SWIPENET: Object detection in noisy underwater scenes

Long Chen[a], Feixiang Zhou[a], Shengke Wang[b], Junyu Dong[b], Ning Li[c], Haiping Ma[d], Xin Wang[e], Huiyu Zhou[a,*]

[a]*School of Computing and Mathematical Sciences, University of Leicester, United Kingdom*
[b]*Department of information science and engineering, Ocean University of China, China*
[c]*College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, China*
[d]*Department of Electrical Engineering, Shaoxing University, China*
[e]*College of Computer and Information, Hohai University, China.*

Abstract

Deep learning based object detection methods have achieved promising performance in controlled environments. However, these methods lack sufficient capabilities to handle underwater object detection due to these challenges: (1) images in the underwater datasets and real applications are blurry whilst accompanying severe noise that confuses the detectors and (2) objects in real applications are usually small. In this paper, we propose a Sample-WeIghted hyPEr Network (SWIPENET), and a novel training paradigm named Curriculum Multi-Class Adaboost (CMA), to address these two problems at the same time. Firstly, the backbone of SWIPENET produces multiple high resolution and semantic-rich Hyper Feature Maps, which significantly improve small object detection. Secondly, inspired by the human education process that drives the learning from easy to hard concepts, we propose the noise-robust CMA training paradigm that learns the clean data first and then move on to learns the diverse noisy data. Experiments on four underwater object detection datasets show that the proposed SWIPENET+CMA framework achieves better or competitive accuracy in object detection against several state-of-the-art approaches.

*Corresponding author
Email address: hz143@leicester.ac.uk (Huiyu Zhou)
URL: https://www2.le.ac.uk/departments/informatics/people/huiyu-zhou (Huiyu Zhou)

Keywords: Underwater object detection, Curriculum Multi-Class Adaboost, sample-weighted detection loss, noisy data

1. Introduction

Autonomous underwater vehicles (AUVs) [1] and remotely operated vehicles (ROVs) [2] equipped with intelligent underwater object detection systems is of great significance for ocean resource exploitation and protection. Unfortunately, complicated underwater environments and lighting conditions introduce considerable noise into the captured images, which has posed massive challenges to intelligent vision-based object detection systems [3]. Therefore, it is crucial to develop novel underwater object detection techniques which effectively handle noise for the AUVs and ROVs applications.

Deep learning based object detection systems have demonstrated promising performance in various applications but still felt short of dealing with underwater object detection. This is because, firstly, underwater detection datasets are scarce and the objects in the available underwater datasets and real applications are usually small. Current deep learning based detectors cannot effectively detect small objects (see an example shown in Fig. 1). Secondly, the images in the existing underwater datasets and real applications accompany considerable noisy data. In the underwater scenes, wavelength-dependent absorption and scattering [3] cause serious visibility loss, contrast decrease and color distortion, generating considerable noisy data. The noisy data refer to the hard object instances, which are visually similar to the complex background in the blurry underwater images. They exaggerate the challenge of inter-class similarity, resulting in the confusion between the object classes and the background class. As shown on the bottom row of Fig. 1, the proposed SWIPENET trained on the noisy data cannot distinguish between the background and the objects.

In this paper, we propose a deep ensemble detector which is effective in dealing with small objects and noisy data in the underwater scenes. To achieve the objectives, we propose a deep backbone network named Sample-WeIghted

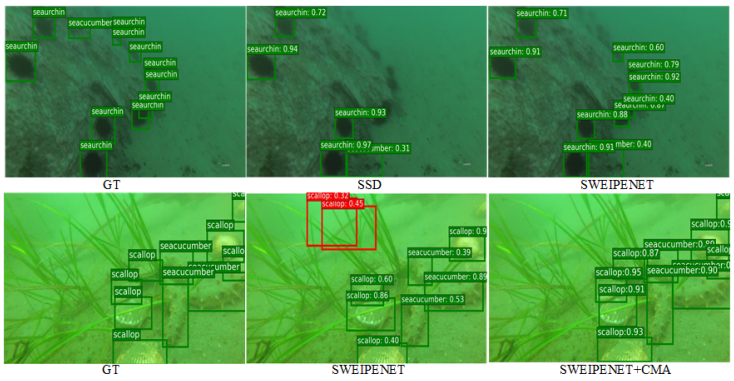

Figure 1: Exemplar images with ground truth (GT) annotations, results of Single Shot MultiBox Detector (SSD) [4], our proposed SWIPENET and SWIPENET+CMA. The top row shows that SSD cannot detect all the small objects while our proposed SWIPENET outperforms SSD in this case. The bottom row shows our proposed SWIPENET treats the background as objects due to the existence of noisy data while our proposed SWIPENET+CMA performs better than the others.

hyPEr Network (SWIPENET), which fully takes advantage of multiple Hyper Feature Maps. To address the noisy data problem, we propose a novel sample-weighted detection loss function and a novel noise-robust training paradigm named Curriculum Multi-Class Adaboost (CMA), used to train the deep ensemble for underwater object detection. Indeed, the sample-weighted detection loss is used to control the influence of the training samples on SWIPENET. It works with the training paradigm CMA to train the proposed deep ensemble detector to reduce errors.

The proposed CMA training paradigm is inspired by the idea in the human education system that starts from learning easy tasks, and then gradually increase learning difficulty levels. This learning concept has been utilised to improve the generalisation ability and accelerate convergence in machine learning algorithms. For example, Derenyi et al. [5] reported theoretical analysis where easy examples should be learnt first due to less noise. They treat the samples misclassified by the Bayesian classifier as noisy data and learn the easier samples first, then improve convergence and the generalisation ability. Mo-

tivated by these works, our CMA training paradigm consists of two training stages: Noise-eliminating (NECMA) and noise-learning (NLCMA) stages. In the noise-eliminating stage, a 'clean' detector (SWIPENET) of being free from the influence of noisy data is formulated by focusing on learning easy samples whilst ignoring learning the noisy data. Then, the previously learnt knowledge by the 'clean' detector is again used to ease the training of the detectors in the noise-learning stage which focuses on learning diverse noisy data. The parameters of the detectors in the noise-learning stage are initialised by those of the 'clean' detector, which help the deep detectors avoiding poor local optimum during training and improving the convergence speed and system generalisation. Finally, to achieve a balance between running time and detection accuracy, we present a selective ensemble algorithm to choose several detectors with a large diversity for the final ensemble. In summary, our contributions can be summarised as follows:

- We propose a novel noise-robust deep detection framework which consists of a backbone network SWIPENET and a novel noise-robust training paradigm CMA. CMA drives the learning from clean to noisy data, it trains a robust deep ensemble detector for the object detection task in the underwater scenes with heterogeneous noisy data and small objects.

- SWIPENET fully takes advantage of both high resolution and semantic-rich Hyper Feature Maps that significantly boost small object detection. It applies a sample-weighted detection loss to control the influence of the training samples on SWIPENET according to their weights, we provide detailed theoretical analysis on the ability of the sample-weighted detection loss in this work.

- To achieve the balance between the detection accuracy and the computational cost, we propose a novel selective ensemble algorithm to choose the best detectors trained with large data diversity.

The rest of the paper is organised as follows. Section 2 gives a brief intro-

duction about the related work. Section 3 describes our proposed SWIPENET backbone, CMA training paradigm and selective ensemble algorithm. Section 4 describes the experimental set-up and Section 5 reports the results of the proposed method on four underwater object detection datasets.

2. Related Work

2.1. Underwater object detection

In recent years, several deep learning frameworks have been proposed in underwater object detection field. Fan et al. [6] proposed a deep network FERNet to extract multi-scale contextual features from the underwater images, they also introduced a anchor refinement module to solve the class imbalance problem. Lin et al. [7] proposed a data augmentation method RoIMix that focuses on interactions between images and mixes proposals among multiple images, this proposal-level data augmentation strategy greatly improves the performance of underwater object detectors. Moreover, several works directly employed general deep object detection networks, such as Faster RCNN [8] and YOLOv3 [9], for underwater object detection task. However, the existence of small objects and noisy objects in underwater datasets greatly degraded the accuracy of these general detection frameworks. To address the small object detection problem, different strategies had been explored. Bosquet et al. [10] proposed an end-to-end spatio-temporal convolutional neural network for small object detection, while Shuang et al. [11] designed a novel scale-balanced loss for deep detection framework, all these two strategies boosted the detection accuracy of the small objects. In blurry underwater scenes, the existence of noisy data confused the detection frameworks that cannot distinguish the noisy objects from the complex background, to address this problem, Chen et al. [12] took the noisy data as outliers, and proposed an Invert Multi-Class Adboost (IMA) algorithm to ignore learning these possible outliers, which achieves good performance on noisy underwater datasets. However, the noisy data contain not only disturbing outliers but also hard objects, which are effective training

samples for deep neural networks. IMA avoiding learning all the noisy data cannot detect many hard objects well that damaged the generalisation ability of the deep model. To improve the generalization on hard object detection, we proposed a novel training paradigm CMA that drives the learning from clean data to noisy data.

