An Underwater Organism Image Dataset and a Lightweight Module Designed for Object Detection Networks

JIAFENG HUANG, TIANJUN ZHANG, SHENGJIE ZHAO, and LIN ZHANG, School of Software Engineering, Tongji University, China
YICONG ZHOU, Department of Computer and Information Science, University of Macau, China

Long-term monitoring and recognition of underwater organism objects are of great significance in marine ecology, fisheries science and many other disciplines. Traditional techniques in this field, including manual fishing-based ones and sonar-based ones, are usually flawed. Specifically, the method based on manual fishing is time-consuming and unsuitable for scientific researches, while the sonar-based one, has the defects of low acoustic image accuracy and large echo errors. In recent years, the rapid development of deep learning and its excellent performance in computer vision tasks make vision-based solutions feasible. However, the researches in this area are still relatively insufficient in mainly two aspects. First, to our knowledge, there is still a lack of large-scale datasets of underwater organism images with accurate annotations. Second, in consideration of the limitation on hardware resources of underwater devices, an underwater organism detection algorithm that is both accurate and lightweight enough to be able to infer in real time is still lacking. As an attempt to fill in the aforementioned research gaps to some extent, we established the Multiple Kinds of Underwater Organisms (MKUO) dataset with accurate bounding box annotations of taxonomic information, which consists of 10,043 annotated images, covering eighty-four underwater organism categories. Based on our benchmark dataset, we evaluated a series of existing object detection algorithms to obtain their accuracy and complexity indicators as the baseline for future reference. In addition, we also propose a novel lightweight module, namely Sparse Ghost Module, designed especially for object detection networks. By substituting the standard convolution with our proposed one, the network complexity can be significantly reduced and the inference speed can be greatly improved without obvious detection accuracy loss. To make our results reproducible, the dataset and the source code are available online at https://cslinzhang.github.io/MKUO-and-Sparse-Ghost-Module/.

CCS Concepts: • **Computing methodologies → Vision for robotics;**

Additional Key Words and Phrases: Benchmark dataset, object detection, lightweight module.

This work was supported in part by the National Natural Science Foundation of China under Grant 61936014, Grant 62272343, and Grant 61973235; in part by the Shanghai Science and Technology Innovation Plan under Grant 20510760400; in part by the Shuguang Program of Shanghai Education Development Foundation and Shanghai Municipal Education Commission under Grant 21SG23; and in part by the Fundamental Research Funds for the Central Universities.
Authors' addresses: J. Huang, T. Zhang, S. Zhao (Corresponding author), and L. Zhang (Corresponding author), School of Software Engineering, Tongji University, No. 4800, CaoAn Rd., Shanghai 200092, China; e-mails: 2010195@tongji.edu.cn, 1911036@tongji.edu.cn, shengjiezhao@tongji.edu.cn, cslinzhang@tongji.edu.cn; Y. Zhou, Department of Computer and Information Science, University of Macau, Taipa University Road, Macau 999078, China; e-mail: yicongzhou@um.edu.mo.

ACM 1551-6857/2024/02-ART147
https://doi.org/10.1145/3640465

ACM Reference Format:
Jiafeng Huang, Tianjun Zhang, Shengjie Zhao, Lin Zhang, and Yicong Zhou. 2024. An Underwater Organism Image Dataset and a Lightweight Module Designed for Object Detection Networks. *ACM Trans. Multimedia Comput. Commun. Appl.* 20, 5, Article 147 (February 2024), 23 pages. https://doi.org/10.1145/3640465

1 INTRODUCTION

The ocean is the lifeline of the earth. According to data from the National Oceanic and Atmospheric Administration, the ocean covers 70% of the earth's surface, and oceanographers believe that in this huge underwater field, more than 80% of the area is still unobserved and undeveloped [35]. As a natural ecosystem made up of biomes in the ocean and their surrounding environment, marine ecosystem affects climate, regulates temperature, ultimately supports all living things, and plays a crucial role in the ecological balance of nature. Many related disciplines have been born to deepen the study of this ecosystem and to better get along with it. For example, marine ecology, as the most crucial part of marine biology, explores the laws of the marine ecosystem by studying the reproduction, growth, distribution and quantitative changes of marine organisms in the marine environment, as well as the interaction between organisms and the environment, in order to provide a scientific basis for the development, utilization, management and enrichment of marine biological resources and the protection of the marine environment and ecological balance. Because of its uniqueness and importance, the study of the marine ecosystem has received increasing attention from countries around the world in recent years. Among these related studies, long-term monitoring and recognition of underwater organism objects, as an important source of data for many upper-level studies, has attracted extensive attention.

There are quite a few traditional methods for long-term monitoring and recognition of underwater organism objects, but they usually cannot fully satisfy both academic and industrial requirements more or less. Taking the most commonly used method in fishery science as an example, operators use traditional marine fishing techniques such as longline fishing and trawling to catch marine organisms in fixed sites in each season and then complete subsequent data analysis. Fish-behavior researchers will analyze the behavioral responses of fishes under the influence of external tools (such as fishing nets) and the relationship between fishing yield and towing time, towing methods, and hydrological changes. The analysis results can be applied to the improvement of fishing gear design and fishing operation principles. Although fishing efficiency has been improved greatly with the development of automatic fishing tools, it's still time-consuming and laborious for researchers. On the one hand, marine fishing occupies a significant amount of researchers' time, which obviously slows down the progress of fishery scientific research. On the other hand, the fishing process consumes a lot of human and material resources, which is usually unaffordable for most researchers. Thus, in most cases, they can only use historical fishing data instead of first-hand data, which will bring inconvenience to research work and weaken the reliability of the results.

The other type of traditional mainstream method, the sonar-based method, relies on sonar equipment to emit acoustic pulse signals and obtain underwater target echo signals. Due to the different physical characteristics between the water environment and the target, echo transducers can convert sonar electrical signals into acoustic signals and then emit them into the water. When acoustic waves propagate in the water and encounter targets (such as fish, underwater buildings, sunken ships, etc.), some of the acoustic signals will undergo refraction, scattering, or absorption by the target, while the other part, also known as an echo signal, will be reflected back and received by the transducer. The transducer then converts the echo signal into the electrical signal and transmits it back to the data processing unit. After the processing, it is displayed in the form of an analyzable echo map. In underwater acoustics, the most popular fish resource assessment method currently

relies on acoustic image processing methods using Multi-beam Imaging Sonar equipment. Such a method has many advantages compared with the fishing-based method, such as no harm to organisms, no damage to the environment, high efficiency, and low consumption of manpower and resources, and can provide long-term real-time observations. However, the acoustic image formed by the projection of the echo signal usually has low accuracy [34]. Zwolinski et al. [60] found experimentally that the standard deviation of hydroacoustics-based fish amount estimation was about 20%, and the study of Appenzeller and Leggett [2] showed that the deviation was even greater, up to about 50%, when fish gathered to produce acoustic shadows. Compared with the aforementioned traditional methods, deploying underwater recording equipment to monitor underwater organisms and conducting further analysis via computer vision techniques is much more reliable and efficient.

A large-scale underwater organism dataset with accurate annotations is of great significance to the vision-based underwater organism analysis. Unfortunately, to the best of our knowledge, such a dataset is still lacking. At present, the existing public datasets either lack accurate annotations due to the limited quality of images acquired in the natural environment, or cover relatively few species. As an attempt to fill in the research gaps to some extent, we acquired data over several weeks from a subtropical aquarium, which simulates the underwater scene of the natural ocean, and established a large-scale dataset, named **MKUO (Multiple Kinds of Underwater Organisms)**. In addition, the research on underwater organism detection based on vision is still relatively lacking, and the explored methods could not fully meet the requirements of high-precision detection and real-time inference. In this article, we attempt to partly tackle this problem and propose a novel lightweight module designed for object detection networks. Substituting the standard convolution in an object detection network by our proposed one can markedly reduce the network complexity and improve the inference speed without observable detection accuracy loss.

The contributions of this article are summarized below:

(1) A public underwater organism dataset MKUO (Multiple Kinds of Underwater Organisms) which contains 10,043 images annotated in bounding box form is proposed. The dataset covers multiple kinds of underwater organisms, including *fish, jellyfish, hermit crab, lobster, turtle, limulus, sea anemone, seahorse, and so on.* The captured organisms are classified according to taxonomy, covering 84 species. As far as we know, it covers the broadest categories compared with other existing annotated underwater organism datasets. Some typical examples of the proposed dataset are offered in Figure 1, where original images and their annotated versions of underwater organisms are shown.

(2) A baseline evaluation of the existing underwater organism detection algorithms and a batch of novel object detection algorithms is conducted on our proposed dataset. The evaluation results on a series of metrics, such as the detection accuracy, the inference speed and the parameter amount are provided for the reference of relevant researchers. Our work will be helpful for subsequent researchers to select the appropriate object detection algorithm as the backbone network for underwater organism detection tasks according to their own research needs.

(3) A novel lightweight module designed for object detection networks, namely Sparse Ghost Module, is proposed. By substituting the standard convolution with it, the number of parameters and the model size of the object detection network could be effectively reduced and the inference speed would also be improved with almost no accuracy loss. Thus, the improved network will occupy less memory and storage space and could better meet the real-time inference requirements, which is beneficial for the deployment of underwater monitoring equipment.

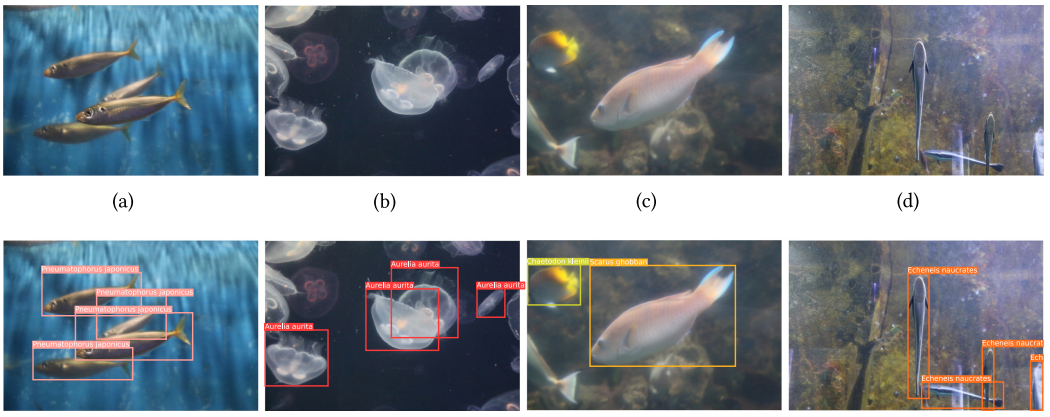


Fig. 1. Some typical samples of our MKUO dataset. In each column, the image without annotations is shown on the top, and the corresponding result with annotations is shown at the bottom. From left to right, the samples of *Pneumatophorus japonicus*, *Aurelia aurita*, *Scarus ghobban* and *Echeneis naucrates* are given, respectively.

