


## Article

# Active Mask-Box Scoring R-CNN for Sonar Image Instance Segmentation

Fangjin Xu <sup>†</sup>, Jianxing Huang <sup>†</sup>, Jie Wu <sup>†</sup> and Longyu Jiang <sup>\*</sup> 

School of Computer Science and Engineering, Southeast University, Nanjing 210000, China; xjf1109@seu.edu.cn (F.X.); chiehhsing\_huang@seu.edu.cn (J.H.); jie\_wu@seu.edu.cn (J.W.)

\* Correspondence: jly@seu.edu.cn

<sup>†</sup> These authors contributed equally to this work.

**Abstract:** Instance segmentation of sonar images is an effective method for underwater target recognition. However, the mismatch among positioning accuracy found by boxIoU and classification confidence, which is used as NMS score in current instance segmentation models; and the high annotation cost of sonar images, are two major problems in the task. To tackle these problems, in this paper, we present a novel instance segmentation method called Mask-Box Scoring R-CNN and embedded it in our proposed deep active learning framework. For the mismatch problem between boxIoU and NMS score, Mask-Box Scoring R-CNN uses a boxIoU head to predict the quality of the bounding boxes. We amend the non-maximum suppression (NMS) score predicted by BoxIoU to preserve high-quality bounding boxes in inference flow. To deal with the annotating problem, we propose a triplets-measure-based active learning (TBAL) method and a balanced-sampling method applicable for deep learning. The TBAL method evaluates the amount of information of unlabeled samples from the aspects of classification confidence, positioning accuracy, and mask quality. The balanced-sampling method selects hard samples from the dataset to train the model to improve performance. The experimental results show that Mask-Box Scoring R-CNN achieves improvements of 1% in boxAP and 1.3% boxAP on our sonar image dataset compared with Mask Scoring R-CNN and Mask R-CNN, respectively. The active learning framework with TBAL and balanced sampling can achieve a competitive performance with less labeled samples than other frameworks, which can better facilitate underwater target recognition.

**Keywords:** deep learning; active learning; instance segmentation; sonar image



**Citation:** Xu, F.; Huang, J.; Wu, J.; Jiang, L. Active Mask-Box Scoring R-CNN for Sonar Image Instance Segmentation. *Electronics* **2022**, *11*, 2048. <https://doi.org/10.3390/electronics11132048>