2.2. Curriculum leaning paradigm

In the human education system, it may confuse the learner if s/he directly learns the hard knowledge in the beginning. Instead, the beginner starts from learning easy knowledge while skipping disturbing hard knowledge, in such way, the learning exercise is efficient and effective [13]. This idea is also widely used in many machine leaning algorithms. For example, curriculum learning [14, 15] and self-pace learning [16, 17] are two representatives inspired by the idea of learning easier aspects of the task before moving into a difficult level. Both approaches have been reported to provide better generalisation for the used model [18]. However, Curriculum learning requires the samples in the datasets to be ranked in the order of incremental difficulty levels, but preparing such datasets is not trivial at all in practice. Self-pace learning addresses the sample order issue by training the used model and ranking the samples according to the samples' loss values using the learned model. It assumes the samples with low loss values are easy samples. One major drawback of self-pace learning is that it does not incorporate prior knowledge into the learning and hence loose the generalisation ability. Moreover, both curriculum learning and self-pace learning methods only train a single model without considering the limited capacity of the single model to learn diverse data [19, 20]. The developed models may be over-fit on some samples and under-fit on other samples. In our work, we combine the learning tricks from Curriculum Learning and Multi-Class Adaboost into a novel noise-robust training paradigm CMA, which dynamically trains multiple detectors on the samples with a large diversity and combines them into a unified noise-robust deep ensemble detector.

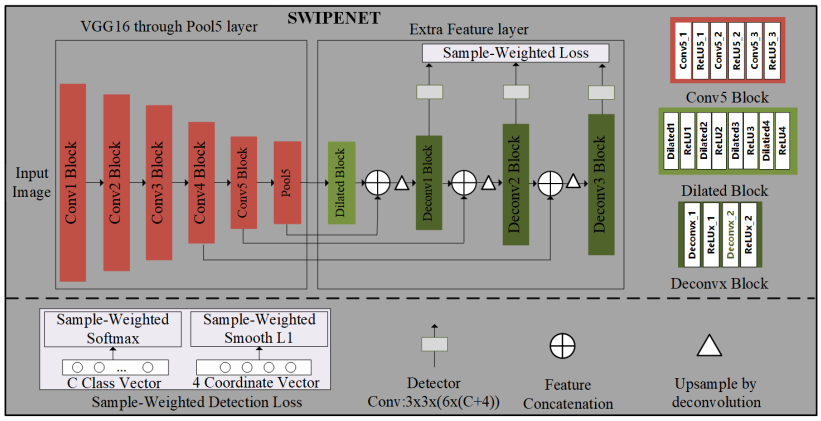

Figure 2: The structure of our proposed SWIPENET backbone.

3. SWIPENET+CMA framework

Deep learning has shown great advantages over other techniques in various computer vision tasks due to its powerful feature representation capacity. As the the large-scale underwater datasets increase, we aim to develop a novel deep detection framework for underwater object detection. The complete framework consists of a backbone network SWIPENET and a noise-robust training paradigm CMA. We first introduce the backbone of SWIPENET and its sample-weighted loss function. Then, we present the CMA training paradigm. The preliminary version of SWIPENET was published in our previous conference paper [12]. To complement the work of [12], we provide detailed theoretical analysis on the ability of the sample-weighted detection loss in this work.

3.1. Sample-WeIghted hyPEr Network (SWIPENET)

3.1.1. The backbone of SWIPENET

The SWIPENET backbone includes several high resolution and semantic-rich Hyper Feature Maps inspired by Deconvolutional Single Shot Detector (DSSD) [21]. Different from DSSD, we design a dilated convolution block in SWIPENET to obtain large receptive fields without sacrificing detailed information that support object localization (large receptive fields lead to strong

semantics). The proposed dilated convolution block consists of 4 dilated convolution layers with ReLU activation and its detailed implementation can be found in Supplementary Section 1. Fig. 2 illustrates the overview of our proposed SWIPENET, which consists of multiple convolution blocks, a novel dilated convolution block, multiple deconvolution blocks and a novel sample-weighted loss. The front layers of the SWIPENET are based on the architecture of the standard VGG16 model [22] (truncated at the Pool5 layer). Then, we add the proposed dilated convolution block to extract high semantic while keep the resolutions of the feature maps. Finally, we up-sample the feature maps using deconvolution and add skip connection to construct multiple Hyper Feature Maps on the deconvolution layers.

3.1.2. Sample-Weighted detection loss

We propose a novel sample-weighted detection loss function which enables SWIPENET to control the influence of the training samples according to their weights. It cooperates with a novel sample re-weighting algorithm, namely Curriculum Multi-Class Adaboost, to address the noisy data problem in underwater object detection.

Technically, our sample-weighted detection loss L consists of a sample-weighted softmax loss L_{cls} for the bounding box classification and a sample-weighted smooth L1 loss L_{reg} for the bounding box regression:

$$L = \frac{\alpha_1}{\ddot{N}} L_{cls}(pre\_cls, gt\_cls, \bar{w}) + \frac{\alpha_2}{\bar{N}} L_{reg}(pre\_loc, gt\_loc, \bar{w}) \tag{1}$$

where $\ddot{N}$ and $\bar{N}$ are the numbers of all the training samples and positive training samples respectively, α_1 and α_2 denote the weight terms of classification and regression losses. The sample-weighted softmax loss L_{cls} is formulated as

$$L_{cls} = -\sum_{i=1}^{\ddot{N}} \sum_{c=1}^{C+1} \bar{w}_i^m gt\_cls_i^c log(pre\_cls_i^c) \tag{2}$$

$$pre\_cls_i^c = \frac{e^{net_i^c}}{\sum_{\bar{c}=1}^{C+1} e^{net_i^{\bar{c}}}} \tag{3}$$

where $\bar{w}_i^m$ denotes the sample weight for the i-th sample computed in the m-th iteration of CMA in Subsection 3.2. Denote pre_cls_i and gt_cls_i as the

predicted and ground truth class vectors for the i-th sample, these two vectors are $C+1$-D vectors (C object classes plus one background class). $pre\_cls_i^c$ and $gt\_cls_i^c$ denote the c-th element of the predicted and ground truth class vectors for the i-th sample (referring to Supplementary Fig. 1 for better understanding). $gt\_cls_i^c = 1$ if the i-th sample belongs to the c-th class, $gt\_cls_i^c = 0$ otherwise. net_i^c is the classification prediction from the detection network. L_{reg} is the sample-weighted smooth L1 loss, formulated as follows:

$$L_{reg} = \sum_{i=1}^{\bar{N}} \sum_{l \in Loc} \bar{w}_i^m Smooth_{L_1}(pre\_loc_i^l - gt\_loc_j^l) \tag{4}$$

$$Smooth_{L_1}(x) = \begin{cases} 0.5x^2 & \textbf{if } |x| < 1 \\ |x| - 0.5 & \textbf{otherwise} \end{cases} \tag{5}$$

$$pre\_loc_i^l = net_i^l, l \in Loc \tag{6}$$

pre_loc_i and gt_loc_i denote the predicted and ground truth coordinate vectors for the i-th sample, these two vectors are 4-D vectors (the coordinate information $Loc = (cx, cy, w, h)$ includes the coordinates of center (cx, cy) with width w and height h. $pre\_loc_i^l$ and $gt\_loc_i^l$ denote the l-th element of the predicted and the ground truth coordinate vectors for the i-th positive training sample respectively. net_i^l is the coordinate prediction from the detection network.