The rest of this article is organized as follows. Section 2 introduces related studies and our contributions. Section 3 provides details of the dataset. Section 4 introduces our evaluation experiment on the dataset. Section 5 details the lightweight module and experiments we have done. Finally, Section 6 concludes the article.

2 RELATED WORK

2.1 Underwater Organism Datasets

In the past few decades, many underwater organism datasets have been released. In this article, we summarize these datasets mainly into two categories, the unannotated datasets and annotated ones.

For the unannotated datasets, Van Horn et al. [46] proposed a comparatively well-known dataset consisting of 369 fish images collected from *InaturaList*[1] which is a social network of naturalists, citizen scientists, and biologists built on the concept of mapping and sharing observations of biodiversity across the globe. Wildfish [58] and Wildfish++ [59] are two large-scale unlabeled fish benchmark datasets recently proposed by Zhuang et al. They are composed of 54,459 fish images classified into 1,000 categories and 103,034 fish images classified into 2,348 categories, respectively. The data of these two datasets are mainly obtained from various professional fish knowledge websites, including *FishBase*[2], *Florida Museum*[3], *Discover Life*[4], *Encyclopedia of Life*[5], *Shorefishes*[6], *Fishes of Australia*[7], *Underwater Photography-Fish Database*[8], and search engines such as *Google*[9] and

[1] *InaturaList*: https://www.inaturalist.org
[2] *FishBase*: http://www.fishbase.org
[3] *Florida Museum*: https://www.floridamuseum.ufl.edu
[4] *Discover Life*: http://www.discoverlife.org
[5] *Encyclopedia of Life*: http://eol.org
[6] *Shorefishes*: http://biogeodb.stri.si.edu/sftep
[7] *Fishes of Australia*: http://fishesofaustralia.net.au
[8] *Underwater Photography-Fish Database*: http://www.fishdb.co.uk
[9] *Google Images*: https://images.google.com

Flickr[10]. These aforementioned datasets are large enough to cover a large number of fish species, but due to the lack of object-level annotations, they are only applicable to the research of fish image classification algorithms.

In addition to the above-mentioned unannotated fish datasets, there are also some public annotated datasets. One of the most popular underwater datasets used for fish detection is the **F4K** (**Fish4Knowledge**) dataset [5]. It was recorded by ten cameras set up in Taiwan, China from 2010 to 2013 and consisted of videos and images with various marine fish and accurate annotations. Besides, it is also used as part of the benchmark of the SEACLEF mission, which is a sub-task of the series challenge organized by the LifeCLEF lab every year from 2014 to 2017. During investigations of rocky seafloor environments in Southern California coastal waters, the Southwest Fisheries Science Center established another labeled wild fish dataset collected by forward-tilting digital cameras deployed on **Remotely Operated Vehicles (ROVs)** and named it Labeled Fishes in the Wild [9]. The other two worth mentioning fish datasets are the Croatian fish dataset [20] and the QUT fish dataset [1]. The Croatian fish dataset [20] consisted of cropped images of twelve different fish species. For the QUT fish dataset [1], except for the images collected underwater, observations of the out-of-the-water fishes are also covered. Sixteen species of fish images were collected in the widely used ImageNet [11]. As a large-scale general image detection database, ImageNet includes more than 1,000 categories of organisms besides fish labeled with common names. All images in the ImageNet have image-level labels, and some images have object-level labels in the form of bounding boxes. The Brackish dataset [38] contains 14,518 images collected from the saltwater straits, and several common underwater organisms such as *fish, crab, jellyfish, shrimp, and starfish* are observed. However, according to the authors' description, due to the turbidity of the seawater, even marine biologists with expertise in the local marine environment can only classify the captured fishes roughly as big fishes and small fishes. In addition to fish, benthos such as scallops and corals are also the key objects in the research of marine organisms. The HabCam dataset [7] contains 2.5 million annotated images, mainly scallops and a few starfish. These images were taken along the continental shelf of the east coast of the United States. The Tasmania Coral Point Count [14] was recorded in 22 diving missions using AUVs on the southeast coast of Tasmania in 2008. Another well-known annotated coral reef dataset is the Moorea Labeled Corals [3] which is a subset of the Moorea Labeled Corals Long Term Ecological Research [4] project in French Polynesia. Although the above datasets are rich enough, there are still some imperfections:

(1) The above datasets focus on a particular species of fish or coral, except for the Brackish dataset [38] which focuses on half-a-dozen different species of underwater organisms. In fact, in addition to cartilaginous fishes, bony fishes and corals, there are also many underwater organisms that marine scholars pay attention to. For example, jellyfish and sea anemones are cnidarians, hermit crabs and lobsters are arthropods, and so on.
(2) The annotation accuracy of datasets is also an important guarantee for related studies. The optimal classification way in our opinion is to classify organisms by scientific name according to taxonomy. Unfortunately, the ways of species classification of existing datasets are usually defective. Specifically, Labeled Fishes in the Wild [9] only distinguishes between fishes and non-fishes, the Brackish dataset [38] only roughly divides the categories such as big fish, small fish, and the like, while most other existing datasets use unofficial common names as class names for captured marine organisms.
(3) Some datasets, such as F4K [5] and Labeled Fishes in the Wild [9], use manual cropping to focus on one instance in the center of the image. This method emphasizes the characteristics

[10] *Flickr*: http://flickr.com

Table 1. An Overview of Underwater Organism Datasets

Underwater dataset	Year	Classes	Coverage	Instances/image	Images	Label
Fish4Knowledge [5]	2012	23	Fish	0/1	27,370	Masks
ImageNet$_{(FishClass)}$ [11]	2012	16	Fish	More than 1	21,134	Bounding boxes
Tasmania Coral Point Count [14]	2012	13	Coral	More than 1	1,258	Points
Moorea Labeled Corals [3]	2012	9	Coral	More than 1	2,055	Points
Labeled Fishes in the Wild [9]	2015	2 (fish or not)	Fish	0/1	3,167+2 videos	Bounding boxes
Croatian Fish Dataset [20]	2015	12	Fish	1	794	Bounding boxes
iNaturalist$_{(FishClass)}$ [46]	2018	369	Fish	1	8,942	Image-level
WildFish [58]	2018	1,000	Fish	1	54,459	Image-level
Brackish Dataset [38]	2019	6	Fish, Starfish, Shrimp, Jellyfish, Crab	More than 1	14,518	Bounding boxes
WildFish++ [59]	2020	2,348	Fish	1	103,034	Image-level
MKUO (Ours)	2022	84	Fish, Jellyfish, Hermit crab, Lobster, Turtle, Limulus, Sea anemone, Seahorse, etc.	More than 1	10,043	Bounding boxes

of the organism while disregarding the features of its surrounding environment to some extent. The absence of the background environment may lead to the detector relying solely on the appearance of underwater organisms, overlooking the influence of the underwater environment on organism classification. Furthermore, this approach may cause the model to overfit these images, leading to incorrect recognition of underwater organisms in diverse environments.

(4) Underwater organisms exhibit unique morphological characteristics, which are crucial for accurate identification and classification, when observed from specific perspectives. Capturing as many morphological characteristics as possible can enhance the diversity, applicability, and reliability of datasets. Unfortunately, existing underwater organism datasets are often captured from fixed perspectives and may contain incomplete or biased morphological characteristics. This limitation can impede the diversity of data, hinder subsequent research on ecology and species behavior diversity, and restrict the application scenarios of these datasets.

For the consideration of the above points to be improved, we propose a novel underwater organism dataset, namely MKUO (Multiple Kinds of Underwater Organisms), with bounding box formed annotations. Compared with the existing datasets, our dataset has the following characteristics: wide coverage range of species, precise species classification, high-quality annotations and comprehensive morphological characteristics. Due to its uniqueness, it is an important supplement to the existing annotated underwater organism datasets.

Table 1 summarizes the characteristics of typical existing underwater organism datasets and our proposed one, MKUO. In Section 3, our dataset will be introduced in further detail.

2.2 Underwater Organism Detection

In recent years, more and more research related to underwater organism detection has been reported. To classify coral reef fishes, Villon et al. [47] trained a traditional **Support Vector Machine (SVM)** classifier on the **Histogram of Oriented Gradients (HOG)** features and fine-tuned a **Convolutional Neural Network (CNN)**. Then, they evaluated the performance of these two methods and found that CNNs overwhelmingly outperformed traditional classification methods. Salman et al. [41] compared traditional classification methods such as SVM, **k-Nearest Neighbor (k-NN)** and **Sparse Representation Classifier (SRC)** with CNNs and got a similar conclusion. The average classification accuracy using CNNs on the LifeCLEF14 [18] and the LifeCLEF15 [19] fish datasets exceeded 90%, while the accuracies of traditional methods were much lower. Siddiqui et al. [42] trained a deep CNN with the cross-layer pooling trick to deal with the problem of limited labeled training data and achieved high performance for the classification of fish images.

Cai et al. [6] proposed a modified YOLOv3 [39] taking MobileNetv1 [17] as the backbone, which has the capability of providing the accurate number of fishes and can be used to determine the breeding actions accordingly in a real breeding farm. Wulandari et al. [51] compared the mainstream object detection networks (including Faster-RCNN [40], SSD [29], RetinaNet [26], YOLOv3 [39], etc.) on an underwater dataset, finding that all these models have their own advantages and disadvantages and there is no model fully suited for underwater object detection.

Except for fish, scallop detection and coral reef detection have also received a lot of research passions in the ocean monitoring field. One of the most prominent scallop detection algorithms was developed by Dawkins et al. [10] using a series of cascaded Adaboost classifiers. A more recent research study was presented by Ovchinnikova et al. [37] who employed a CNN detector to identify king scallop in images of the seabed. Their results strongly suggest that the application of machine learning and low-cost imaging are merited. Coral reefs have attracted plenty of interests from marine biologists around the world, but it's quite difficult to monitor them. To assist biologists, Mahmood et al. [32] fine-tuned VGGNet and made use of it to automatically analyze corals' cover at three sites in Western Australia. Another approach was investigated by Beijbom et al. [4] which achieved state-of-the-art detection performance by fusing standard reflection images with fluorescence images of corals using a 5-channel CNN. Recently, Soukup [43] proposed a method based on Mask RCNN for automatic annotation, localization and pixel-wise parsing of the coral reefs from underwater images.

