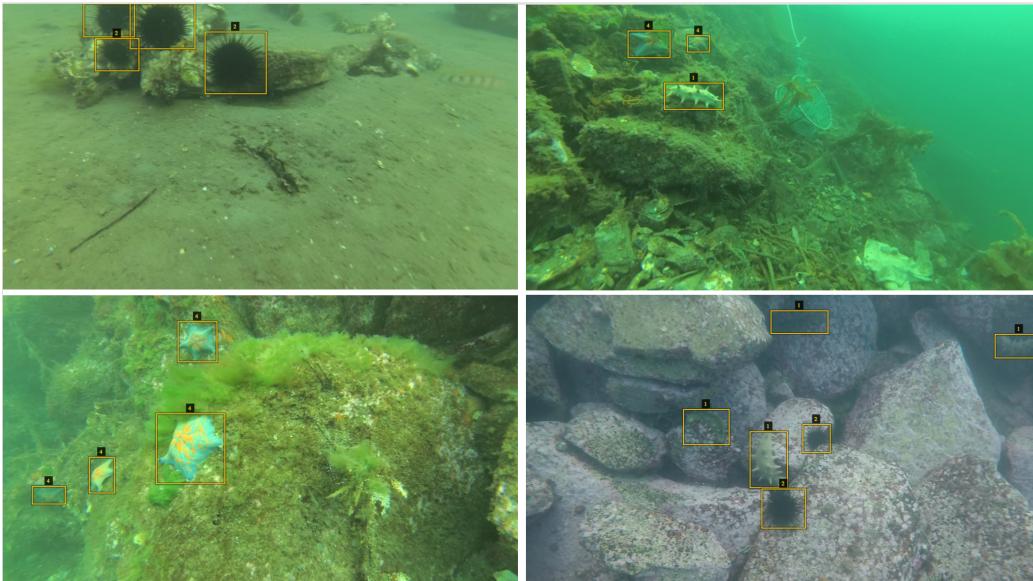


Fig. 6. 水下目标检测部分结果展示图

结 论

本实验当中使用 mmdetection 框架当中的 cascade rcnn 水下目标检测取得了很好的结果。深度学习网络结合我们使用的 trick: 1、Mixup 2、DCN 与多尺度训练测试 3、global context ROI 4、旋转数据增强。5. 运动模糊 6. attention 机制 能部分缓解数据集是稀缺，水下数据集和实际应用中的对象通常较小，环境杂乱等问题。目标检测用途广泛，未来还有许多的应用空间值得我们挖掘。

参 考 文 献

- Chen K, Wang J, Pang J, et al. MMDetection: Open mmlab detection toolbox and benchmark[J]. arXiv preprint arXiv:1906.07155, 2019.
- Cai Z, Vasconcelos N. Cascade r-cnn: Delving into high quality object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 6154–6162.
- Ren S, He K, Girshick R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[J]. arXiv preprint arXiv:1506.01497, 2015.
- Zhang H, Cisse M, Dauphin Y N, et al. mixup: Beyond empirical risk minimization[J]. arXiv preprint arXiv:1710.09412, 2017.
- Wang X, Cai Z, Gao D, et al. Towards Universal Object Detection by Domain Attention[C]//CVPR. 2019.
- Bochkovskiy A, Wang C Y, Liao H Y M. Yolov4: Optimal speed and accuracy of object detection[J]. arXiv preprint arXiv:2004.10934, 2020.

7. Liu W, Anguelov D, Erhan D, et al. Ssd: Single shot multibox detector[C]//European conference on computer vision. Springer, Cham, 2016: 21–37.
8. Zhong Q, Li C, Zhang Y, et al. Cascade region proposal and global context for deep object detection[J]. Neurocomputing, 2020, 395: 170–177.