Academic Editor: Yu Zhang

Received: 4 May 2022

Accepted: 27 June 2022

Published: 29 June 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

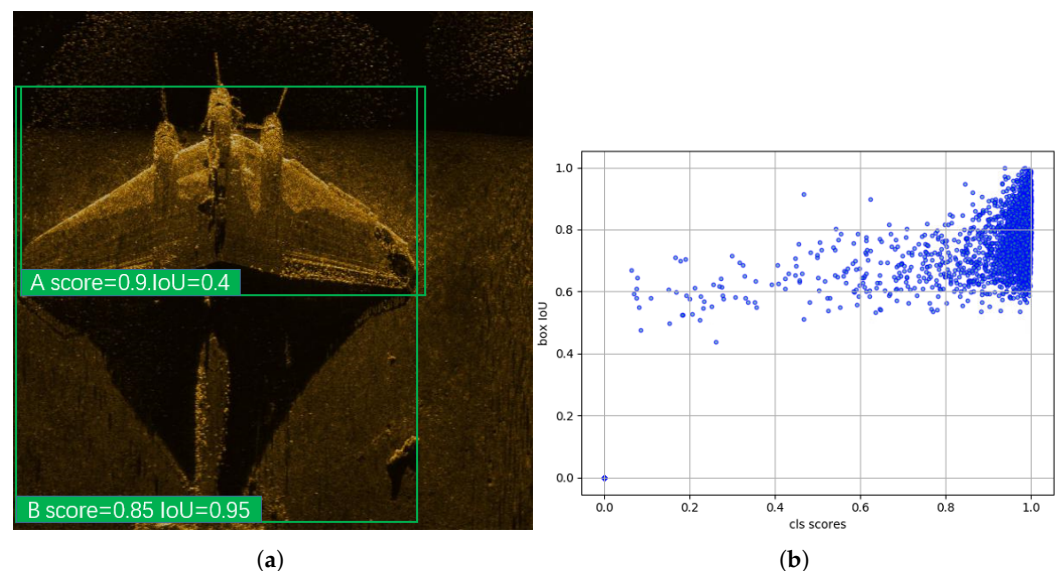
## 1. Introduction

Sonar images are generated by a sonar system, which uses sound waves as an information carrier to achieve a good signal propagation effect. Instance segmentation tasks for sonar images aim at detecting and segmenting each instance in a sonar image, which can facilitate underwater target recognition effectively; examples are searching for corpses, shipwrecks, plane wrecks, etc. However, due to the inherent limitations of sonar equipment and the disturbance of external environment, such as reverberation and environmental noise, sonar images are of poor quality due to blur edges and various types of noises. Therefore, instance segmentation tasks for sonar images are a great challenge. Classical methods for image instance segmentation, which are mostly based on the gray-scale value [1], spatial information, or edge information of an image [2], often obtain unsatisfactory results due to the inherent flaws of sonar images.

In recent years, convolutional neural networks (CNNs) have propelled the applications of deep learning methods in the computer vision field. A large number of excellent works have emerged in classification [3,4], object detection [5–8], semantic segmentation [9–13], etc. The instance segmentation task includes two sub-tasks: object detection and semantic segmentation. Therefore, current instance segmentation methods are mainly based on

detection or segmentation. Detection-based methods detect objects first and generate masks accordingly. This kind of method has excellent options: two-stage and one-stage detectors. The two-stage model Mask R-CNN [14] is built on the top of Faster R-CNN [6] by adding a mask branch to generate object masks. The mask branch is a fully convolutional network [10] which is used to generate a mask for each region of interest (RoI). Recently, the one-stage models have made great progress as well, reaching high inference speed but with somewhat diminished accuracy. YOLACT [15] breaks instance segmentation into two parallel tasks, one to generate a set of prototype masks and the other to predict mask coefficients for each instance. It can achieve a comparable trade-off between speed and accuracy. Segmentation-based methods identify each instance from the aspect of pixels by classifying each pixel first and then identifying instances accordingly [16–18].

Classification confidence is not well correlated with bounding box quality in these deep learning-based techniques. The classification branch and bounding box regression branch only care of their own tasks and are not sensitive to parallel tasks. The mismatch causes some bounding boxes with higher confidence to suppress those with lower confidence, despite them having higher IoU with ground truth in the NMS process. This will degrade the quality of the predicted bounding boxes and eventually lead to declining segmentation and detection accuracy. Figure 1a shows that a bounding box with high classification confidence has a low IoU with ground truth. To deal with the problem, in this paper, we propose a novel two-stage instance segmentation model with a boxIoU head called Mask-Box Scoring R-CNN, which uses the results of the classification branch to modify the bounding box regression branch. The boxIoU head is a tiny convolutional network and is parallel to the R-CNN head. It evaluates each predicted box with a score instead of using classification confidence. We then redesigned the pipeline of post-processing stage. The predicted boxes with low confidences of R-CNN head are no longer suppressed by NMS but filtered using an extremely low threshold (e.g., 0.005). Therefore, boxes with low confidence but high IoU can be preserved for further calibration.



**Figure 1.** (a) Mismatch between classification confidence and bounding box quality. The positioning accuracy of B is obviously higher, but it will be suppressed by A in a NMS process; (b) boxIoU vs. classification scores in a sonar image dataset.

Moreover, deep learning-based methods need huge amounts of labeled data for supervised learning. However, labeling a mask at the pixel-level for each instance in sonar images is complicated and time-consuming work. Active learning provides a solution to this problem, as its fundamental idea is to use fewer labeled samples to obtain better performance [19]. The value of a sample can be measured through strategies such as finding

the amount of information or representativeness according to different learning scenarios, and the corresponding active learning methods are the informativeness measure-based active learning (IBAL) method and the representativeness measure-based active learning (RBAL) method. Nevertheless, due to the complexity of instance segmentation tasks, active learning related to segmentation mostly focuses on the instance-agnostic semantic segmentation [20–23], and there is no specific active learning method that is suitable for instance-aware segmentation. Thus, in this paper, we propose a triplets-measure-based active learning (TBAL) method that is applicable to instance segmentation and present a balanced sampling method to mine valuable batches of samples. Using these high-value samples along with the TBAL method to train models can effectively improve the learning efficiency of the model and achieve the purpose of reducing the cost of dataset labeling.

Finally, we combine M-B Scoring R-CNN, TBAL, and balanced sampling to cope with both problems in a unified framework and present a deep active instance segmentation method.