In the gradient based optimisation algorithm, the loss function plays a key role in providing the gradients for updating the model parameters in the back-propagation process. The sample's gradient magnitude in the derivative of the loss function determines its impact on the updating of the DNNs. In our proposed sample-weighted detection loss, the sample weight $\bar{w}_i^m$ is able to adjust the sample's gradient magnitude. Hence, we are able to investigate how the sample weight influences the sample's impact on the feature learning of DNNs. Denote the parameter of the detector as θ, the derivative of the sample-weighted detection loss $\frac{\partial L}{\partial \theta}$ is derived as (the detailed derivation process can be found in

the Supplementary Section 3):

$$\frac{\partial L}{\partial \theta} = \begin{cases} \frac{\alpha_1}{\ddot{N}} \sum_{i=1}^{\ddot{N}} \sum_{c=1}^{C+1} \bar{w}_i^m gt\_cls_i^c (pre\_cls_i^c - 1) \frac{\partial net_i^c}{\partial \theta} \\ + \frac{\alpha_2}{\bar{N}} \sum_{i=1}^{\bar{N}} \sum_{l \in Loc} \bar{w}_i^m (pre\_loc_i^l - gt\_loc_j^l) \frac{\partial net_i^l}{\partial \theta} \\ \qquad\qquad \textbf{if } |pre\_loc_i^l - gt\_loc_j^l| < 1 \\ \frac{\alpha_1}{\ddot{N}} \sum_{i=1}^{\ddot{N}} \sum_{c=1}^{C+1} \bar{w}_i^m gt\_cls_i^c (pre\_cls_i^c - 1) \frac{\partial net_i^c}{\partial \theta} \\ \pm \frac{\alpha_2}{\bar{N}} \sum_{i=1}^{\bar{N}} \sum_{l \in Loc} \bar{w}_i^m \frac{\partial net_i^l}{\partial \theta} \qquad \textbf{otherwise} \end{cases} \tag{7}$$

From Eq. (7), we witness that the sample's gradient magnitude in the derivative is influenced by two factors. The first one is the accuracy of the predicted class and coordinates. For the i-th training sample with ground truth class c (i.e., $gt\_cls_i^c = 1$), the closer $pre\_cls_i^c$ and $pre\_loc_i^c$ to the ground truth, the smaller the gradient magnitude for the i-th sample. Second, the sample's weight $\bar{w}_i^m$. Suppose all of the training samples have the same prediction accuracy. The smaller the weight is, the smaller gradient magnitude is attached to the i-th sample. For example, if we assign a weight of 100 and 1 to the same positive sample respectively, then the gradient magnitude of the former one may be around 100 times that of the later one. Hence, the feature learning of DNN is dominated by high-weight samples while the low-weight samples contribute far less to the update of the DNN features.

3.2. Curriculum Multi-class Adaboost (CMA)

Underwater images suffer from different degradations, e.g. poor lighting, noise and blurs. These factors generate considerable noisy data, which are hard object instances and visually similar to the background. The deep neural networks directly trained on the noisy data may lead to significant errors between the object classes and the background as shown on the bottom of Fig. 1. Inspired by the human education system that learns from easy to hard samples, we here propose a novel training paradigm, namely Curriculum Multi-class Adaboost (CMA), to address the noisy data problem in underwater object detection.

3.2.1. The overview of the CMA

CMA is developed based on Multi-Class Adaboost (MA) [23], which trains multiple base classifiers sequentially and assign a weight value α_m to each clas-

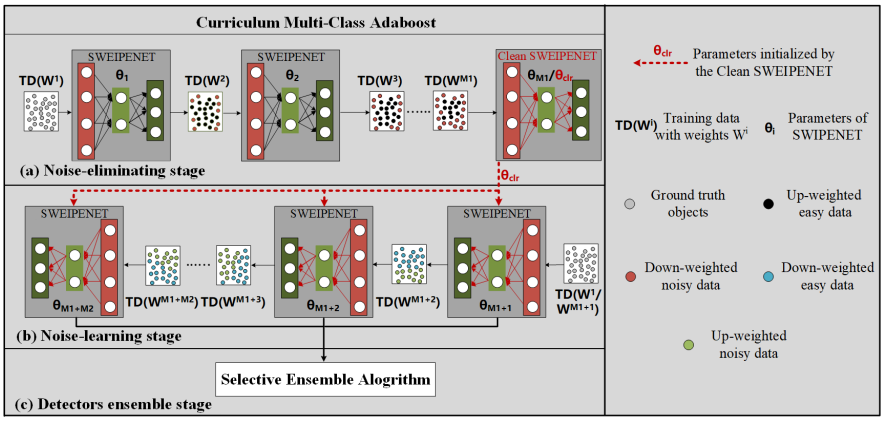

Figure 3: The overview of CMA training paradigm. It consists of the (a) Noise-eliminating stage (NECMA): gradually reduce the weights of the possible noisy data to obtain a 'clean' detector which is free from the influence of the noisy data. (b) Noise learning stage (NLCMA): learn diverse noisy samples by increasing their weights to boost the generalisation ability. The parameters of each detector in NLCMA are initialised by those of the 'clean' SWIPENET, that alleviates the local optimum problem and accelerate the convergence, and (c) Detectors ensemble stage: ensemble multiple detectors to boost the generalisation ability.

sifier according to its error rate E_m. When training each classifier, the samples misclassified by the preceding classifier are assigned a higher weight, allowing the following classifier to focus on learning these samples. Finally, all the weak base classifiers are combined to form an ensemble classifier with corresponding weight values.

Different form MA, our proposed CMA algorithm consists of three stages as shown in Fig. 3: noise-eliminating stage (denotes as NECMA), noise-learning stage (denotes as NLCMA), and detectors ensemble stage. In each training iteration of NECMA, we reduce the weights of the undetected objects as they are likely to be noisy data [5]. The sample-weighted detection loss enables the next SWIPENET to only focus on learning the high-weight clean data. By gradually reducing the influence of the noisy data, the detectors in the NECMA stage produces less errors between the objects and the complex background. However, after several iterations, the deep detector may over-fit over the clean, easy samples as their weights are too high after several rounds of re-weighting

exercises. Therefore, we terminate the NECMA stage when the performance does not improve anymore, and the detector achieving the best detection accuracy is selected as the 'clean' detector (SWEIPENET). The 'clean' detector can detect the easy objects well but always fails to detect many hard objects as it ignores learning the noisy object instances. Hence, we propose the NLCMA training stage, which focuses on learning diverse hard samples by increasing their weights. The complete CMA algorithm greatly improves the generalisation capability of our detection framework on the noisy data.

The proposed CMA training paradigm (the pseudocode can be found in Supplementary Algorithm 1) iteratively trains M detectors, including M_1 iterations for NECMA and M_2 iterations for NLCMA. We assume the best performing detector (i.e, the 'clean' detector S_{clr} parameterised by θ_{clr}) in NECMA is achieved in the M_1-th iteration, M_1 is experimentally obtained. Denote I_{train} as the training images with the ground truth objects $B = \{b_1, b_2, ..., b_N\}$, N is the number of the objects in the training set, $b_j = (cls, cx, cy, w, h)$ is the annotation of the j-th object. We denote w_j^m as the weight of the j-th object in the m-th iteration. Each object's weight is initialised to $\frac{1}{N}$ in the first iteration, i.e. $w_j^1 = \frac{1}{N}, j = 1, ..., N$.

In the m-th iteration of CMA, we firstly compute the weights of the positive training samples. If the i-th positive sample matches the j-th object during the training, we compute the i-th positive sample's weight $\bar{w}_i^m$ using Eq. (8).

$$\bar{w}_i^m = N * w_j^m, 0 < w_j^m < 1 \tag{8}$$

where w_j^m denotes the weight of the j-th object in the m-th iteration. The weight of the positive sample is N times that of its matched object. This is because the initial weight of each object in CMA is $\frac{1}{N}$, and the initial weight of each positive training sample in the sample-weighted detection loss is 1. Secondly, we use the re-weighted samples to train the m-th detector S_m. Thirdly, we run the m-th detector on the training set and receive the detection results $D_m = \{d_1, d_2, ..., d_i\}$ while $d_i = (cls, score, cx, xy, w, h)$ is the i-th predicted outcome, including the predicted class (cls), score ($score$) and coordinates (cx, cy, w, h).

The error rate E_m of the m-th detector is computed based on the percentage of the undetected objects.

$$E_m = \sum_{j=1}^{N} w_j^m I(b_j) / \sum_{j=1}^{N} w_j^m \tag{9}$$

where

$$I(b_j) = \begin{cases} 0 \ \textbf{if} \ \exists \, d \in D_m, \ s.t. b_j.cls == d.cls \wedge IoU(b_j, d) \geq \theta \\ 1 \ \textbf{otherwise} \end{cases} \tag{10}$$

In Eq. (10), if there exists a detection d which belongs to the same class as the j-th ground truth object b_j (i.e. $b_j.cls == d.cls$) and the Intersection over Union (IoU) between the detection and the j-th object is larger than the threshold θ (0.5 here), we set $I(b_j) = 0$, indicating the j-th object has been detected and $I(b_j) = 1$ is the undetected. Fourthly, we compute the m-th detector's weight α_m using Eq. (11), which is used when we ensemble different detectors.

$$\alpha_m = log \frac{1 - E_m}{E_m} + log(C - 1) \tag{11}$$

$$w_j^m \leftarrow \frac{w_j^m}{z_m} exp(\alpha_m (1 - I(b_j))) \tag{12}$$

where C is the number of the object classes. Finally, we update each object's weight w_j^m and train the following detector. In the first M_1 iterations of NECMA stage, we reduce the weights of the undetected objects by Eq. (12) that enables the next detector to pay less attention to possible noisy data. In the last M_2 iterations of the NLCMA stage, we increase the weights of the undetected objects by Eq. (13), whereas the detector turns to learning the diverse hard data. z_m is a normalisation constant. The same iteration repeats again till all M detectors have been trained.

$$w_j^m \leftarrow \frac{w_j^m}{z_m} exp(\alpha_m I(b_j)) \tag{13}$$

It is noticed that when CMA changes from NECMA to NLCMA, i.e., in the $M_1 + 1$-th iteration, we must re-initialise the weight of each object as $\frac{1}{N}$. In each iteration of NLCMA, we initialise the parameter of each detector with the

parameter θ_{clr} of the 'clean' SWIPENET. This initialisation strategy provides a good initialisation for the following deep detectors which is important for the deep networks to avoid the local optimum problem in training and improve generalisation [5].