As pioneering works, in the above-mentioned studies, researchers usually selected the classical object detection networks as backbones, and then further fine-tuned them for application in underwater organism detection tasks. However, the object detection networks they selected are quite different from the recently proposed algorithms which have made many unprecedented explorations or expansions, making them unique in terms of universality, time complexity or precision performance and other research concerns. Due to the lack of comprehensive comparison and evaluation of these algorithms, it is often difficult for researchers to choose the appropriate network as the solution according to their own research needs. In order to facilitate the follow-up research, we selected a series of object detection algorithms proposed since 2020 and also some classical algorithms (including Faster-RCNN [40], SSD [29], RetinaNet [26] and YOLOv3 [39]) and evaluated them on our MKUO dataset, so as to analyze the performance of different object detection algorithms on underwater organism images. The characteristics of some typical selected object detection networks are briefly introduced in turn below.

Aiming to improve the performance and generalization ability of the detection model, Kong et al. [22] overcame the limitation of the anchor box and proposed a novel anchor-free object detection framework, "FoveaBox". To tackle the difficulty of accurate localization when the distances between the anchors and the corresponding targets are large, Wang et al. [49] proposed an approach, named "**Side-Aware Boundary Localization**" (**SABL**), where each side of the bounding box is respectively localized with a dedicated network branch. "Dynamic RCNN" proposed by Zhang et al. [52] is able to adjust the label assignment criteria (IoU threshold) and the parameters of regression loss function (SmoothL1 Loss) automatically based on the statistics of proposals during training. "YOLOv5" [45] presented by Ultralytics is a group of object detection models pretrained on the COCO [27] dataset, which used various techniques to improve accuracy such as mosaic data enhancement, adaptive anchor frame calculation, adaptive image scaling, focus structure, cross-stage local network structure, and so on, and achieved superior performance. Zhang et al. [54] proposed an **IoU-aware Classification Score (IACS)** as a joint representation of object presence confidence and localization accuracy, and built a dense object detector, called "VFNet", which can offer an accurate ranking of all candidate detections based on the IACS. "Sparse RCNN" proposed by Sun et al. [44] is a completely sparse object detection method designed to make the

object detection network get rid of the limitations brought by dense anchors. Ge et al. [33] presented some experienced improvements to the YOLO series, formed a novel anchor-free detector, named "YOLOX", whose design is very exquisite and absorbs a lot of tricks from other former work, i.e., the decoupled head and the leading label assignment strategy SimOTA, to achieve state-of-the-art object detection results. In this article, in order to facilitate the follow-up research, we have made detailed analyses and comparisons of these methods on our MKUO dataset to provide sufficient support for subsequent related research.

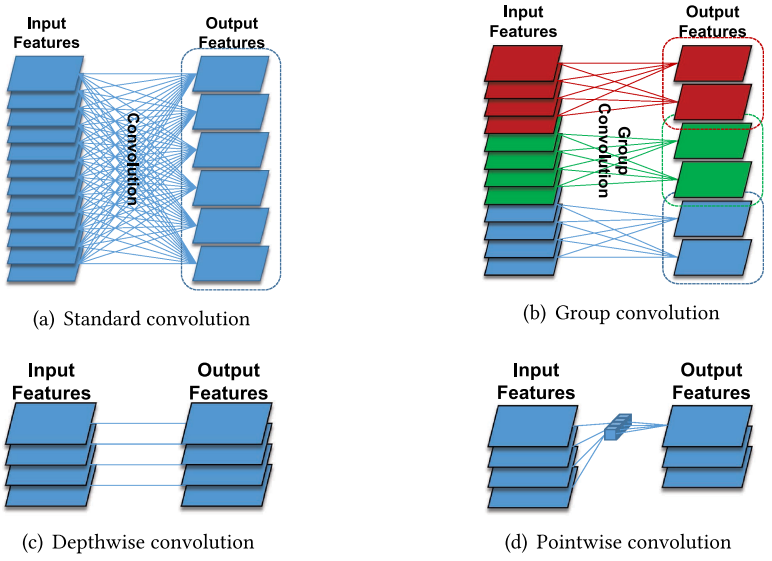
2.3 Lightweight Design of Object Detection Network

Although deep learning brings high benefits for feature extraction, classic networks are often accompanied by dependence on strong computing power and large memory space in the training process. In the underwater environment, the hardware performances of calculating units are usually limited, that is, they do not have the same memory space, storage space and computing power as those large-scale servers or workstations deployed on land. Usually, applying classical object detection networks to underwater organism detection tasks can meet the accuracy requirements, but encounter efficiency problems, which are ultimately storage problems and speed problems:

(1) ***The storage problem.*** Deep networks with hundreds of layers often have a large number of parameters. Saving and using these parameters require a huge amount of memory space and storage space. If the memory space and storage space of the device is insufficient, the application of the algorithm will be directly affected.
(2) ***The speed problem.*** In the practical application of object detection algorithms, in order to meet the real-time requirements, the single frame inference speed of millisecond level is usually required, which brings great challenges to the inference efficiency of the selected algorithm.

A forward inference of ResNet-152 [16], the model that for the first time surpassed human in image classification tasks, requires approximately 11.3 billion floating-point calculations (and consumes 240MB of storage). A recent natural language processing model, OpenAI's GPT-3 [36], has 175 billion parameters. Google's updated Switch Transformer even has 1.6 trillion parameters. Although such network architectures have great advantages in prediction accuracy, they are difficult to apply to an online practical environment. In order to facilitate deployment in actual scenarios, real-time object detectors for different devices have still been under development in recent years. For example, the development of MCUNet [25] and NanoDet-Plus [31] focused on the deployment on low-power single-chips and aimed to improve the inference speed on edge CPU. The YOLOX [33] focuses on achieving high inference speed on various GPUs. More recently, the research studies of real-time object detectors have focused on the design of efficient architectures. Existing real-time object detectors are mostly based on MobileNetv1 [17], ShuffleNet [57], or GhostNet [15], and their architectures can be optimized using the strategy proposed in CSPNet [48]. Besides, a series of tricks have also been proposed by researchers to build efficient network architectures, such as network pruning, knowledge distillation, and low-level quantization.

Another lightweight solution to this dilemma is to design more efficient basic modules. In 2012, Krizhevsky et al. [23] split the convolution operations to run them on two GPUs in parallel to speed up the inference. As the rudiment of group convolution, whose illustration is shown in (b) of Figure 2, this idea has been continuously developed and improved in the follow-up works. In 2017, multiple powerful lightweight CNN models such as SqueezeNet [30], MobileNetv1 [17], ShuffleNet [57], and Xception [8] were published one after another, greatly accelerating the development of this field. The Fire Module proposed by SqueezeNet [30] used to replace the common 3×3 convolution is composed of a squeeze layer with 1×1 convolution kernels and an expanded



(a) Standard convolution

(b) Group convolution

(c) Depthwise convolution

(d) Pointwise convolution

Fig. 2. Illustrations of (a) standard convolution, (b) group convolution, (c) depthwise convolution, and (d) pointwise convolution.

layer composed of 1×1 and 3×3 convolution kernels. It reduces the calculation and the number of parameters by using smaller convolution kernels. Depthwise Separable Convolution proposed by Google in MobileNetv1 [17] decouples a convolution operation into two steps, depthwise convolution and pointwise convolution. The schematic diagrams of these two convolution structures are shown in (c) and (d) of Figure 2. The former is the group convolution whose group number equals the number of input channels, which means that it uses a single convolution kernel for each input channel. The latter one applies a 1×1 convolution to linearly combine the outputs in the depthwise convolution, so as to integrate the features between different channels. In order to solve the problem that the feature information between groups is not interconnected in group convolution, ShuffleNet [57] explored a channel shuffle operation to shuffle the groups and improve the performance of group convolution. The lightweight modules in Xception [8] are similar to those in MobileNetv1 [17], but the operation order is opposite. It uses pointwise convolution and depthwise convolution, in turn, to fully decouple the cross-channel correlations and spatial correlations of the input feature mapping. In addition, this work also mentioned the feasibility of replacing the larger convolution kernel with the smaller convolution kernels to reduce the computational complexity (for example, substituting a 5×5 convolution kernel with two 3×3 convolution kernels). Two other important works proposed recently are GhostNet [15] and MicroNet [24]. By analyzing the feature maps of the input images generated by ResNet-50, the authors of GhostNet [15] found that there were many similar feature map pairs like overlays of each other. They advocate a cheaper linear operation to generate these similar feature maps to achieve lightweight models, and propose the Ghost Module. The Ghost Module divides a standard convolutional layer into two parts. The first part involves ordinary convolutions but the total number of kernels will be rigorously controlled. Based on the feature maps obtained from the first part, a series of simple linear operations (the linear operation here used in practice is group convolution as shown in (b) of Figure 2) are used for generating more feature maps. MicroNet [24] studies how to decompose the dense convolution process into sparse ones with the idea of matrix decomposition, so as to integrate sparse connectivity into convolution without reducing the number of network layers.

In this article, we integrate sparse group-adaptive convolution into the Ghost Module and propose a novel lightweight module named Sparse Ghost Module. Using such a lightweight module to replace the standard convolution in the object detection networks can reduce the calculation and the number of model parameters, and improve the inference speed with almost no loss of prediction accuracy. Compared with the existing lightweight modules, our module not only reduces the complexity of the network, but also maintains high detection accuracy. In Section 5, more information about our lightweight module will be further detailed.

3 THE MULTIPLE KINDS OF UNDERWATER ORGANISMS DATASET

Our dataset was collected at the Xiaomeisha Ocean World in Shenzhen, Guangdong Province, China. Its aquarium has over twenty fish display tanks with different themes, breeding a variety of rare fish species with a quantity of over 10,000. The polar aquarium, jellyfish aquarium, shark aquarium, turtle island, and other pavilions showcase a wider range of underwater organisms. The climate of this area is mainly subtropical to tropical transitional maritime climate, with warm water suitable for the reproduction of marine life populations. The seawater and ecological environment of the aquarium simulate the marine environment, which ensures that the data we collect is as close to the real environment as possible. For the data collection of MKUO dataset, we used a handheld Nikon Z5 digital camera which is a 24.3 megapixel full frame digital camera with an internal processor of Speed 6. The camera body is equipped with five axis anti shake. The camera has 273 auto focal points and a sensitivity range of ISO 100-51,200. Our recording process lasted for 10 days, with two operators alternating day and night to record. After focusing on recording about eight categories of underwater organisms every day, a total of 84 species of underwater organisms were filmed, and 111 videos of 4.92 *GB* in total size were acquired. In collaboration with fisheries experts from the Eastern China Sea Fisheries Research Institute, we accurately labeled 111 video data. After multiple rounds of screening, 10,043 clear images were selected and cropped to the size of 684×456 to form the MKUO dataset.