This paper is organized as follows. In Section 2, we review the related works, including Mask Scoring R-CNN and two active learning methods. In Sections 3.1 and 3.2, we describe the detailed content of our proposed instance segmentation model and active learning method. The experimental results of the Mask-Box Scoring R-CNN and TBAL method are reported in Section 4, along with comparisons to the related works. Finally, we conclude our work in Section 5.

## 2. Related Work

### 2.1. Mask Scoring R-CNN

Mask Scoring R-CNN [24] is a state-of-the-art instance segmentation model that was built on the Mask R-CNN framework recently. Mask R-CNN uses a region proposal network to generate RoIs and extracts the feature vectors of RoIs by mapping them to the original feature map. It outputs the segmentation and detection results through three parallel heads. Mask Scoring R-CNN modifies the misalignment in Mask R-CNN between mask quality and classification score caused by the inappropriate measurement of mask quality using the classification score. The author assumes that the mask quality should be positively correlated with the classification score, yet in most cases the mask quality of an instance is poor despite its high classification score. To address the problem, Mask Scoring R-CNN uses a maskIoU head to regress the IoU between the predicted mask and corresponding ground truth mask and then predict mask score, which is denoted as  $s_{mask}$ . The predicted mask score is calculated by multiplying classification score and mask IoU,  $s_{mask} = s_{cls} \cdot s_{iou}$ , where  $s_{cls}$  is the classification score and  $s_{iou}$  is the predicted mask IoU regressed by the maskIoU head. The maskIoU head takes the predicted mask and features from RoIAlign layer as input and feeds the concatenation into four convolution layers, followed by four fully connected layers to regress the mask IoU. The regression process is applied only to the ground truth class. In this manner, the predicted mask score can consider the influence of both classification and mask, leading to a better correlation between them.

### 2.2. Active Learning

Active learning is a sub-field of machine learning which aims to lower the cost of annotation work and solve the problem of incomplete supervision. In the sampling process of active learning, informativeness and representativeness are two main query criteria [25]. The former is used to measure the effectiveness of an unlabeled sample on reducing the uncertainty of the model and always selects a sample with a higher degree of uncertainty. The latter measures the degree to which the sample can represent the structure of the input pattern and chooses samples with clustering algorithms.

Uncertainty sampling [26] is the most commonly used informativeness method. Entropy [27] is a classical sampling strategy belonging to this category. The traditional entropy-based active learning method can be written as

$$x_H^* = \operatorname{argmin}_x \sum_{i=1}^m P_\theta(y_i|x) \log P_\theta(y_i|x), \quad (1)$$

where  $y_i$  is the label of  $i$ th sample,  $P_\theta$  is the probability of  $x$  being predicted as  $y_i$  under the condition of parameter  $\theta$ , and  $m$  is the class number. It selects the sample with the maximum entropy value to query. In two-stage instance segmentation models, the softmax probabilities after the fully connected layers of the R-CNN head are used as classification probabilities.

Another commonly used query strategy of the informativeness criterion is the expected model change, which queries samples that would contribute the most to the model. Expected gradient length (EGL) [28] is a general method in this category. The EGL method can be applied in the training processes of models trained with gradient descent and select samples that maximize the gradient of the objective function. Let  $\nabla l_\theta(L)$  be the gradient of the objective function  $l$  with respect to the model parameter  $\theta$  and  $\nabla l_\theta(L \cup \langle x, y \rangle)$  be the new gradient after training sample  $(x, y)$  is appended to  $L$ . Then, the norm for new sample is

$$x_{EGL}^* = \|\nabla l_\theta(L \cup \langle x, y \rangle)\|, \quad (2)$$

where  $\|\cdot\|$  represents the second normal form of the gradient, that is, the length of the gradient.

The representativeness method mines the internal structure of the sample data using clustering to ensure the same distribution of the selected queried sample as that of the unlabeled sample. Nguyen et al. [29] clustered the samples into a tree structure and pruned the tree according to the label to retain the most representative samples. Liu et al. [30] applied the clustering method to an active learning strategy and selected representative samples of each cluster to train the model.