3.2.2. Selective ensemble algorithm

An ensemble model may be more accurate than a single model, but brings in additional computational overhead. Recent works have pointed out that the ensemble of selective deep models may not only be more compact but also stronger in the generalization ability than that of the overall deep models [24]. To reduce the computational costs, we only select a few detectors with large diversity for the final ensemble.

We here propose a greedy selection algorithm to select candidate detectors for the final ensemble. Firstly, we construct a candidate ensemble set E to add up the selected detectors, and initialise it with the detector achieving the highest detection accuracy among all the M_2 detectors in NLCMA as these detectors have not been confused by noisy data. Then, we gradually select a single detector D_{m^*} having the largest diversity with all the detectors in the candidate ensemble set and add it to the ensemble set, as formulated in Eq. (14).

$$D_{m^*} = \underset{m, D_m \notin E}{\arg\max} \sum_{D_n \in E} Q_{mn} \tag{14}$$

Here, we apply the commonly used Q statistic [25] to measuring the diversity of two detectors' performance.

$$Q_{mn} = \frac{N^{11}N^{00} - N^{01}N^{10}}{N^{11}N^{00} + N^{01}N^{10}} \tag{15}$$

Q_{mn} denotes the diversity between the performance of detectors D_m and D_n. N^{11} and N^{00} are the numbers of the objects detected and missed by the two detectors respectively. N^{01} is the total number of the objects missed by D_m and detected by D_n, N^{10} is the total number of the objects detected by D_m and missed by D_n. Maximum diversity is achieved at $Q_{mn} = -1$ when the two

detectors make different predictions (i.e., $N^{11} = N^{00} = 0$), and the minimum diversity is achieved at $Q_{mn} = 1$ when the two detectors generate identical predictions (i.e., $N^{01} = N^{10} = 0$).

After all the candidate detectors have been selected, we ensemble them into a unified ensemble detector according to their weights computed by Eq. (11) in CMA and their diversity weight in the ensemble set. We assign a higher weight to the detector with a larger diversity. This enables the ensemble detector to detect diverse objects in the underwater scenes. We compute the diversity weight div_m of detector D_m as its average diversity with all the detectors in the ensemble set (by Eq. (16)).

$$div_m = \sum_{D_n \in E, n \neq m} Q^*_{mn}/(|E| - 1) \tag{16}$$

The value of Q_{mn} lies in [-1,1]. For better representing the weights of the detection model, we normalise Q_{mn} as Q^*_{mn} using Eq. (17). The value of Q^*_{mn} lies in [0,1], and the larger diversity the large value of the diversity weight.

$$Q^*_{mn} = 0.5(1 - Q_{mn}) \tag{17}$$

The final weight λ_i of detector D_i is formulated as

$$\lambda_i = \frac{div_i * \alpha_i}{\sum_{m=1}^{M^*} div_m * \alpha_m} M^*, i = 1, ..., M^* \tag{18}$$

In the testing stage, we use the weights to re-score the detection boxes. M^* denotes the number of the selected detectors, and $M^*/\sum_{m=1}^{M^*} div_m * \alpha_m$ in Eq. (18) is a normalisation term, scaling the score of the box to fall in [0,1] after re-scoring. In particular, we first run all M^* selected SWIPENETs on the testing set I_{test} and produce a M^* detection set Det_m.

$$Det_m = D_m(I_{test}), m = 1, 2, ..., M^* \tag{19}$$

Afterwards, we re-score each detection d in Det_m using λ_m.

$$d.score = \lambda_m d.score, d \in Det_m \tag{20}$$

Finally, we combine all the detections and utilise Non-Maximum Suppression to remove the overlapped detections.

4. Experiments Setup

In this section, we first introduce the experimental datasets. Then, we describe the implementation details.

4.1. Datasets

To demonstrate the effectiveness of the proposed method, we conduct comprehensive evaluations on four underwater object detection datasets. The former three underwater object detection datasets, including URPC2017, URPC2018, and URPC2019, come from the Underwater Robot Picking Contest [1], which is held by National Natural Science Foundation of China and Dalian Municipal People's Government. The fourth data set ChinaMM comes from the underwater image enhancement contest [2] and can be downloaded on the website of the contest. All the four datasets provide bounding-box level annotations.

The URPC2017 and ChinaMM datasets contain three object categories, including seacucumber, seaurchin and scallop. URPC2017 contains 18,982 training images and 983 testing images. ChinaMM contains 2,071 training images and 676 validation images. The URPC2018 and URPC2019 datasets contain four object categories, including seacucumber, seaurchin, scallop and starfish. URPC2018 and URPC2019 have published the training set, but the testing set is not publicly available. Hence, we randomly split the training set of URPC2018 into 1,999 training images and 898 testing images, and split the training set of URPC2019 into 3,409 training images and 1,000 testing images. URPC2017 have much more noisy data than the other three datasets. Moreover, all four datasets suffer from the class imbalance problem that the scallop and starfish

[1] Underwater Robot Picking Contest `http://en.cnurpc.org/index.html`

[2] Underwater Image Enhancement Contest `https://rwenqi.github.io/chinaMM2019uw/`

categories contain much more data than the seacucumber and seaurchin categories.

4.2. Implementation details

All the experiments are conducted on a server with an Intel Xeon CPU @ 2.40GHz and a single Nvidia Tesla P100 GPUs with a 16 GB memory. For our proposed detection framework, we implement it using the Keras framework, and train it with the Adam optimisation algorithm. We use an image scale of 512x512 as the input for both training and testing. On URPC2017, the batch-size is 16, and the learning rate is 0.0001. Our models often diverge when using a high learning rate due to unstable gradients, and all the detectors in the ensemble achieve the best performance after running 120 epochs. On URPC2018 and URPC2019, the batch-size is 16. We first train each detector in the ensemble with a learning rate 0.001 for 80 epochs, and then train them with a learning rate 0.0001 for another 40 epochs. On ChinaMM, the batch-size is 16, and the learning rate is 0.001. Each detector in the ensemble runs 120 epochs that enables each SWIPENET in CMA to achieve sufficient training. The source code will be made available at:`https://github.com/LongChenCV/SWIPENET+CMA`.

5. Ablation studies

In this section, we conduct the ablation experiments to investigate the influence of different components on our SWIPENET+CMA framework, including the skip connection, the dilated convolution block and the CMA training paradigm. In the next section, we compare our method against several state-of-the-art (SOAT) detection frameworks on four datasets.

5.1. Ablation studies on the skip connection and dilated convolution block

To investigate the influence of skip connection, we design the first baseline network UWNET1 which has the same structure as SWIPENET except that it does not contain skip connection between the low and high layers. The second network UWNET2 replaces the dilated convolution block in UWNET1

Table 1: Ablation studies on four datasets. Skip indicates skip connection, and Dilation indicates dilated convolution block. mAP indicates mean Average Precision(%).

Network	Skip	Dilation	URPC2017	URPC2018	URPC2019	ChinaMM
UWNET1		✓	40.4	61.2	55.0	73.9
UWNET2			38.3	58.1	54.2	71.0
SWIPENET	✓	✓	42.1	62.2	57.6	76.1

with standard convolution block to learn the influence of the dilated convolution block. Table 1 shows the performance comparison of different networks on four datasets, we observe that SWIPENET performs better than UWNET1. The gains come from the skip connection which passes fine detailed information of the lower layers such as object boundary to the high layers that are important for object localisation. Compared to UWNET2, UWNET1 performs better because the dilated convolution block in UWNET1 brings much semantic information to the high layers which enhances the classification ability. We also present the mean Average Precision (mAP) of UWNET2 and SWIPENET for the objects with different sizes in Fig. 4 and Supplementary Section 6, from which we observe the skip connection and dilated convolution block largely improves the small object detection accuracy. For example, for small objects (S) of seacucumber, seaurchin and scallop categories, SWIPENET improves 5%∼6% mAP over UWNET2 on URPC2018 and ChinaMM.