3.1 Label Rules and Categories Information

The standardization degree of classification and the accuracy of labeling are important criteria for judging the quality of a dataset. Unfortunately, many factors in the marine underwater environment (lack of light in the deep sea, high chlorophyll concentration of seawater, high scattering turbidity concentration of seawater, and different absorption coefficients of different wavelengths of light at different depths of seawater, etc.) will affect the image quality, and thus the images of the seafloor collected in the real environment are usually blurry and have obvious chromatic aberration. Classifying underwater organisms from these images is not an easy task even for professional marine biologists [38].

Unlike most datasets that collect images in the ocean, our dataset is collected in a large aquarium. Due to the aquarium's sufficient ambient light, our dataset has clearer images than deep ocean ones, which helps to display organisms' features more clearly. Therefore, with the help of fishery experts, we easily obtained the common names of all underwater organisms in our dataset. Common names of organisms are easy to remember and record, and are often used as an alternative to scientific names. However, marine ecologists may sometimes find new underwater organisms without names, or encounter some underwater organisms with the same common name or with multiple common names. In these cases, scientific names are more suitable as markers of species. To standardize the classification, we used scientific name annotations in our dataset. The Binomial Nomenclature has been the standard form of the scientific name since Linnaeus [28] proposed it in 1753. As Binomial Nomenclature suggests, the name given to each species has two parts, the generic name (identifies the genus to which the species belongs) and the species name

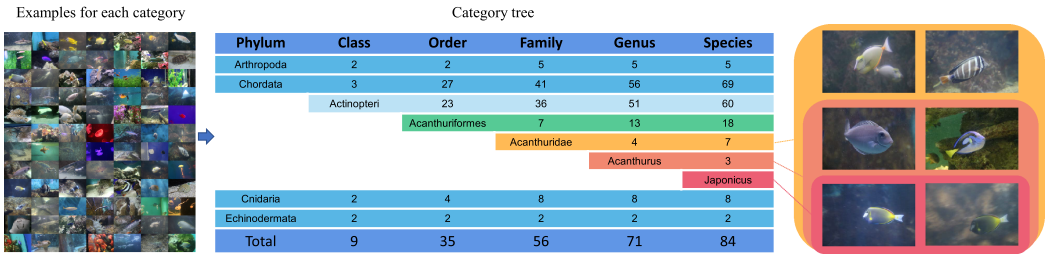


Fig. 3. The visualization shows the category tree of our proposed MKUO dataset and shows one example for each category.

(distinguishes the species within the genus). The generic name is formed by Latin grammaticalized nouns, but its word source can be from Latin words, Greek words or other Latinized words, and the initial letter of the name must be capitalized. The species name is an adjective in Latin, and its initial letter is not capitalized. For accurate annotations, we performed a manual comparison of images on *Baidu online encyclopedia*[11] and *Wikipedia*[12] based on common names to obtain organisms' scientific names. It should be noted that the scientific name provided in the online encyclopedia may be an unaccepted version of the ***WoRMS***[13] **(World Register of Marine Species)**. We also corrected the unaccepted scientific names to the accepted version in *WoRMS* which ensures that the final classification results are consistent with those names in *WoRMS*.

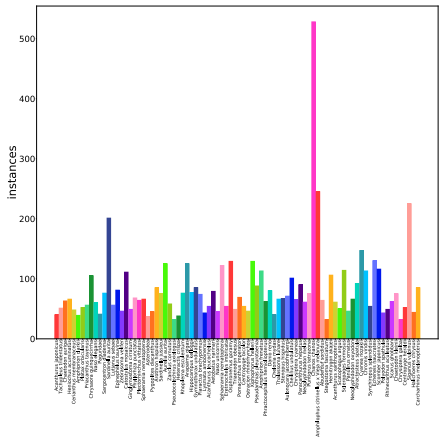
What's more, Figure 3 shows the category tree of our dataset, which is divided hierarchically by taxonomic phylum, class, orders, family, genus and species. The captured organisms include four phyla, nine classes, 35 orders, 56 families, 71 genera and 84 species. It should be noted that the category tree is not completely filled. For example, two species of turtles, *Eretmochelys imbricata* and *Chelonia mydas* have no class information on *WoRMS* and belong to the superclass *Reptilia*, which does not influence us to label them with their scientific names. The dataset contains one hybrid species, which we labeled as the scientific name combination of its parent species, called *Amphilophus citrinellus* × *Vieja melanurus*. In addition, we classify one kind of hermit crab into the family level *Paguridae*, and one kind of starfish into the class level *Asteroidea* without subclassifying because their common names are general and vague. Other categories are all subdivided into the species level. To the best of our knowledge, this dataset has the broadest organism species coverage among all annotated underwater organism datasets.

3.2 Dataset Overview

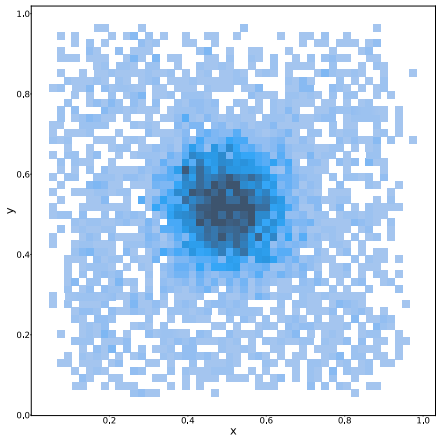
In total, our MKUO dataset contains 10,043 clear underwater organism images with bounding box form annotations. The resolution of each image is 684 × 456, and the number of objects captured in each image varies from one to dozens.

Figure 4 shows the visualization of some annotation information in the MKUO dataset, including the number of instances in each category, positions and size information of bounding boxes. The statistics show that except for the category *Chanos chanos*, which contains 1,084 labeled instances, the number of labeled instances in each category varies between 69 and 595. In summary, there are 14,476 annotated instances in our MKUO dataset. (c) of Figure 4 is the accumulated graph of the first 1,000 normalized bounding boxes, showing their shapes and distributions. More detailed statistics

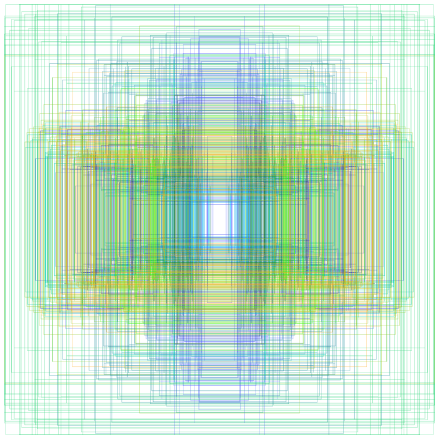
[11] *Baidu*: https://baike.baidu.com
[12] *Wikipedia*: https://en.wikipedia.org
[13] *WoRMS*: https://www.marinespecies.org/index.php



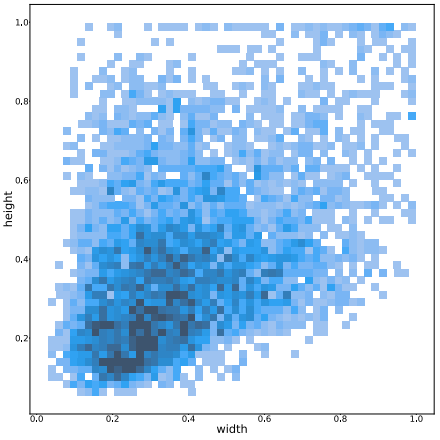
(a) Number of instances per category



(b) Bounding boxes' center location



(c) Accumulated normalized bounding boxes



(d) Bounding boxes' width and height

Fig. 4. The visualization of the annotation information in MKUO dataset on four typical aspects. (a) shows the total number of labeled instances in each category. (c) is the accumulated graph of the normalized bounding boxes (only the first 1,000 labels are visualized here to avoid excessive overlap of rectangles). (b) and (d) respectively show the locations and sizes of the normalized bounding boxes, which is displayed by the counting histogram divided into 50 cells on each dimension.

are shown in (b) and (d) of Figure 4. Figure 4(b) is the histogram of the joint distribution of the bounding boxes' locations. It can be seen that the centers of the bounding boxes were concentrated near the center of the image and spread over the whole image. Figure 4(d) is the histogram of the height and width of the bounding boxes showing that the dimensions of the bounding boxes cover a wide range of sizes. It can be seen that the height and width of most bounding boxes are less than half of the image.

To sum up, as shown in Table 1, our proposed MKUO dataset has the following three advantages compared with existing underwater organism image datasets:

(1) ***Wide coverage range of species.*** Unlike most existing underwater organism datasets, which only contain one or several kinds of underwater organisms, our MKUO dataset contains

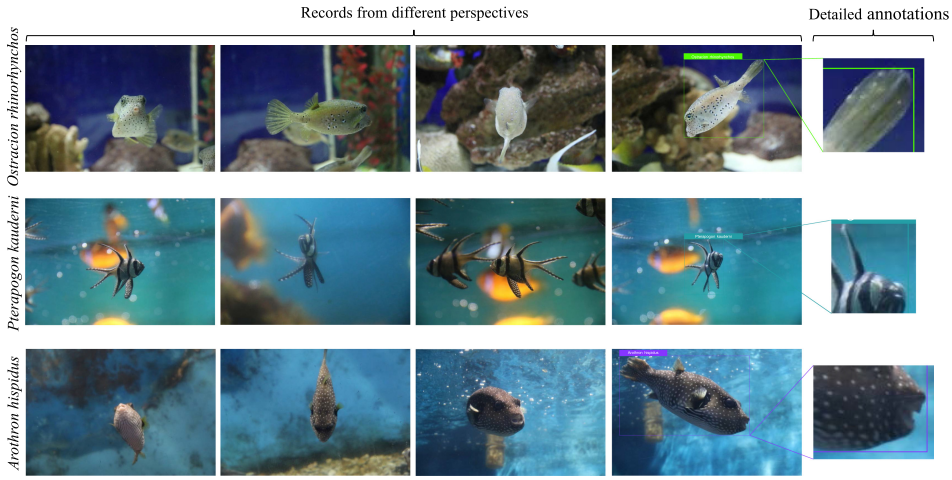


Fig. 5. Some examples in the MKUO dataset. From top to bottom are *Ostracion rhinorhynchos*, *Pterapogon kauderni*, and *Arothron hispidus*, respectively. The recorded data from different perspectives and detailed annotations are displayed from left to right.

84 species of underwater organisms, including *fish, jellyfish, hermit crab, lobster, turtle, limulus*, and so on. (as shown in Table 1). As the underwater organism dataset with the broadest species coverage to our knowledge (as shown in Figure 3), it is conducive to distinguishing different species of underwater organisms in the follow-up scientific research work.