### 3. Materials and Methods

#### 3.1. M-B Scoring R-CNN

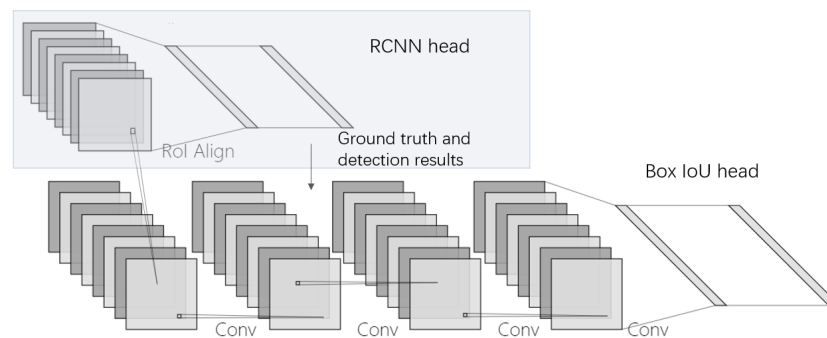
##### 3.1.1. Motivation

We know that instance segmentation is divided into two sub-tasks: the localization task and semantic segmentation within the object bounding box. For segmentation task, as mentioned above, Mask Scoring R-CNN [24] utilizes a maskIoU head to predict pixel-level IoU between the predicted mask and its matched ground truth mask to improve the quality of predicted masks. However, for the localization task, we use metric positioning accuracy using boxIoU( $AP_{box}$ ), when we use the classification confidence as the NMS score. The classification confidence and positioning accuracy are misaligned. Figure 1b shows the visualized results of the distributions of boxIoU and classification scores in our sonar image dataset. It can be observed in the figure that there is no obvious correspondence between the classification scores and the quality of the bounding boxes. The NMS flow only involves classification confidences but ignores the true quality of bounding boxes.

It is rare to use the classification score, maskIoU, or boxIoU as the direct optimization object in one two-stage framework. Based on this, we propose a novel instance segmentation model called Mask-Box Scoring R-CNN (M-B Scoring R-CNN). Our model can predict classification confidence, maskIoU, and boxIoU, denoted as class score, mask score, and box score, for each instance. We use the product of the box score and the classification score as the new NMS score so that bounding boxes with high boxIoU are preserved in NMS flow.

### 3.1.2. boxIoU Head

The boxIoU head is a convolutional neural network parallel with the R-CNN head to predict boxIoU as box score in every bounding box. It is placed after the RoI Align [14] layer to guarantee the same input features as those of the R-CNN head. In this way, (a) the task of the boxIoU head is tightly related to detection: they have the same input, which means the boxIoU head has strong relevance to the R-CNN head; (b) the boxIoU head shares the pooling layer with the R-CNN head so that the parameters can be reduced. Figure 2 shows the structure of the boxIoU head. Concretely, the proposed boxIoU head is composed of four convolution layers and two fully connected layers to control the number of parameters within an acceptable range. It takes the  $7 \times 7 \times 256$  features after RoI Align as input. The kernel size and filter numbers of the convolutional layers are set to 3 and 256 respectively, and the outputs of both fully connected layers are set to 1024.



**Figure 2.** The boxIoU head uses four convolutional layers with kernel size = 3 followed by two fully connected layers. The feature size is  $7 \times 7 \times 256$  after RoI Align.

### 3.1.3. Training Loss

In the training flow, the boxIoU head takes the proposals of the RoI Align layer, the regression results of the bounding box branch in the R-CNN head, and their corresponding ground truth as input. The target is set as the IoU between the predicted result of the bounding box regression branch and the ground truth. Proposals with IoU greater than 0.5 are chosen as the training tensors, which is the same as in the R-CNN head and mask head. The boxIoU head is optimized with  $l_2$  loss:

$$L_{boxiou} = \sum_{i=1}^N (target - y_i)^2, \quad (3)$$

where  $y$  denotes the predicted boxIoU. The total loss of M-B Scoring R-CNN is

$$L = L_{cls} + L_{boxreg} + L_{mask} + L_{maskiou} + L_{rpn} + L_{boxiou}, \quad (4)$$