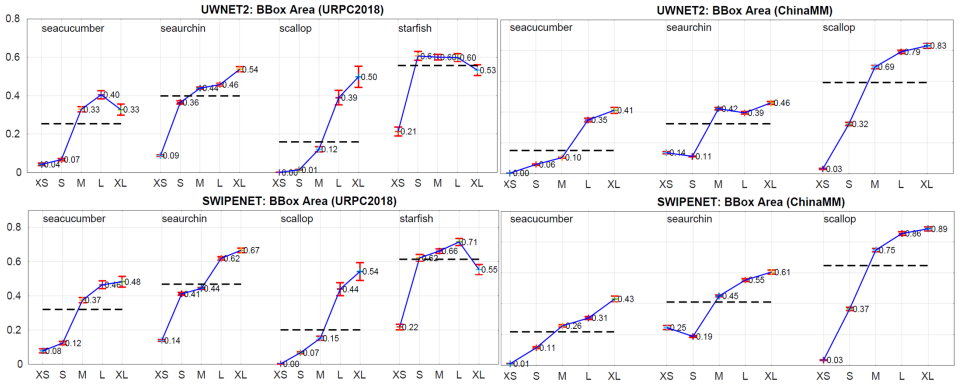

Figure 4: The mean Average Precision of UWNET2 and SWIPENET for objects with different object sizes on URPC2018 and ChinaMM. The object size is measured as the pixel area of the bounding box. XS (bottom 10%)=extra-small; S (next 20%)=small; M (next 40%)=medium; L (next 20%)=large; XL (next 10%)=extra-large.

5.2. Ablation studies on CMA

Table 2: The performance (mAP(%)) of SWIPENET in each iteration of CMA on test set of four datasets. The red numbers indicate the results of the 'clean' SWIPENETs.

Dataset	Stage	NECMA					NLCMA						
	Iteration	1	2	3	4	5	1	2	3	4	5	6	7
URPC2017	Single	42.1	44.2	45.3	40.5	37.2	47.5	47.2	46.2	47.9	**48.0**	47.0	47.6
	Ensemble	42.1	45.0	46.3	45.3	44.2	47.5	48.6	49.8	52.3	**52.5**	52.5	52.5
URPC2018	Single	62.2	63.3	62.4	61.2	59.3	65.0	64.8	**65.3**	64.5	64.5	63.9	64.3
	Ensemble	62.2	64.5	64.0	62.8	62.1	65.0	65.4	66.9	67.5	**68.0**	68.0	68.0
URPC2019	Single	57.6	58.5	57.2	56.9	56.1	**61.8**	61.5	61.6	61.0	59.5	61.5	61.0
	Ensemble	57.6	59.9	59.0	59.0	59.5	61.8	62.4	**63.9**	63.9	63.9	63.9	63.9
ChinaMM	Single	76.1	77.5	78.3	76.5	74.8	80.4	79.8	**82.3**	81.4	79.5	80.0	79.3
	Ensemble	76.1	78.5	79.9	77.8	78.5	80.4	81.9	83.4	**85.6**	85.5	85.6	85.6

In this subsection, we investigate the influence of CMA, including NECMA and NLCMA, on the final detection results. In our experiments, the number of the iterations of NECMA is set to 5 and the number of the iterations of NLCMA is set to 7. Table 2 shows the performance of the single model and the ensemble model in each iteration of NECMA and NLCMA.

The role of NECMA. From Table 2, in NECMA, we observe that the 'clean' SWIPENET is achieved in the 3rd iteration on URPC2017 and ChinaMM, and in the 2nd iteration on URPC2018 and URPC2019. So we set $M_1 = 3$ on UPRC2017 and ChinaMM, and $M_1 = 2$ on URPC2018 and URPC2019. The 'clean' SWIPENETs perform much better than the detectors in the 1st iteration. We assume this is because the noisy data confuse the detectors in the 1st iteration. Fig. 5 and Supplementary Section 7 show the top three false positives for the 1st detector, i.e. the SWIPENET trained without CMA, we can see that the background error (detecting the backgrounds as the objects) has much influence on the detectors than the localisation error (inaccurate localisation). To further verify this assumption, we use the detection analysis tool of [26] to analyse the false positives of the 1st detector and the 'clean' detector in NECMA. Fig. 6 and

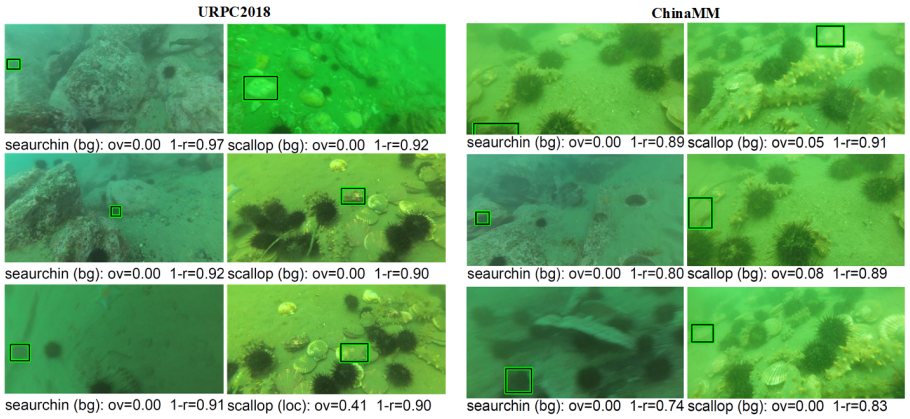

Figure 5: Examples of top false positives of the SWIPENET without CMA: We show the top three false positives (FPs) for the seaurchin and scallop categories on URPC2018 and ChinaMM. The text indicates the type of error ("loc"=localization; "bg"=confusion with backgrounds), the amount of overlap ("ov") with a true object, and the fraction of correct examples that are ranked lower than the given false positive ("1-r", for 1-recall). Localization errors are due to insufficient overlaps.

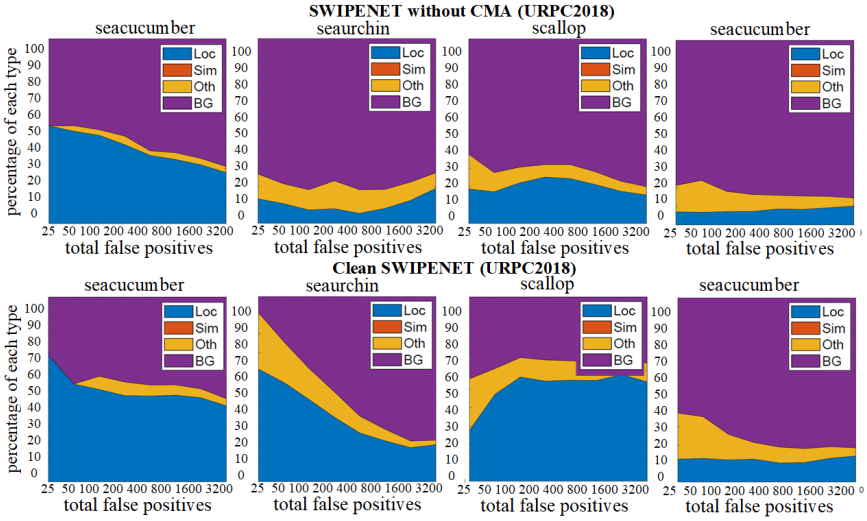

Figure 6: The distribution of top-ranked false positive types of the SWIPENET without CMA and the 'clean' SWIPENET for each category on URPC2018. The false positive types include localisation error (Loc), confusion with similar categories (Sim), with others (Oth), or with background (BG).

Supplementary Section 7 show the distribution of the top-ranked false positives for each category on four datasets. We can see that the 1st detector cannot well distinguish the objects with complex background and generate much more background errors than the 'clean' detector. NECMA gradually reduces the influence of the noisy data on the single detector by decreasing their weights, and the background error clearly decreases in the detection results of the 'clean' SWIPENET. However, the performance of the single detectors after the 'clean' SWIPENET is less satisfactory. This is because most of the detected objects are continuously up-weighted and the detectors over-fit over these high-weight objects.

Table 3: Performance comparisons of different initialization strategies in NLCMA on URPC2017.