(2) ***Precise species classification and location information.*** With the help of fishery experts from the Eastern China Sea Fisheries Research Institute, our dataset has accurate annotations which classify the instances precisely to species level (according to biology taxonomy). Each image in the dataset has been filtered multiple times and manually annotated, providing precise annotation at the pixel level in the form of bounding boxes (as shown in Figure 5), which is helpful for the future research of object detection algorithms.

(3) ***Complete morphological characteristics.*** With the influence of recording time, perspective and other factors, underwater organisms often show different morphological characteristics (as shown in (d) and (h) of Figure 1, only when photographing from the back, *Echeneis naucrates* shows a unique sucker like the dorsal fin, while from other perspectives it has no obvious features). Different from the fixed recording position of most datasets at present, our MKUO dataset recorded underwater organisms from different perspectives (as shown in Figure 5). This is helpful for the algorithm to learn the organisms' high-dimensional characteristics, which can not only be used to distinguish different organisms but also help to detect organisms of the same species under different postures.

3.3 Discussion on Application in Real Underwater Environment

Clear images were collected in the aquarium to establish the MKUO dataset. Constructing a dataset in this way is conducive to showcasing biodiversity and fully capturing the biological characteristics of underwater organisms, which are important to the underwater object detection task but usually difficult to be obtained in real underwater environments. In fact, even collecting data from the real environment, the lighting and hydrogeological conditions still vary obviously in different regions, naturally causing the differences in observations even though the observed organisms are the same. Thus, in the authors' opinion, collecting data from a real underwater environment

is not a suitable solution to solve the generalization problems of training models since it's almost impossible to collect real data covering all existing water environments.

Although there is a gap between our MKUO dataset and real underwater ocean data, and the models trained on the MKUO dataset may not yet have the ability to be directly applied in real ocean scenarios, the gap can be filled in via the underwater environment simulation to some extent, and such a technology is relatively mature currently. Before conducting object detection in real underwater scenes, it is often suggested to analyze, classify (Jerlov water types for example), and simulate the scenes based on different optical water types. After that, researchers can easily complete the mutual conversion of clear images from the MKUO dataset and simulated underwater images, thereby completing high-precision underwater object detection tasks. This is just in line with our research path for underwater organism monitoring tasks and we will continue to strive in this field in the future.

4 EVALUATION OF OBJECT DETECTION NETWORKS ON THE MKUO DATASET

As mentioned in Section 2.2, in many related works, researchers selected classic networks, and then further fine-tuned them to apply to underwater organism detection tasks. The object detection networks they selected, though, are quite different from the recently proposed algorithms which have made many unprecedented explorations or expansions, making them unique in terms of universality, time complexity, precision performance and other research concerns. It is of great significance to apply these algorithms to underwater organism detection. However, due to the lack of comprehensive comparison and evaluation of these recently proposed algorithms in underwater datasets, it is often difficult for researchers to choose appropriate networks as solutions according to their research needs. In order to fill in the gap in this regard, we compared and evaluated the performance of these classic networks and some recently proposed networks on our MKUO dataset, and analyzed their potential in underwater organism detection. In this section, we evaluated Faster-RCNN [40], SSD [29], RetinaNet [26], YOLOv3 [39] and 13 other advanced object detectors proposed since 2020 on the MKUO dataset to obtain baseline results for future reference. In order to evaluate as comprehensively as possible, the object detectors we selected have different design principles and architectures. In terms of whether anchor boxes are required, these networks include anchor-based networks, such as PAA [21] and YOLOv5 [45], and anchor free networks, such as FoveaBox [22] and ATSS [55]. In terms of detection steps, there are two-stage networks such as Double-Head RCNN [50], Dynamic RCNN [52] and Sparse RCNN [44], and one-stage networks such as YOLOX [33] and TOOD [13].

For the sake of fairness, the training process for all networks did not use pre-trained weights, and all models were trained on our proposed MKUO dataset and then evaluated. The dataset is split randomly into 48% training, 32% validation and 20% testing data. That is to say, the training set, validation set and testing set contain 4,820 images, 3,214 images and 2,009 images, respectively. The workstation used for training consists of two GEFORCE RTX 2080 Ti GPUs with 11GB GPU memory. Each network is trained for 300 epochs using its original training regime, and the weights that perform best in the validation set during the training process were taken as the evaluation target.

4.1 Evaluation Metrics

Our evaluations are mainly conducted on two aspects, the accuracy and the complexity. Next, we will introduce the metrics we utilized in detail.

(1) ***Accuracy metrics: Average Precision (AP) and mean Average Precision (mAP).***
In the evaluation metrics of Pascal VOC [12], the **average precision (AP)** of the detection is defined as the magnitude of the area under the precision/recall curve, with the recall on

the horizontal axis and the precision on the vertical axis, which is due to the fact that a good object detection algorithm needs to have good precision at different recall levels. Before VOC2010, the AP is defined as the mean precision at a set of eleven equally spaced recall levels $[0, 0.1, \ldots, 1]$:

$$AP = \frac{1}{11} \sum_{r \in \{0, 0.1, \ldots 0.9, 1.0\}} p_{\text{interp}}(r). \tag{1}$$

The precision at each recall level r is interpolated by taking the maximum precision for which the corresponding recall exceeds r:

$$p_{\text{interp}}(r) = \max_{\tilde{r}:\tilde{r} \geq r} p(\tilde{r}) \tag{2}$$

where $p(\tilde{r})$ is the measured precision at recall $\tilde{r}$. To improve precision and ability to measure differences between methods with low AP, from VOC2010 onwards the method of computing AP changed to use all data points rather than sampled at a fixed set of uniformly-spaced recall values. It is computed as in the following steps:

(a) Compute a version of the measured precision/recall curve with precision monotonically decreasing, by setting the precision for recall r to the maximum precision obtained for any recall $r' \geq r$.

(b) Compute the AP as the area under this curve by numerical integration.

Another frontier dataset in the research field of object detection, MS COCO [27], evaluated the AP at different **Intersections over Unions (IoUs)** $[0.5 : 0.05 : 0.95]$ for a total of 10 IoUs and averaged the AP at these thresholds as the result at the end, denoted as $\text{mAP}_{0.5:0.95}$. While Pascal VOC [12] only evaluates the AP value of those results whose IOU is at the threshold of 0.5. Compared to Pascal VOC [12], the evaluation matrix in the MS COCO [27] dataset is more comprehensive and widely used by recent object detection research, since not only the classification ability of the object detection model is evaluated, but also the localization ability of the detection model is reflected. Both $\text{AP}_{0.5}$ and $\text{mAP}_{0.5:0.95}$ were used as the evaluation metrics in this article, and $\text{mAP}_{0.5:0.95}$ took the first place of evaluation metrics on accuracy in subsequent comparison.

(2) ***Complexity metrics: the number of Parameters (Params), the FLoating point OPerations (FLOPs) and the inference Time (Time_{infer}).***

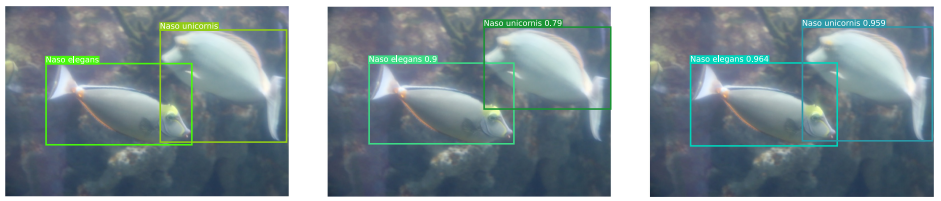
The complexities of compared models are mainly manifested on two aspects, space complexities and time complexities. On the aspect of space complexities, we utilized Params and FLOPs as corresponding metrics, while for time complexities, the Time_{infer} was adopted. It's worth mentioning that the Time_{infer} here includes the pre-processing time, algorithm operation time and post-processing time. Pre-processing includes loading images to GPU and resizing image processing (resize, pad, etc.), algorithm operation refers to the time that the network runs in the GPU and post-processing mainly includes **Non-Maximum Suppression (NMS)** and sending the results back to the CPU. For each model to be evaluated, we used a single GPU to test multiple groups of images, and finally calculated the average inference time of the model on one single image.

4.2 Precision and Complexity Results

As shown in Table 2, we list the evaluation results of each object detector using the aforementioned five accuracy or complexity metrics. Through the statistical analysis, in terms of the accuracy, the $\text{mAP}_{0.5:0.95}$ values of most evaluated networks fluctuate around 70%, and the $\text{AP}_{0.5}$ values fluctuate around 90%. In terms of the complexity, the minimum number of parameters of these models is as low as 31.98 M (TOOD), and the maximum number is more than 100 M (Sparse RCNN). The FLOPs

Table 2. Results of Compared Models Evaluated on the Proposed MKUO Dataset

Model	$mAP_{0.5:0.95}$(%)	$AP_{0.5}$(%)	Params(M)	FLOPs(G)	$Time_{infer}$(ms)
FoveaBox [22]	73.5	92.1	36.20	68.16	25.8
Double-Head RCNN [50]	73.7	92.9	47.14	352.14	93.5
ATSS [55]	75.7	92.9	32.08	67.83	25.5
SABL [49]	76.7	93.6	41.99	137.20	37.5
Dynamic RCNN [52]	73.2	92.9	41.55	77.49	28.3
PAA [21]	76.0	93.2	32.08	67.83	82.6
YOLOv5 [45]	79.9	96.5	46.60	109.20	5.1
VFNet [54]	60.9	77.9	32.68	63.72	30.1
Sparse RCNN [44]	74.1	90.8	106.07	54.69	33.6
YOLOX [33]	81.2	96.0	54.21	128.43	13.1
DINO [53]	78.5	93.9	47.71	95.30	59.5
DDQ [56]	78.9	93.4	44.92	71.69	62.9
TOOD [13]	68.6	85.5	31.98	60.93	29.5
YOLOv3 [39]	50.1	78.4	61.97	64.56	27.6
Faster-RCNN [40]	69.6	93.5	41.55	77.94	36.5
SSD [29]	70.0	92.8	34.84	128.16	17.6
RetinaNet [26]	76.7	93.3	37.83	79.59	35.4



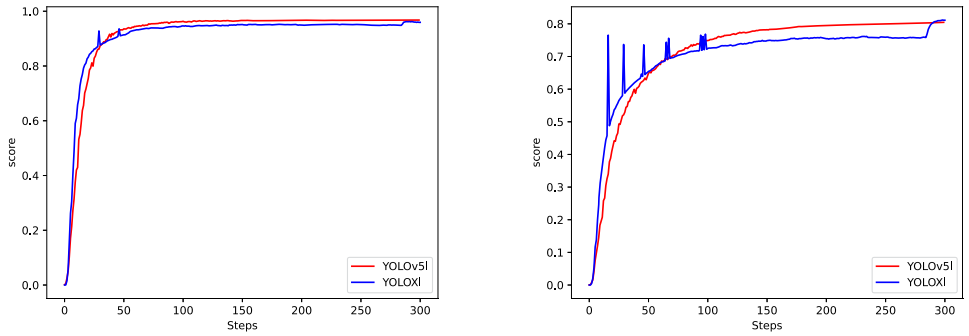
(a) Ground truth (b) Results from YOLOv5l (c) Results from YOLOXl

Fig. 6. Typical examples of detection results of YOLOv5l and YOLOXl. (a) is the labeled image of a *Naso elegans* and a *Naso unicornis*, (b) is the detection results of YOLOv5l, and (c) is the detection results of YOLOXl.

of these models vary from 54.69 *G* to 352.14 *G*, and the inference times vary from 5.1 milliseconds to 93.5 milliseconds.