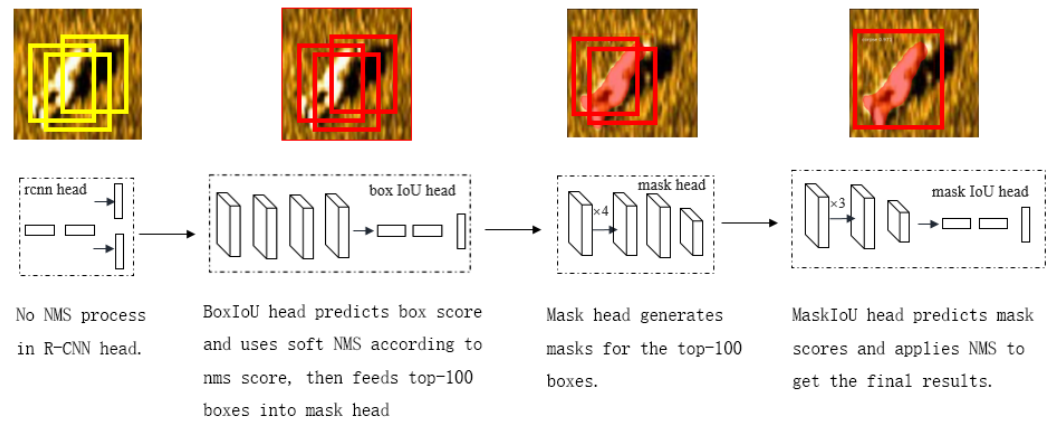
where  $L_{cls}$  is cross-entropy loss of classification,  $L_{boxreg}$  is the  $smooth_{L_1}$  loss of bounding box regression,  $L_{mask}$  is the binary cross-entropy loss of the mask head,  $L_{maskiou}$  is the  $l_2$  loss of the mask IoU head, and  $L_{rpn}$  is the loss of the RPN. We use Equation (4) to end training.

### 3.1.4. Inference

To deal with the mismatch between NMS score (we say it is cls score in a two-stage model) and bounding box quality, we use soft NMS with a new NMS score amended by box score, and we redesigned the post-processing part of M-B Scoring R-CNN for inferring. Firstly, features are fed into R-CNN head to regress the bounding boxes. In this process, the R-CNN head filters out boxes with extremely low confidence via a small threshold (i.e., 0.005) instead of applying NMS. This step reduces the computational cost of the soft-NMS later. Secondly, the bounding box regression results of the R-CNN head and



the ground truth bounding box are passed to the boxIoU head to predict the *box score* for each proposal, which is used to calibrate the predicted box. Specifically, the new *nms score* is calculated by multiplying *box score* with *class score*, and Soft-NMS [31] is applied to decrease the weights of boxes with lower scores. Thirdly,  $k$  top scoring boxes (e.g.,  $k = 100$ ) are selected to be fed into the mask head to predict a mask for each instance. Finally, maskIoU head predicts *mask score* and multiplies it with *nms score* to apply NMS. The model's outputs are bounding boxes and masks calibrated by *box score* and *mask score*. The inference process of four heads in M-B Scoring R-CNN is illustrated in Figure 3.



**Figure 3.** The inference process of M-B Scoring R-CNN. The yellow boxes show region proposals before NMS, and the red boxes show region proposals after soft NMS.

### 3.2. Active Learning with M-B Scoring R-CNN

#### 3.2.1. Triplets-Based Active Learning Method

Traditional IBAL methods mainly consider the amount of information from the perspective of classification (e.g., the entropy method and the EGL method). However, in the instance segmentation task, the predicted result is evaluated from three aspects: classification, detection, and segmentation. Therefore, it is necessary to describe the amount of sample information more comprehensively in order to achieve better performance on the whole framework.

In the inference stage of M-B Scoring R-CNN, we can obtain the classification score  $c_i^j$ , bounding box score  $b_i^j$ , and mask score  $m_i^j$  for the  $j$ th bounding box of  $i$ th sample in unlabeled sample pool  $D_u$ . The triplets  $\langle c_i^j, b_i^j, m_i^j \rangle$  can reflect the amount of information of an instance. An unlabeled instance contains more information if the three scores are low and the model is less confident about the sample. We use mean to depict the score of triplets and standard deviation to depict the difference. A higher standard deviation means that the model does not performing well in one or more aspects and the instance contains more information. The amount of information of an instance is defined as

$$I_t = 1 - \text{mean}(c_i^j, b_i^j, m_i^j) + \text{std}(c_i^j, b_i^j, m_i^j). \quad (5)$$