Initialization strategy	Iteration	1	2	3	4	5	6	7
Clean initialization	Single	47.5	47.2	46.2	47.9	**48.0**	47.0	47.6
	Ensemble	47.5	48.6	49.8	52.3	**52.5**	52.5	52.5
Random initialization	Single	**40.6**	39.8	38.4	39.2	37.5	37.4	36.7
	Ensemble	40.6	**40.8**	40.6	40.0	40.5	40.0	40.0
1st detector initialization	Single	**43.0**	43.0	42.4	41.6	41.0	41.0	39.7
	Ensemble	43.0	**43.5**	42.5	43.0	42.9	42.5	42.7

The role of NLCMA. In NLCMA, we initialise each detector using the parameter learned in the 'clean' SWIPENET. This initialisation strategy provides a good initialisation for the following detectors which avoid getting stuck in poor local minima during the training. With this initialisation strategy, the detectors converge much faster during the training, shown in Fig. 7 (we also take the testing set as the validation set and investigate the influence of this initialisation strategy on the validation loss). To further verify the effectiveness of the clean initialisation strategy, we design two comparison initialisation strategies, including random initialisation (i.e., initialising each detector in NLCMA with random weights) and 1st detector initialization (i.e., initialising each detector in NLCMA with the weights of the 1st detector in NECMA). The performance

comparisons of different initialization strategies in NLCMA are presented in Table 3 and Supplementary Section 11, where the two comparison initialisation strategies present much worse mAP for both single and ensemble models. This is because the clean SWIPENET has learnt the basic feature representations of the objects from easy data. These basic feature representations work as the prior knowledge that help the detectors in NLCMA discover the minor difference between the noisy data and the background. For the other two comparison initialisation strategies, the detectors in NLCMA directly focus on learning the noisy data without any prior knowledge, which cannot learn discriminate feature representations to distinguish the noisy data with the background and hence frequently mistreat the background as the objects.

From Table 2, we can see all the detectors in NLCMA perform better than the 'clean' detectors. This is because the detectors in NECMA take all the undetected objects as the noisy data and ignore learning them, however, the undetected objects also contain many hard objects, which are hard to be detected due to their minor discrepancies with the backgrounds. The 'clean' detector trained by NECMA can only detect the easy objects well but mis-detect many hard objects that limits the generalization of the detector. Different from detectors in NECMA, the detectors in NLCMA are able to detect the hard objects with the help of the 'clean' SWIPENET. The fundamental knowledge learnt by the 'clean' SWIPENET helps the following detectors identify the minor discrepancies between the hard objects and the backgrounds.

5.3. Ablation studies on the selective ensemble algorithm.

We investigate the influence of selective ensemble algorithm (SE) on the performance of the final ensemble detector. Fig. 7 (the right figure) shows the performance of the ensemble detector with different numbers of the selected detectors. The SE algorithm reduces the number of the detectors in the final ensemble. For example, the ensemble detector without SE achieves the best mAP on URPC2017 and URPC2018 when we ensemble five detectors, but the ensemble detector with SE achieves the same mAP by only integrating three

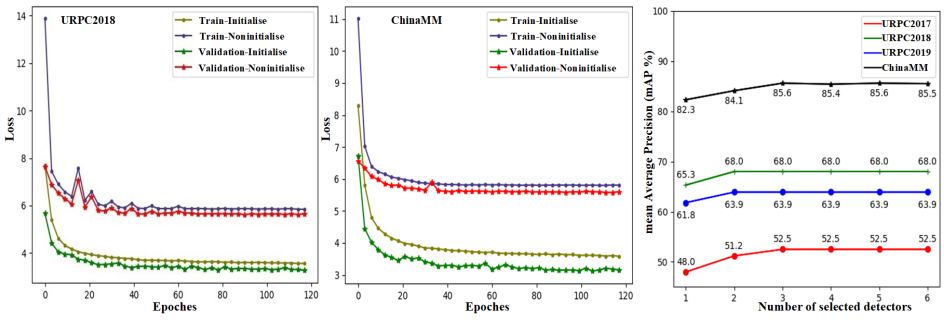

Figure 7: The learning curve of SWIPENETs with and without initialisation by the 'clean' SWIPENET on URPC2018 (left) and ChinaMM (middle), and the performance of the ensemble with different numbers of detectors (right).

selected detectors on URPC2017 and two selected detectors on URPC2018. This demonstrates some of the detectors do not help boosting the final performance in the ensemble. Few detectors with large diversity are sufficient to achieve the best performance. The selective ensemble algorithm surely helps reduce the computational overhead.

Table 4: Comparison with small object detection frameworks on URPC2018 and ChinaMM.

Dataset	URPC2018					ChinaMM			
Methods	seacucumber	seaurchin	scallop	starfish	mAP	seacucumber	seaurchin	scallop	mAP
DSSD [21]	48.4	75.3	38.2	64.0	56.5	54.5	82.0	79.4	72.0
FCOS [27]	43.2	76.5	47.5	69.4	59.1	57.7	83.1	78.7	73.2
RetinaNet [28]	52.5	74.9	43.1	69.8	60.1	59.6	82.0	81.0	74.2
FPN [29]	57.7	76.9	38.1	70.6	60.9	58.0	82.1	81.6	73.9
RetinaNet(S-α) [30]	54.4	76.5	**52.4**	71.7	63.8	60.8	82.0	82.7	75.2
FPN(S-α) [30]	**59.1**	77.0	39.2	71.4	61.7	62.0	82.4	82.7	75.7
OursnoCMA	46.4	**84.0**	40.2	78.2	62.2	63.0	83.5	81.9	76.1
OursSingle	54.8	81.5	46.6	**78.4**	**65.3**	**77.0**	**84.7**	**85.2**	**82.3**

6. Comparison with SOAT detection frameworks

In this section, we first compare our proposed method with latest small object detection methods. Then, we compare it with several SOAT underwater object detection frameworks. Finally, we compare our CMA learning paradigm with other learning paradigms.

6.1. Comparison with small object detection frameworks

Following the latest small object detection work [31], we select DSSD [21], RetinaNet [28], FCOS [27], FRCNN-FPN [29], and layer fusion strategy S-α [30] as the small object detection comparison methods. For fair comparison, we only compare our single models OursnoCMA (the SWIPENET trained without CMA) and OursSingle (the best single model achieved in the CMA) with other detection frameworks without considering the ensemble model.

Implementation details. For RetinaNet and FCOS, we use ResNet50 [32] as the backbone network. For DSSD and FRCNN-FPN, we use their original backbone networks. Following [30], we use FRCNN-FPN and RetinaNet with layer fusion strategy S-α as the detection frameworks. Both use ResNet50 [32] backbone. The comparison methods are tuned to have the best performance.

The experimental results on URPC2018 and ChinaMM are shown in Table 4, from which we observe OursnoCMA performs much better than DSSD, this is because multiple down-sampling operations lost many useful features, which are importance for accurate small object localization, these features cannot fully recovered by up-sampling operations once lost. The dilated convolution block in SWIPENET retains these features that benefits object localisation. On three datasets, OursSingle achieves the best performance, its advantage comes from the SWIPENET backbone and the noisy eliminating strategy. It is worth noting that FCOS and RetinaNet and FPN frameworks apply much deeper backbones (ResNet50) than our SWIPENET, but OursnoCMA still achieves better performance than the former three frameworks on URPC2018 and ChinaMM, this demonstrates the multiple Hyper Features in SWIPENET is able to detect multi-scale objects well. FPN with S-α achieves the best performance on URPC2019 (the results can be found in Supplementary Section 5), this is because the layer fusion strategy S-α greatly boost the performance of small object detection, but it cannot solve the noise problem.

6.2. Comparison with underwater object detection frameworks

We compare our method against several detection frameworks have ever applied for underwater object detection in recent literature [12, 33], we only select the comparison methods whose source code is public available online, including IMA [12], SSD [4], YOLOv3 [9], FRCNN [8], RetinaNet [28], FCOS [27], FreeAnchor [34] and GHM [35].

Table 5: Comparison with underwater object detection methods on URPC2018 and ChinaMM.