Among these evaluated networks, YOLOX and YOLOv5 achieve the highest accuracies on $mAP_{0.5:0.95}$ and $AP_{0.5}$, respectively, with excellent inference speeds. It is worth noting that both YOLOv5 and YOLOX include a series of network models with similar structures, and can be classified into "small", "medium", and "large" according to different depths and widths. The network models we choose in our experiments are "large" models, YOLOv5l and YOLOXl. Next, more evaluation results about these two models will be given.

Figure 6 shows a typical sample of the results inferred by YOLOv5l and YOLOXl on the testing set and the corresponding manually labeled ground truth, and the image of the sample includes a *Naso elegans* and a *Naso unicornis*. It can be seen from the figure that YOLOX is slightly better than YOLOv5 in edge positioning and classification confidence of the bounding boxes, which is consistent with the $mAP_{0.5:0.95}$ results obtained from the quantitative evaluation results in Table 2. Besides, the accuracy scores along with the evolvement of training of these two compared models are shown in Figure 7. This figure shows that YOLOv5 converges more smoothly than YOLOX, which may be due to YOLOX's use of the decoupling head. It is also worth mentioning for the last



(a) The $AP_{0.5}$ curves of compared models

(b) The $mAP_{0.5:0.95}$ curves of compared models

Fig. 7. The accuracy scores along with the evolvement of training of two compared models on our MKUO dataset. (a) shows the curves of metric $AP_{0.5}$ and (b) corresponds to metric $mAP_{0.5:0.95}$.

Table 3. The Species on Which Compared Schemes Perform Best, Second Best and Worst

Model	Best species	Second best species	Worst species
FoveaBox [22]	*Asteroidea*	*Centropyge heraldi*	*Sardinella aurita*
Double-Head RCNN [50]	*Asteroidea*	*Heteractis crispa*	*Polyodon spathula*
ATSS [55]	*Asteroidea*	*Centropyge heraldi*	*Polyodon spathula*
SABL [49]	*Asteroidea*	*Heteractis crispa*	*Polyodon spathula*
Dynamic RCNN [52]	*Asteroidea*	*Centropyge heraldi*	*Sardinella aurita*
PAA [21]	*Asteroidea*	*Centropyge heraldi*	*Sardinella aurita*
YOLOv5 [45]	*Tachypleus tridentatus*	*Zebrasoma velifer*	*Sardinella aurita*
VFNet [54]	*Centropyge heraldi*	*Asteroidea*	*Polyodon spathula*
Sparse RCNN [44]	*Asteroidea*	*Heteractis crispa*	*Sardinella aurita*
YOLOX [33]	*Asteroidea*	*Centropyge heraldi*	*Sardinella aurita*
TOOD [13]	*Heteractis crispa*	*Centropyge heraldi*	*Polyodon spathula*
DINO [53]	*Asteroidea*	*Centropyge heraldi*	*Sardinella aurita*
DDQ [56]	*Centropyge heraldi*	*Asteroidea*	*Sardinella aurita*
YOLOv3 [39]	*Asteroidea*	*Sphaeramia nematoptera*	*Aetobatus narinari*
Faster-RCNN [40]	*Asteroidea*	*Centropyge heraldi*	*Sardinella aurita*
SSD [29]	*Heteractis crispa*	*Centropyge heraldi*	*Polyodon spathula*
RetinaNet [26]	*Asteroidea*	*Centropyge heraldi*	*Polyodon spathula*

15 epochs before the end of the training of YOLOX, the Data Augmentation (Mosaic and Mixup) is turned off, which is very helpful for accuracy improving, since the augmented training images are far from the true distribution of natural images. Such an operation brings a significant improvement to YOLOX in $mAP_{0.5:0.95}$ and helps YOLOX catch up with and surpass the performance of YOLOv5.

4.3 Performances of Models on Various Species

The species on which compared schemes perform best, second best and worst are summarized in Table 3. It can be seen that among 84 classes of organisms, most object detectors performed best in *Asteroidea*, which is a kind of starfish and has a unique shape. Besides, two other kinds of underwater organisms, *Heteractis crispa* and *Centropyge heraldi*, are also easily detected by most object detectors. The former is a kind of anemone with purple on the top of its tentacles, and the

latter is a kind of ornamental fish with bright yellow color. The characteristics of these three kinds of organisms are unique and obvious, thus it is easy to distinguish them from other species. From the results, the species with low detection accuracy of these algorithms mainly include *Polygon spathula* and *Sardinella aurita*. The former, also known as American paddlefish, is a kind of large-size sturgeon, and the latter is a kind of small sardine. According to our analysis, on the one hand, the low detection accuracy of *Polygon spathula*, which is **Vulnerable (VU)** determined by **IUCN (International Union for Conservation of Nature)** [14], is due to that the number of instances captured in our dataset is relatively less. On the other hand, the low detection accuracy of *Sardinella aurita* is due to its lack of unique biological characteristics, and it is still a challenging study to distinguish different sardines.

5 SPARSE GHOST MODULE: A NOVEL LIGHTWEIGHT CONVOLUTION MODULE

Aiming at the problems of high computing complexity and slow inference speed in the application of existing object detection algorithms based on deep learning in underwater organism observation, we propose a novel lightweight module, namely Sparse Ghost Module. The module could complete standard convolution with much fewer parameters and floating point operations, helping to reduce the storage space and calculation required by the object detection networks, so as to adapt to the underwater hardware platform.

The Sparse Ghost Module focuses on eliminating the data redundancy of the traditional convolutional kernel in different channels. In well-trained deep neural networks constructed using traditional convolution methods, the rich feature maps can ensure a comprehensive understanding of input data, but they usually bring significant storage burden. Specifically, traditional deep convolutional neural networks usually generate many similar feature-map pairs in the same layer, called ghost maps [15]. Unlike the implementations of tensor products used in traditional dense convolutions, in order to generate these feature maps more cheaply, our Sparse Ghost Module uses matrix decomposition and low-rank approximation to generate them, effectively reducing computational complexity.

5.1 Details of Sparse Ghost Module

Different from the Ghost Module [15] (which is described in detail in Section 2.3), our proposed Sparse Ghost Module is composed of two parts. In the first part, a strictly controlled number of ordinary convolution kernels is used to generate intrinsic feature maps, and in the second part, the group-adaptive convolutions are used to generate more feature maps. Figure 8 shows the illustration of the Ghost Module and the Sparse Ghost Module.

Compared to Ghost Module [15] which uses group convolution to generate feature maps, we apply low-rank approximations in the second part of Sparse Ghost Module according to the idea of matrix decomposition. To concisely describe the idea, we assume a $k \times k$ convolution kernel $\boldsymbol{W}$ with the same number of input and output channels ($c_{\text{in}} = c_{\text{out}} = c$) and ignore bias terms. The kernel matrix $\boldsymbol{W}$ could be sparsely decomposed as,

$$\boldsymbol{W} = \boldsymbol{P}\Phi\boldsymbol{Q}^T \tag{3}$$

where $\boldsymbol{W}$ is a $c \times c$ matrix, $\boldsymbol{Q}$ is a $c \times \frac{c}{r}$ matrix that compresses the number of channels by the channel reduction ratio r, and $\boldsymbol{P}$ is a $c \times \frac{c}{r}$ matrix that expands the number of channels back to c. Furthermore, the corresponding convolution is sparsely decomposed into two group-adaptive

[14] *IUCN*: the international organization working in the field of nature conservation which has the most comprehensive collection of authoritative publications, reports, guides and databases supporting the fields of conservation and sustainable development.

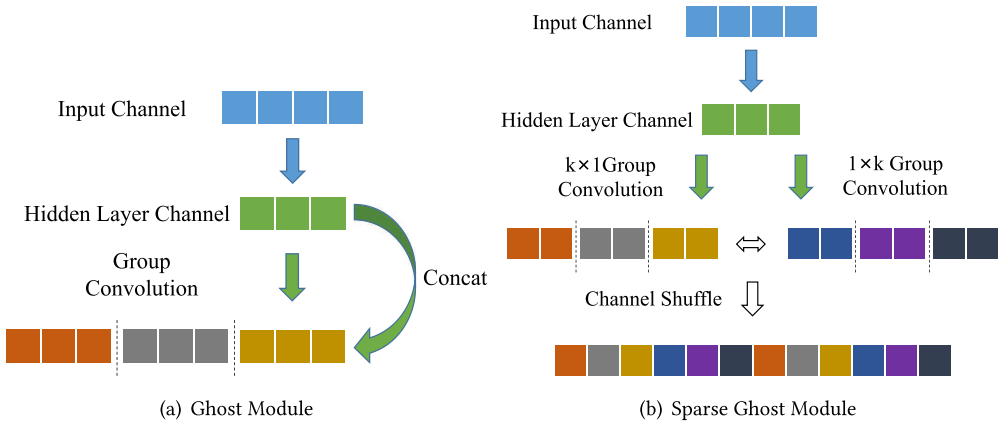


Fig. 8. Illustration of the Ghost Module and the Sparse Ghost Module whose $c_{\text{in}} = 4$, $c_{\text{out}} = 12$ and $s = 4$. (a) is the Ghost Module and (b) is the Sparse Ghost Module.