Suppose sample  $i$  contains  $n$  instances. Its amount of information is

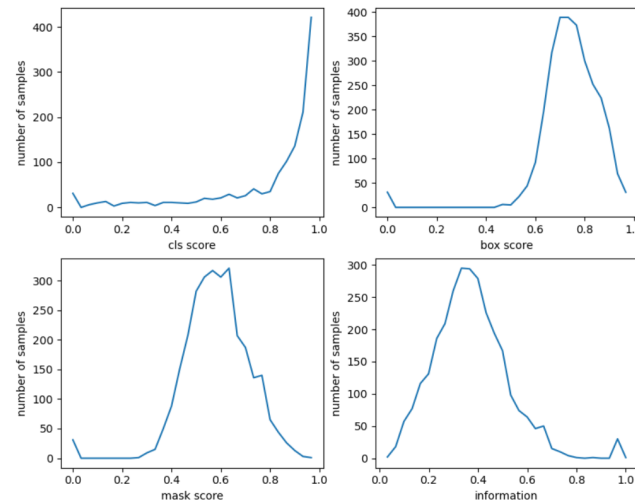
$$I_s = \sum_{k=1}^n I_t^k. \quad (6)$$

The result of the query is obtained after the calculation for each sample in  $D_u$  and sampling are completed.

#### 3.2.2. Balanced Sampling

Active learning aims to reduce the cost of sample labeling while ensuring the performance of the model, so it is necessary to select the most valuable samples. Figure 4 shows

the distributions of classification score, box score, mask score, and amount of information for all samples in unlabeled sonar image sample pool  $D_u$ . As can be seen in Figure 4, the peaks of score curves are close to the right-hand side, and the peak of the information curve is closer to the left-hand side, which indicates that the number of easy samples with higher scores is larger. Therefore, most of the training samples are easy if sampled with random sampling, in which case they will contribute less to the model's performance. Hard examples that have low scores are needed in IBAL methods. However, if the scores of the samples are extremely low, they may be outliers or very hard examples, which will degrade the performance as well. Therefore, we propose a balanced sampling method to exclude easy examples as often as possible and avoid outliers when sampling. In addition, we designed a batch query method to replace the one-by-one query in the IBAL method.



**Figure 4.** Sample scores and information distribution of the sonar image dataset.

We denote the amount of information in a sample as  $I$ ; the maximum value in the sample pool as  $I_{max}$  and the minimum value as  $I_{min}$ ; the amount of instance information falling in the interval with  $I$  as its center and the length of  $\varepsilon$ , as  $Num(I)$ . Then, the information density, denoted as  $D(I)$ , can be written as

$$D(I) = \frac{Num(I)}{l_\varepsilon(I)} = \frac{\sum_{k=1}^N \delta_\varepsilon(I_k, I)}{l_\varepsilon(I)}, \quad (7)$$

where

$$\delta_\varepsilon(x, y) = \begin{cases} 1 & , y - \frac{\varepsilon}{2} \leq x < y + \frac{\varepsilon}{2} \\ 0 & , \text{otherwise} \end{cases} \quad (8)$$

$$l_\varepsilon(I) = \min(I + \frac{\varepsilon}{2}, I_{max}) - \max(I_{min}, I - \frac{\varepsilon}{2}) \quad (9)$$

Let the number of samples in the unlabeled sample pool be  $N$ , and the weight coefficient of  $I$  can be defined as  $\beta$ :

$$\beta = \frac{N}{D(I)} = \frac{N l_\varepsilon(I)}{\sum_{k=1}^N \delta_\varepsilon(I_k, I)}. \quad (10)$$

$D(I)$  can be regarded as the number of samples in the unit interval of information.  $D(I)/N$  represents the number of samples near the amount of information  $I$ . As can be seen in Figure 4 that easy samples account for most of the samples; thus, there are more samples with small amounts of information, and  $\beta$  is smaller. Conversely,  $\beta$  is larger in the high-information area. We use  $\beta$  as the weight coefficient of balanced sampling and then apply the roulette algorithm to sample the re-weighted samples.