Dataset		URPC2018					ChinaMM			
Methods	Backbone	seacu	seaurchin	scallop	starfish	mAP	seacu	seaurchin	scallop	mAP
SSD [4]	VGG16 [22]	44.2	84.4	35.8	78.1	60.6	47.3	80.3	78.1	68.6
YOLOv3 [9]	DarkNet53 [9]	35.7	83.0	34.0	77.9	57.7	33.1	80.2	77.9	63.7
FRCNN [8]	VGG16 [22]	43.3	83.0	32.0	74.5	58.2	38.5	77.9	77.1	64.5
FRCNN [8]	ResNet50 [32]	41.1	83.2	34.5	77.2	59.0	41.0	81.0	78.1	66.7
FRCNN [8]	ResNet101 [32]	44.3	82.5	34.7	77.5	59.8	51.7	81.5	79.5	70.9
FRCNN [8]	FPN [29]	57.7	76.9	38.1	70.6	60.9	58.0	82.1	81.6	73.9
IMA [12]	SWIPENET [12]	52.8	84.1	42.9	78.0	64.5	68.3	83.3	84.5	78.7
RetinaNet [28]	ResNet50 [32]	52.5	74.9	43.1	69.8	60.1	59.6	82.0	81.0	74.2
FCOS [27]	ResNet50 [32]	43.2	76.5	47.5	69.4	59.1	57.7	83.1	78.7	73.2
FreeAnchor [34]	ResNet50 [32]	46.2	72.3	42.5	71.4	58.1	41.9	80.6	76.9	66.4
GHM [35]	ResNet50 [32]	52.4	78.4	42.1	71.5	61.1	53.7	82.1	82.3	72.7
OursSingle	SWIPENET	54.8	81.5	46.6	78.4	65.3	77.0	84.7	85.2	82.3
OursCMA	SWIPENET	**56.4**	**84.6**	**50.9**	**79.9**	**68.0**	**82.2**	**87.1**	**87.6**	**85.6**

Implementation details. For SSD, we use VGG16 [22] as the backbone. For Faster RCNN, we use four backbones including VGG16, ResNet50 [32], ResNet101 [32] and FPN [29]. For YOLOv3, we use its original DarkNet53 network. RetinaNet, FCOS, FreeAnchor and GHM all use ResNet50 [32] as the backbones. The comparison methods are tuned to have the best performance.

Table 5 shows the experimental results on URPC2018 and ChinaMM, where OursCMA achieves the best performance than other comparison methods. FRCNN with FPN performs better than FRCNN with ResNet101, ResNet50 and VGG16, where the deeper backbone FPN plays a critical role. OursSingle, the best single SWIPENET trained using CMA, outperforms the other frameworks by a large margin on three datasets, demonstrating the superiority of our

proposed CMA in dealing with noisy data. It performs even better than the ensemble model trained with the IMA algorithm. This is because IMA regards all the undetected objects as outliers and ignore learning them, which loses considerable effective hard training samples. The undetected objects tend to be noisy data or outliers, they also contain many hard object instances. Ignoring these hard object instances, IMA can only detect the easy objects well but cannot detect many hard objects. Similarly, GHM avoids learning noisy data, it can avoid the influence of the noisy data but cannot generalize well on the hard object instances. RetinaNet is easily to overfit on the noisy data because it employed the focal loss to train the detection network which emphasis on learning hard, noisy data. Different from IMA and GHM, NLCMA stage of CMA focuses on learning hard object instances by increasing their weights, that improve the generalization on hard objects instances. OursCMA further improves OursSingle. The gain comes from its capacity to detect the diverse hard object instances. Fig. 8 and Supplementary Section 8 show the Precision/Recall curves of different detection methods on four datasets, where we observe OursCMA (black curve) achieves the best performance across all the object categories on URPC2017, URPC2018 and ChinaMM. Fig. 9 presents visualization of object detection results of different detection frameworks on URPC2018 and ChinaMM (the visualization on URPC2017 and URPC2019 can be found in the Supplementary Section 9), most of the detection frameworks cannot detect the small objects well, some of them detected the backgrounds as the objects. Among them, OursCMA performs best.

6.3. Comparison with representative learning paradigms

CMA combines the learning tricks from Multi-Class Adaboost [12] and Curriculum Learning [15], hence, we also conduct additional experiments to further compare our CMA learning paradigm with these two learning paradigms.

Implementation details. SWIPENET+MA train multiple detectors using the Multi-Class Adaboost algoithm and finally ensemble them into a unified model, focusing on learning undetected samples by up-weighting their weights.

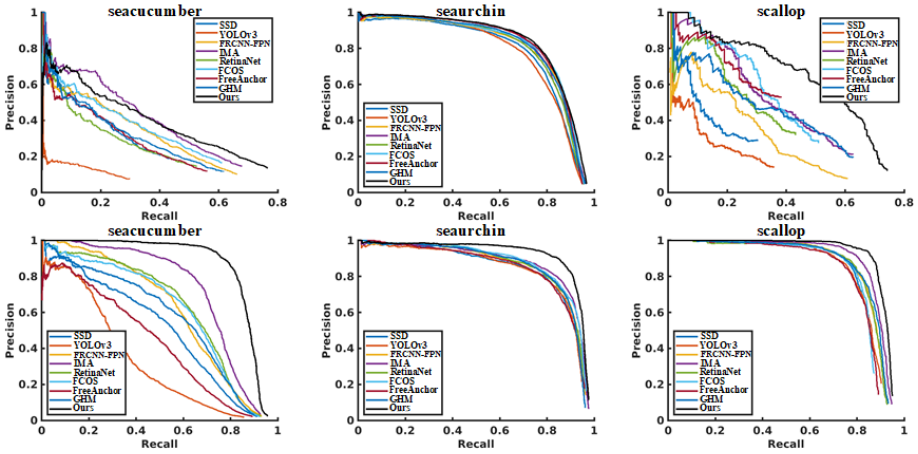

Figure 8: Precision/Recall curves of different detection methods on URPC2017 (top row) and ChinaMM (bottom row).

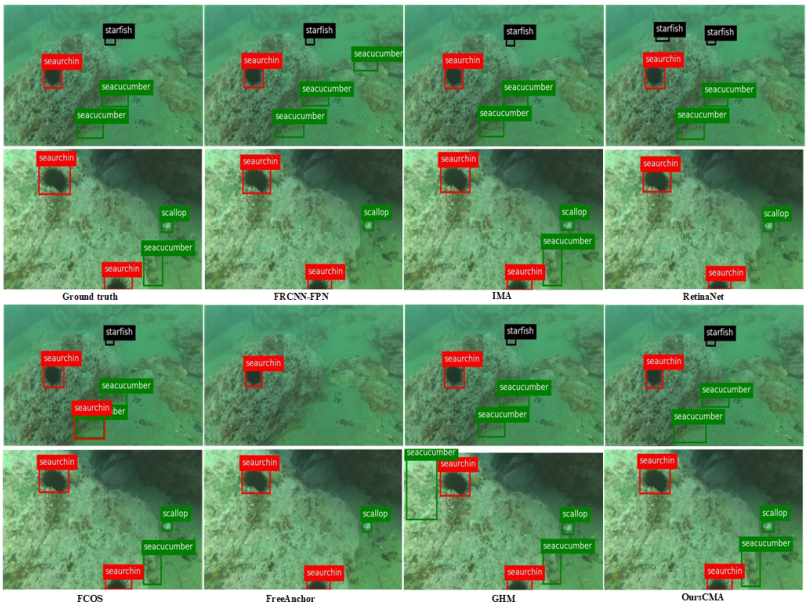

Figure 9: Visualization of object detection results of different detection frameworks on URPC2018 (top row) and ChinaMM (bottom row).

Table 6: The performance (mAP(%)) of SWIPENET in each iteration of different training paradigm on the test set of URPC2017, URPC2018 and ChinaMM.

Dataset	Iteration	1	2	3	4	5	6	7	8
URPC2017	SWIPENET+CMA	42.1	45.0	46.3	47.5	48.6	49.8	52.3	**52.5**
	SWIPENET+MA	**42.1**	41.0	40.5	39.2	39.5	38.8	40.2	39.8
	SWIPENET+Curr	42.1	41.0	**43.9**	-	-	-	-	-
URPC2018	SWIPENET+CMA	62.2	64.5	65.0	65.4	66.9	67.5	**68.0**	68.0
	SWIPENET+MA	**62.2**	62.0	61.0	61.2	60.1	58.8	60.2	59.3
	SWIPENET+Curr	62.2	62.1	**63.8**	-	-	-	-	-
URPC2019	SWIPENET+CMA	57.6	59.9	61.8	62.4	**63.9**	63.9	63.9	63.9
	SWIPENET+MA	**57.6**	56.2	57.0	57.6	56.9	56.8	55.8	56.3
	SWIPENET+Curr	57.6	56.9	**60.8**	-	-	-	-	-
ChinaMM	SWIPENET+CMA	76.1	78.5	79.9	80.4	81.9	83.4	**85.6**	85.5
	SWIPENET+MA	76.1	**77.0**	76.5	76.0	75.5	75.7	75.0	74.7
	SWIPENET+Curr	76.1	75.5	**78.2**	-	-	-	-	-

SWIPENET+Curriculum first trains a detector on the easy samples, then fine-tunes the detector of hard samples, since curriculum paradigm needs to define the easy and hard training samples: Similar to [5] that takes misclassified samples as the hard samples, we take the undetected objects as hard samples and the detected objects as easy samples. Specially, we first train a detector on all the training data, then we test the detector on the training data, the detected objects as easy and undetected objects as hard samples.