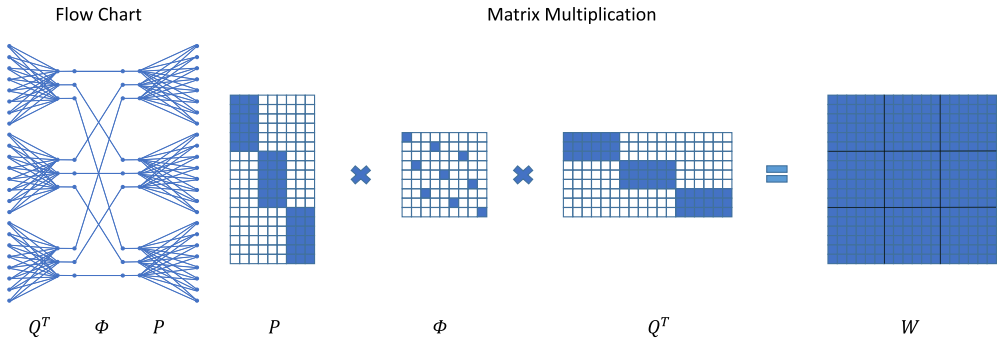


Fig. 9. Sparsely decomposing a pointwise convolution into two group-adaptive convolutions, where $c = 18, r = 2$ and $g = 3$. The matrix $\boldsymbol{W}$ can be divided into $g \times g$ blocks, and the rank of each block is 1.

convolutions, where the number of groups is g. That is, $\boldsymbol{P}$ and $\boldsymbol{Q}$ are diagonal block matrices with g blocks, and each block corresponds to the convolution of a group of channels. Φ is a $\frac{c}{r} \times \frac{c}{r}$ permutation matrix for channels shuffling. In the most special case, to sparsely decompose a pointwise convolution, as MicroNet [24] has done, the group-adaptive convolutions' kernel size is 1×1, and the computational complexity is $O = \frac{2c^2}{rg}$, as shown in Figure 9.

As mentioned above, in order to sparsely decompose the convolution in the second part of the Sparse Ghost Module, the original convolution should be replaced by two linked group-adaptive convolutions with the kernel size of $k \times k$, while some adjustments are made to build the Sparse Ghost Module. Firstly, two group-adaptive convolutions are used to process the same feature map from the hidden layer. The kernel sizes of these two convolutions are $k \times 1$ and $1 \times k$, respectively. Secondly, the two output feature maps are directly concatenated and a channel shuffle is applied after the concatenation, as shown in Figure 8. This adjustment is reasonable for the following two reasons: (1) Two strip convolution kernels could be used to approximate the square kernel when the kernel size k is relatively small; (2) Since the object detection network is usually deep, channel shuffle could ensure the channel connectivity between the group-adaptive convolutions of the adjacent Sparse Ghost Modules. This adjustment would also bring two benefits: (1) The number

Table 4. Experimental Results on Lightweight YOLOv5l and YOLOXl

Model	$mAP_{0.5:0.95}$ (%)	$mAP_{0.5}$ (%)	Params (M)	$Size_{weights}$ (MB)	FLOPs (G)	$Time_{inference}$ (ms)
$YOLOv5l_{(base)}$	79.9	96.5	46.6	93.8	109.2	5.1
YOLOv5l-Depthwise Separable [17]	50.8	87.2	14.0	28.8	30.5	4.0
YOLOv5l-Fire [30]	77.6	96.1	17.4	35.9	38.9	4.1
YOLOv5l-Ghost [15]	77.5	96.2	12.6	26.0	27.3	6.0
YOLOv5l-Sparse Ghost	78.2	96.2	12.4	25.8	26.5	4.4
$YOLOXl_{(base)}$	81.2	96	54.21	434.4	128.43	13.11
YOLOXl-Depthwise Separable [17]	58.9	92.6	14.94	120.7	29.83	8.06
YOLOXl-Fire [30]	78.1	95.9	18.69	150.8	39.08	8.34
YOLOXl-Ghost [15]	77.8	96.2	14.57	117.6	31.62	9.01
YOLOXl-Sparse Ghost	78.3	95.9	13.94	112.8	29.65	8.88

of parameters of the strip convolution kernel is less than that of the square convolution kernel; (2) Compared with the structure before adjustment, the number of channels to be processed by each group-adaptive convolution is reduced by half, and then the computational complexity is naturally lower.

The proposed Sparse Ghost Module could be easily integrated into the existing well-designed object detection networks to reduce their complexities. In order to analyze the profits of using the Sparse Ghost Module on memory consumption and inference speed, we calculated the theoretical compression ratio of the Sparse Ghost Module. Before that, for the convenience of calculation, the theoretical compression ratio of the Ghost Module should be calculated first. Assuming that the convolution kernel size of standard convolution is $k \times k$, the input channel size is c_{in}, and the output channel size is c_{out}. When the number of hidden channels is set to $\frac{c_{\text{out}}}{s}$ and the kernel size of the group convolution is set to k, the parameter compression ratio cr_{Ghost} of the Ghost Module [15] can be calculated as,

$$cr_{\text{Ghost}} = \frac{c_{\text{out}} \cdot c_{\text{in}} \cdot k \cdot k}{\frac{c_{\text{out}}}{s} \cdot c_{\text{in}} \cdot k \cdot k + (s-1) \cdot \frac{c_{\text{out}}}{s} \cdot k \cdot k} = \frac{s \cdot c_{\text{in}}}{c_{\text{in}} + s - 1} \approx s \tag{4}$$

where $s \ll c_{\text{in}}$. In comparison, the parameter compression ratio $cr_{\text{SparseGhost}}$ of the Sparse Ghost Module can be calculated as,

$$cr_{\text{SparseGhost}} = \frac{c_{\text{out}} \cdot c_{\text{in}} \cdot k \cdot k}{\frac{c_{\text{out}}}{s} \cdot c_{\text{in}} \cdot k \cdot k + 2 \cdot \frac{s}{2} \cdot \frac{c_{\text{out}}}{s} \cdot k \cdot 1} = \frac{s \cdot c_{\text{in}} \cdot k}{c_{\text{in}} \cdot k + s} \geq \frac{s \cdot c_{\text{in}}}{c_{\text{in}} + s - 1} \approx s. \tag{5}$$

This inequality only takes the equal sign when $s = k = 2$, where $s, k \in \{q | q > 1, q \in \mathbb{N}\}$, and generally s and k are no less than 3. Thus, in most cases, the compression ratio of the Sparse Ghost Module obviously exceeds that of the ordinary Ghost Module.

5.2 Experiments on Lightweight Object Detection Network

In order to verify the practical effectiveness of the Sparse Ghost Module, a series of experiments were conducted. YOLOv5 and YOLOX were selected as experimental subjects since these two object detection networks performed relatively well in the evaluation of underwater organism detection (refers to Section 4.2). By replacing the standard convolutions in these two networks with different classic lightweight modules (including Depthwise Separable Modules [17], Fire Modules [30] and Ghost Modules [15]) and Sparse Ghost Modules, we obtained a series of experimental results about the lightweight effects of these modules. All experiments were conducted on our proposed MKUO dataset, and the results were obtained on the testing set after each model was trained on the training set (refer to Section 4 for the division of the dataset). As shown in Table 4, compared with the existing classic lightweight modules, the Sparse Ghost Module significantly reduces the computation and the number of parameters at the minimum accuracy loss. Besides,

whether for YOLOv5l or YOLOXl, among the results of various lightweight networks, the network using the Sparse Ghost Modules was the best in terms of precision ($mAP_{0.5:0.95}$ and $mAP_{0.5}$) and network complexity (the number of network parameters, parameter file size and the floating point operations). Although the lightweight network using Depthwise Separated Module [17] has a slight advantage in inference speed, the corresponding detection accuracy has decreased significantly. Considering all metrics comprehensively, the lightweight effect of the Sparse Ghost Module is the best among all compared schemes in our experiments, which means that it is more suitable for application in an underwater environment with limited hardware conditions.

6 CONCLUSION

In this article, a novel large-scale annotated underwater organism dataset named MKUO was established, which consists of 10,043 images and 14,476 annotations. The dataset contains 84 species of underwater organisms (including *fish, jellyfish, hermit crab, lobster, turtle, limulus, sea anemone, seahorse, etc.*), and it has the broadest organism species coverage among all existing annotated underwater organism datasets to the best of our knowledge. Based on our MKUO dataset, a comprehensive evaluation of the existing underwater organism detection algorithms and typical universal object detection algorithms was conducted. The evaluation results on both accuracy metrics and complexity ones were provided for the reference of future research. Our work is helpful for subsequent researchers to select the proper object detection algorithm for underwater organism detection tasks according to their own needs. Besides, in consideration of the hardware limitations of underwater devices, a novel lightweight convolution module named Sparse Ghost Module was proposed. By substituting the standard convolution in the object detection networks with our Sparse Ghost Module, the calculation and the number of parameters of the networks can be significantly reduced and the inference speed can naturally be improved without noticeable accuracy loss.

REFERENCES

[1] K. Anantharajah, Z. Ge, C. McCool, S. Denman, C. Fookes, P. Corke, D. Tjondronegoro, and S. Sridharan. 2014. Local inter-session variability modelling for object classification. In *IEEE Winter Conference on Applications of Computer Vision*. 309–316.

[2] A. R. Appenzeller and W. C. Leggett. 1992. Bias in hydroacoustic estimates of fish abundance due to acoustic shadowing: Evidence from day–night surveys of vertically migrating fish. *Canadian Journal of Fisheries and Aquatic Sciences* 49, 10 (1992), 2179–2189.

[3] O. Beijbom, P. J. Edmunds, D. I. Kline, B. G. Mitchell, and D. Kriegman. 2012. Automated annotation of coral reef survey images. In *IEEE Conference on Computer Vision and Pattern Recognition*. 1170–1177.

[4] O. Beijbom, T. Treibitz, D. I. Kline, G. Eyal, A. Khen, B. Neal, Y. Loya, B. G. Mitchell, and D. Kriegman. 2016. Improving automated annotation of benthic survey images using wide-band fluorescence. *Scientific Reports* 6, 1 (2016), 1–11.

[5] B. J. Boom, P. X. Huang, J. He, and R. B. Fisher. 2012. Supporting ground-truth annotation of image datasets using clustering. In *International Conference on Pattern Recognition*. 1542–1545.

[6] K. Cai, X. Miao, W. Wang, H. Pang, Y. Liu, and J. Song. 2020. A modified YOLOv3 model for fish detection based on MobileNetv1 as backbone. *Aquacultural Engineering* 91 (2020), 102117:1–9.

[7] Northeast Fisheries Science Center. 2022. Habitat mapping camera (HABCAM). https://habcam.whoi.edu/data-and-visualization/

[8] F. Chollet. 2017. Xception: Deep learning with depthwise separable convolutions. In *IEEE Conference on Computer Vision and Pattern Recognition*. 1800–1807.

[9] G. Cutter, K. Stierhoff, and J. Zeng. 2015. Automated detection of rockfish in unconstrained underwater videos using Haar cascades and a new image dataset: Labeled Fishes in the Wild. In *IEEE Winter Applications and Computer Vision Workshops*. 57–62.

[10] M. Dawkins, C. Stewart, S. Gallager, and A. York. 2013. Automatic scallop detection in benthic environments. In *IEEE Workshop on Applications of Computer Vision*. 160–167.