### 3.2.3. SIFT and Sparse Max Pooling Encoding

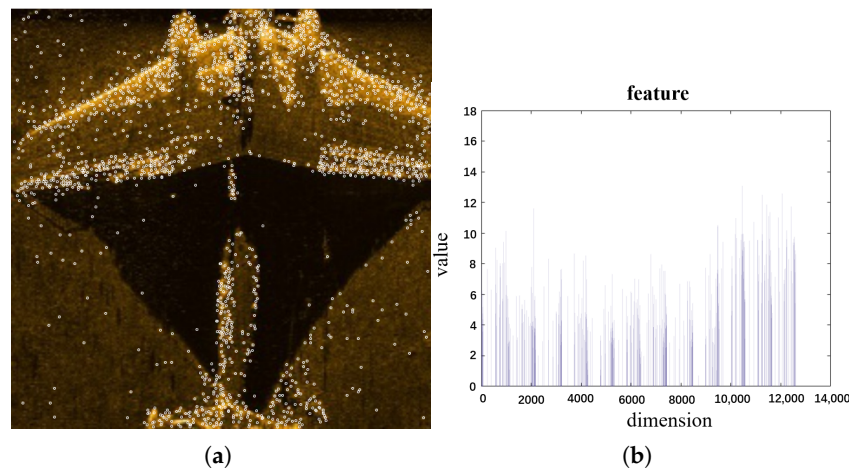
The RBAL method can select the most representative sample in the sample pool via clustering the extracted features, which can improve the generalization ability and robustness of the model. In our work, we use SIFT [32] and sparse max pooling to encode features. Each image is represented by a series of sparse features that are sent to max pooling to generate final coding. The feature points are illustrated in Figure 5a, most of which are concentrated on the edge and inside of the object, and a small number fall in the background due to the effect of noise in the sonar image. Then, we run  $k$ -medoids clustering on these feature points to find the center points with strong feature description ability and encode them. Suppose there are  $|V|$  center points after clustering. A feature vocabulary  $V = [v_1, \dots, v_{|V|}]$  can be obtained based on these center points, where  $v$  is a feature point.

Supposing that there are  $|F|$  feature points in a sonar image and each feature point is denoted as  $f$ ; then,  $F = \{f_i\}_{i=1}^{|F|}$ . For each  $f_i$ , it can be sparsely coded through  $|V|$ —that is,  $f_i = s_i V$ . The sparse coding of image features can be obtained after max pooling  $s_i$ :

$$\phi(I) = [\phi^1, \dots, \phi^{|V|}], \quad (11)$$

$$\phi^j = \max(s_i(j)), \quad (12)$$

where  $s_i(j)$  is the sparse coding with respect to  $f_j$ . In this paper,  $|V|$  is 12,600; that is,  $\phi^j$  is a 12,600-dimensional vector. Figure 5b shows the coding result.



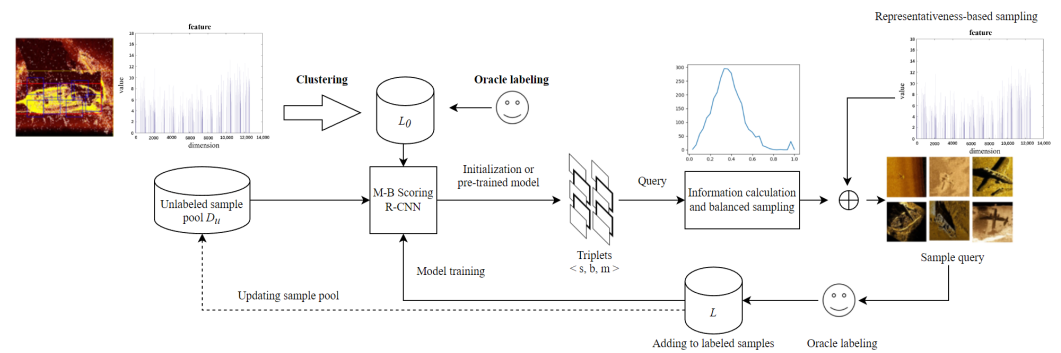
**Figure 5.** (a) Feature points extracted with SIFT; (b) sparse coding result.

The final representative sampling result needs to be selected according to the feature description. Suppose the samples are clustered into  $K$  clusters via  $k$ -medoids clustering, and then  $N/K$  samples are selected from each cluster if  $N$  initial training samples are needed.

### 3.2.4. Deep Active Learning Framework

To deal with the problems of instance segmentation for sonar image in a unified manner, we propose a deep active learning framework for the task of instance segmentation for sonar images, called Active Mask-Box Scoring R-CNN, as is shown in Figure 6. It uses the proposed M-B Scoring R-CNN as the model and applies representativeness and informativeness measure-based methods in different stages.