Table 6 shows the performance comparison of different training paradigms. Our CMA performs much better than the other training paradigms on all four datasets. After the 1st iteration, MA enable the detectors to focus on learning the hard data that degrade the system performance. This is because the noisy data confuse the detectors that cannot learn discriminate feature representations to distinguish the objects from the backgrounds. On the four datasets, Curriculum decays the performance in the 2nd iteration but boosts the performance in the 3rd iteration. This is because Curriculum trains the detector using insufficient easy samples in the 2nd iteration. After having fine-tuned over the remaining hard samples, the performance is better than that in the 1st

iteration. The gains come from the easy-to-hard training strategy and sufficient training data. However, CMA still performs much better than Curriculum. This is because the underwater datasets contain considerable diverse data resources due to frequently changing illuminations and environments, the ensemble model is able to learn diverse data and performs much better than the single model trained using the Curriculum paradigm whose generalisation ability is limited.

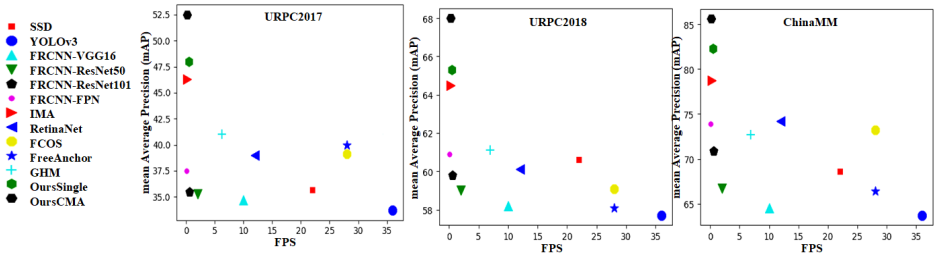

Figure 10: Running time (Frames Per Second, FPS) vs mean Average Precision (mAP) of different detection frameworks.

7. Conclusion

This paper proposes a noise-robust detection framework SWIPENET+CMA for underwater object detection. In the framework, the SWIPENET backbone can extract robust features for accurate small object detection. The noise-robust CMA training paradigm first trains a 'clean' detector which is free from the influence of noisy data. Then, based on the 'clean' detector, multiple detectors focusing on learning diverse noisy data are trained and incorporated into a unified deep ensemble of strong noise immunity. **(Insights)** This paper demonstrates the necessity of addressing the noisy issue for the underwater object detection task, it also offers a compelling insight on the training strategy of deep detectors in underwater scenes where noisy data exist. **(Strengths)** Our proposed method well-handles the noise issue in underwater object detection and achieves the excellent performance on the challenging underwater datasets. **(Weaknesses)** However, since it is an ensemble deep model, the time complexity is much higher than current popular single models (as shown in Fig. 10).

(Future works) Hence, in our future work, reducing the computational complexity of our proposed method is of vital importance. In the future work, we will extend our proposed method to more general application sceneries where considerable noise exits.

ACKNOWLEDGEMENT

Thanks for National Natural Science Foundation of China and Dalian Municipal People's Government providing the underwater object detection datasets for research purposes. Haiping Ma is supported by Zhejiang Provincial Natural Science Foundation of China under Grant No. LY19F030011.

References

[1] A. Sahoo, S. K. Dwivedy, P. Robi, Advancements in the field of autonomous underwater vehicle, Ocean Engineering 181 (2019) 145–160.

[2] P. I. Macreadie, D. L. McLean, P. G. Thomson, J. C. Partridge, D. O. Jones, A. R. Gates, M. C. Benfield, S. P. Collin, D. J. Booth, L. L. Smith, et al., Eyes in the sea: unlocking the mysteries of the ocean using industrial, remotely operated vehicles (rovs), Science of the Total Environment 634 (2018) 1077–1091.

[3] D. Akkaynak, T. Treibitz, A revised underwater image formation model, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 6723–6732.

[4] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A. C. Berg, Ssd: Single shot multibox detector, in: European conference on computer vision, Springer, 2016, pp. 21–37.

[5] I. Derényi, T. Geszti, G. Györgyi, Generalization in the programed teaching of a perceptron, Physical Review E 50 (4) (1994) 3192.

[6] B. Fan, W. Chen, Y. Cong, J. Tian, Dual refinement underwater object detection network, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX 16, Springer, 2020, pp. 275–291.

[7] W.-H. Lin, J.-X. Zhong, S. Liu, T. Li, G. Li, Roimix: Proposal-fusion among multiple images for underwater object detection, in: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2020, pp. 2588–2592.

[8] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, Advances in neural information processing systems 28 (2015) 91–99.

[9] A. Farhadi, J. Redmon, Yolov3: An incremental improvement, in: Computer Vision and Pattern Recognition, Springer Berlin/Heidelberg, Germany, 2018, pp. 1804–02.

[10] B. Bosquet, M. Mucientes, V. M. Brea, Stdnet-st: Spatio-temporal convnet for small object detection, Pattern Recognition 116 (2021) 107929.

[11] K. Shuang, Z. Lyu, J. Loo, W. Zhang, Scale-balanced loss for object detection, Pattern Recognition 117 (2021) 107997.

[12] L. Chen, Z. Liu, L. Tong, Z. Jiang, S. Wang, J. Dong, H. Zhou, Underwater object detection using invert multi-class adaboost with deep learning, in: 2020 International Joint Conference on Neural Networks (IJCNN), IEEE, 2020, pp. 1–8.

[13] D. L. Rohde, D. C. Plaut, Language acquisition in the absence of explicit negative evidence: How important is starting small?, Cognition 72 (1) (1999) 67–109.

[14] N. Sarafianos, T. Giannakopoulos, C. Nikou, I. A. Kakadiaris, Curriculum learning of visual attribute clusters for multi-task classification, Pattern Recognition 80 (2018) 94–108.

[15] Y. Bengio, J. Louradour, R. Collobert, J. Weston, Curriculum learning, in: Proceedings of the 26th annual international conference on machine learning, 2009, pp. 41–48.

[16] M. Kumar, B. Packer, D. Koller, Self-paced learning for latent variable models, Advances in neural information processing systems 23 (2010) 1189–1197.

[17] X. Shi, Z. Guo, F. Xing, J. Cai, L. Yang, Self-learning for face clustering, Pattern Recognition 79 (2018) 279–289.

[18] C. Han, D. Zhou, Y. Xie, M. Gong, Y. Lei, J. Shi, Collaborative representation with curriculum classifier boosting for unsupervised domain adaptation, Pattern Recognition 113 (2021) 107802.

[19] J. Xu, W. Wang, H. Wang, J. Guo, Multi-model ensemble with rich spatial information for object detection, Pattern Recognition 99 (2020) 107098.

[20] D. Chakraborty, V. Narayanan, A. Ghosh, Integration of deep feature extraction and ensemble learning for outlier detection, Pattern Recognition 89 (2019) 161–171.

[21] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, A. C. Berg, Dssd: Deconvolutional single shot detector, arXiv preprint arXiv:1701.06659.

[22] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: International Conference on Learning Representations, 2015.

[23] T. Hastie, S. Rosset, J. Zhu, H. Zou, Multi-class adaboost, Statistics and its Interface 2 (3) (2009) 349–360.

[24] L. Yang, Classifiers selection for ensemble learning based on accuracy and diversity, Procedia Engineering 15 (2011) 4266–4270.

[25] L. I. Kuncheva, C. J. Whitaker, Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy, Machine learning 51 (2) (2003) 181–207.

[26] D. Hoiem, Y. Chodpathumwan, Q. Dai, Diagnosing error in object detectors, in: European conference on computer vision, Springer, 2012, pp. 340–353.

[27] Z. Tian, C. Shen, H. Chen, T. He, Fcos: Fully convolutional one-stage object detection, in: Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 9627–9636.

[28] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal loss for dense object detection, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 2980–2988.

[29] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature pyramid networks for object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 2117–2125.

[30] Y. Gong, X. Yu, Y. Ding, X. Peng, J. Zhao, Z. Han, Effective fusion factor in fpn for tiny object detection, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2021, pp. 1160–1168.

[31] Y. Gong, X. Yu, Y. Ding, X. Peng, J. Zhao, Z. Han, Effective fusion factor in fpn for tiny object detection, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2021, pp. 1160–1168.

[32] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[33] C. Liu, Z. Wang, S. Wang, T. Tang, Y. Tao, C. Yang, H. Li, X. Liu, X. Fan, A new dataset, poisson gan and aquanet for underwater object grabbing, IEEE Transactions on Circuits and Systems for Video Technology 32 (5) (2021) 2831–2844.

[34] X. Zhang, F. Wan, C. Liu, X. Ji, Q. Ye, Learning to match anchors for visual object detection, IEEE Transactions on Pattern Analysis and Machine Intelligence.

[35] B. Li, Y. Liu, X. Wang, Gradient harmonized single-stage detector, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 8577–8584.