[11] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and F. Li. 2009. ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*. 248–255.

[12] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. 2010. The Pascal Visual Object Classes (VOC) challenge. *International Journal of Computer Vision* 88, 2 (2010), 303–338.
[13] C. Feng, Y. Zhong, Y. Gao, M. R. Scott, and W. Huang. 2021. TOOD: Task-aligned one-stage object detection. In *IEEE International Conference on Computer Vision*. 3490–3499.
[14] Australian Centre for Field Robotics. 2022. Tasmania Coral Point Count. http://marine.acfr.usyd.edu.au/datasets/
[15] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu. 2020. GhostNet: More features from cheap operations. In *IEEE Conference on Computer Vision and Pattern Recognition*. 1577–1586.
[16] K. He, X. Zhang, S. Ren, and J. Sun. 2016. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*. 770–778.
[17] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. 2017. MobileNets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861* (2017).
[18] A. Joly, H. Goëau, H. Glotin, C. Spampinato, P. Bonnet, W. Vellinga, R. Planqué, A. Rauber, S. Palazzo, B. Fisher, and H. Müller. 2014. LifeCLEF 2014: Multimedia life species identification challenges. In *International Conference of the Cross-Language Evaluation Forum for European Languages*. 229–249.
[19] A. Joly, H. Goëau, H. Glotin, C. Spampinato, P. Bonnet, W. Vellinga, R. Planqué, A. Rauber, S. Palazzo, B. Fisher, and H. Müller. 2015. LifeCLEF 2015: Multimedia life species identification challenges. In *International Conference of the Cross-Language Evaluation Forum for European Languages*. 462–483.
[20] J. JÃďger, M. Simon, J. Denzler, V. Wolff, K. Fricke-Neuderth, and C. Kruschel. 2015. Croatian fish dataset: Fine-grained classification of fish species in their natural habitat. In *British Machine Vision Conference Workshops*. 6.1–6.7.
[21] K. Kim and H. S. Lee. 2020. Probabilistic anchor assignment with IoU prediction for object detection. In *European Conference on Computer Vision*. 355–371.
[22] T. Kong, F. Sun, H. Liu, Y. Jiang, L. Li, and J. Shi. 2020. FoveaBox: Beyond anchor-based object detection. *IEEE Transactions on Image Processing* 29 (2020), 7389–7398.
[23] A. Krizhevsky, I. Sutskever, and G. E. Hinton. 2017. ImageNet classification with deep convolutional neural networks. *Commun. ACM* 60, 6 (2017), 84âĂŞ90.
[24] Y. Li, Y. Chen, X. Dai, D. Chen, M. Liu, L. Yuan, Z. Liu, L. Zhang, and N. Vasconcelos. 2021. MicroNet: Improving image recognition with extremely low FLOPs. In *IEEE International Conference on Computer Vision*. 458–467.
[25] J. Lin, W. Chen, Y. Lin, J. Cohn, C. Gan, and S. Han. 2020. MCUNet: Tiny deep learning on IoT devices. In *Advances in Neural Information Processing Systems*. 11711–11722.
[26] T. Lin, P. Goyal, R. Girshick, K. He, and P. DollÃąr. 2020. Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42, 2 (2020), 318–327.
[27] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. DollÃąr, and C. L. Zitnick. 2014. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision*. 740–755.
[28] C. V. Linnaeus. 1753. *Species Plantarum: Exhibentes Plantas Rite Cognitas, Ad Genera Relatas, Cum Differentiis Specificis, Nominibus Trivialibus, Synonymis Selectis, Locis Natalibus, Secundum Systema Sexuale Digestas*. Vol. 1. Holmiae, Impensis Laurentii Salvii. 572 pages.
[29] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Fu, and A. C. Berg. 2016. SSD: Single shot multibox detector. In *European Conference on Computer Vision*. 21–37.
[30] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Fu, and A. C. Berg. 2017. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size. In *International Conference on Learning Representations*. 1–13.
[31] R. Lyu. 2021. NanoDet-Plus. https://github.com/RangiLyu/nanodet/releases/tag/v1.0.0-alpha-1/
[32] A. Mahmood, M. Bennamoun, S. An, F. Sohel, F. Boussaid, R. Hovey, G. Kendrick, and R. B. Fisher. 2016. Automatic annotation of coral reefs using deep learning. In *OCEANS 2016 MTS/IEEE Monterey*. 1–5.
[33] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun. 2021. YOLOX: Exceeding YOLO series in 2021. arXiv preprint arXiv: 2107.08430 (2021).
[34] O. A. Misund, A. Aglen, and E. FrÃŷnÃęs. 1995. Mapping the shape, size, and density of fish schools by echo integration and a high-resolution sonar. *ICES Journal of Marine Science* 52, 1 (1995), 11–20.
[35] National Oceanic and Atmospheric Administration. 2021. How much of the ocean have we explored? https://oceanservice.noaa.gov/facts/exploration.html
[36] OpenAI. 2020. GPT-3: Language Models are Few-Shot Learners. https://github.com/openai/gpt-3/
[37] K. Ovchinnikova, M. A. James, T. Mendo, M. Dawkins, J. Crall, and K. Boswarva. 2021. Exploring the potential to use low cost imaging and an open source convolutional neural network detector to support stock assessment of the king scallop (Pecten maximus). *Ecological Informatics* 62 (2021), 101233:1–10.
[38] M. Pedersen, J. Bruslund Haurum, R. Gade, and T. B. Moeslund. 2019. Detection of marine animals in a new underwater dataset with varying visibility. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 18–26.
[39] J. Redmon and A. Farhadi. 2018. YOLOv3: An incremental improvement. *arXiv preprint arXiv:1804.02767* (2018).
[40] S. Ren, K. He, R. Girshick, and J. Sun. 2017. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39, 6 (2017), 1137–1149.

[41] A. Salman, A. Jalal, F. Shafait, A. Mian, M. Shortis, J. Seager, and E. Harvey. 2016. Fish species classification in unconstrained underwater environments based on deep learning. *Limnology and Oceanography: Methods* 14, 9 (2016), 570–585.

[42] S. A. Siddiqui, A. Salman, M. I. Malik, F. Shafait, A. Mian, M. R. Shortis, and E. S. Harvey. 2017. Automatic fish species classification in underwater videos: Exploiting pre-trained deep neural network models to compensate for limited labelled data. *ICES Journal of Marine Science* 75, 1 (2017), 374–389.

[43] L. Soukup. 2021. Automatic coral reef annotation, localization and pixel-wise parsing using mask R-CNN. In *Working Notes of CLEF*. 1359–1364.

[44] P. Sun, R. Zhang, Y. Jiang, T. Kong, C. Xu, W. Zhan, M. Tomizuka, L. Li, Z. Yuan, C. Wang, and P. Luo. 2021. Sparse R-CNN: End-to-end object detection with learnable proposals. In *IEEE Conference on Computer Vision and Pattern Recognition*. 14449–14458.

[45] Ultralytics. 2021. YOLOv5. https://github.com/ultralytics/yolov5/

[46] G. Van Horn, O. Mac Aodha, Y. Song, Y. Cui, C. Sun, A. Shepard, H. Adam, P. Perona, and S. Belongie. 2018. The iNaturalist species classification and detection dataset. In *IEEE Conference on Computer Vision and Pattern Recognition*. 8769–8778.

[47] S. Villon, M. Chaumont, G. Subsol, S. Villeger, T. Claverie, and D. Mouillot. 2016. Coral reef fish detection and recognition in underwater videos by supervised machine learning: Comparison between deep learning and HOG+SVM methods. In *Advanced Concepts for Intelligent Vision Systems*. 160–171.

[48] C. Wang, H. M. Liao, Y. Wu, P. Chen, J. Hsieh, and I. Yeh. 2020. CSPNet: A new backbone that can enhance learning capability of CNN. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 1571–1580.

[49] J. Wang, W. Zhang, Y. Cao, K. Chen, J. Pang, T. Gong, J. Shi, C. C. Loy, and D. Lin. 2020. Side-aware boundary localization for more precise object detection. In *European Conference on Computer Vision*. 403–419.

[50] Y. Wu, Y. Chen, L. Yuan, Z. Liu, L. Wang, H. Li, and Y. Fu. 2020. Rethinking classification and localization for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*. 10183–10192.

[51] N. Wulandari, I. Ardiyanto, and H. A. Nugroho. 2022. A comparison of deep learning approach for underwater object detection. *Journal RESTI (Rekayasa Sistem Dan Teknologi Informasi)* 6, 2 (2022), 252–258.

[52] H. Zhang, H. Chang, B. Ma, N. Wang, and X. Chen. 2020. Dynamic R-CNN: Towards high quality object detection via dynamic training. In *European Conference on Computer Vision*. 260–275.

[53] H. Zhang, F. Li, S. Liu, L. Zhang, H. Su, J. Zhu, L. M. Ni, and H. Shum. 2023. DINO: DETR with improved denoising anchor boxes for end-to-end object detection. In *International Conference on Learning Representations*. 1–19.

[54] H. Zhang, Y. Wang, F. Dayoub, and N. SÃijnderhauf. 2021. VarifocalNet: An IoU-aware dense object detector. In *IEEE Conference on Computer Vision and Pattern Recognition*. 8510–8519.

[55] S. Zhang, C. Chi, Y. Yao, Z. Lei, and S. Z. Li. 2020. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In *IEEE Conference on Computer Vision and Pattern Recognition*. 9756–9765.

[56] S. Zhang, X. Wang, J. Wang, J. Pang, C. Lyu, W. Zhang, P. Luo, and K. Chen. 2023. Dense distinct query for end-to-end object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*. 7329–7338.

[57] X. Zhang, X. Zhou, M. Lin, and J. Sun. 2018. ShuffleNet: An extremely efficient convolutional neural network for mobile devices. In *IEEE Conference on Computer Vision and Pattern Recognition*. 6848–6856.

[58] P. Zhuang, Y. Wang, and Y. Qiao. 2018. WildFish: A large benchmark for fish recognition in the wild. In *ACM International Conference on Multimedia*. 1301–1309.

[59] P. Zhuang, Y. Wang, and Y. Qiao. 2021. WildFish++: A comprehensive fish benchmark for multimedia research. *IEEE Transactions on Multimedia* 23 (2021), 3603–3617.

[60] J. Zwolinski, P. G. Fernandes, V. Marques, and Y. Stratoudakis. 2009. Estimating fish abundance from acoustic surveys: Calculating variance due to acoustic backscatter and length distribution error. *Canadian Journal of Fisheries and Aquatic Sciences* 66, 12 (2009), 2081–2095.

Received 23 February 2023; revised 19 November 2023; accepted 8 January 2024