**Figure 6.** Active Mask-Box Scoring R-CNN.

The pipeline of our proposed framework is as follows. The representativeness-based sampling method is firstly used to choose initial samples to initialize M-B Scoring R-CNN. In the  $(t + 1)$ th iteration, the model trained in the  $t$ th iteration is used to predict triplets  $\langle s, b, m \rangle$  for each sample  $i$  in unlabeled sample pool  $D_u$ . The amount of information can be calculated from the triplets. Meanwhile, the remaining unlabeled samples in  $D_u$  are clustered. Samples closest to the cluster center are selected as the representative sampling results and added to samples to be queried. Suppose there are  $N$  samples needing to be queried in each iteration and the balance factor is  $k$ . Then,  $N * k$  samples are queried according to the IBAL method and  $N * (1 - k)$  samples according to the RBAL method. These samples are appended to the labeled training set to train M-B Scoring R-CNN and removed from  $D_u$ .

## 4. Experiments and Results

### 4.1. Sonar Image Dataset

The sonar images we used were captured by side scan sonar and synthetic aperture sonar. They were collected from public photo sharing websites according to the following two principles: one is that the instances contained in the images had to be various in terms of size, illumination, and position; the other is that there had to be no obvious bias to specific features, and the instance features of each category had to be evenly distributed in the feature space. The images in this paper were gathered with different sonar equipment in different territorial waters at different times to ensure the diversity of data. These sonar images contain three categories: corpse, shipwreck, and plane wreck. The dataset was augmented by random cropping and rotating. Finally, 4320 sonar image samples were obtained, each of which was resized to  $600 \times 600$  pixels. We followed the COCO [33] standard to label these samples, each instance with a category, a bounding box, and a mask. These samples were split into a training set and a test set. There were no intersections exist between the training set and test set. The detailed information is shown in Table 1.

**Table 1.** Sonar image dataset information.

Dataset Type	Corpse	Shipwreck	Planewreck
Training Set	1014	1078	1066
Test Set	376	386	400

### 4.2. Training Details

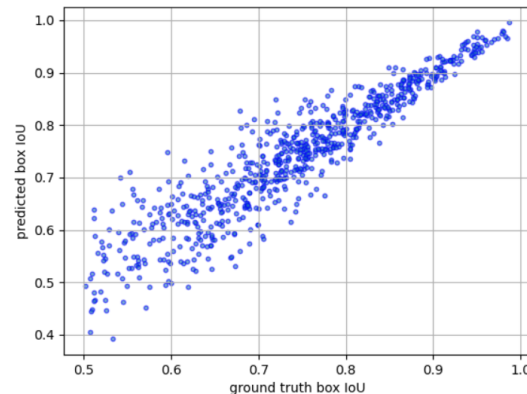
Our proposed M-B Scoring R-CNN was implemented with PyTorch 1.1. The model was initialized with a pretrained model from ImageNet, and we fine-tuned it on the aforementioned 3158 training images. We used ResNet-50 or ResNet-101 as the backbone. The model was optimized using SGD with a momentum of 0.9 and a weight decay 0.0001. The entire model was trained on a computer running Ubuntu 18.04 with Cuda10.0 and cuDNN7.6 with a NVIDIA GeForce RTX 2080.

We use COCO evaluation average precision (AP) [33] to report the results. AP forms different strictness calculation standards by setting different IoU thresholds, which were from 0.5 to 0.95 with an interval of 0.05. In our experiments, we used AP, AP50, and AP75. AP50 (or AP75) means using an IoU threshold 0.5 (or 0.75) to identify whether a predicted bounding box or mask is positive in the evaluation, and the AP is the average of 10 APs with different thresholds.

#### 4.3. M-B Scoring R-CNN Results

We used Mask R-CNN [14] and Mask Scoring R-CNN [24] as the baseline models. The comparison results are shown in Table 2, where boxAP refers to the bounding box average precision and segAP refers to the segmentation average precision. All instantiations of M-B Scoring R-CNN and Mask Scoring R-CNN outperformed Mask R-CNN. Compared with Mask Scoring R-CNN, M-B Scoring R-CNN obtained relatively higher detection performance, as the boxAP improved by about 1% with ResNet-50 or ResNet-101 as the backbone. In addition, M-B Scoring R-CNN obtained much higher scores for AP75 (81.7 AP with ResNet-50 as backbone and 84.4 AP with ResNet-101 as backbone), suggesting that the M-B Scoring R-CNN can predict more bounding boxes of high quality. This improvement derives from the revision of confidence score using box score in the softNMS process, which retains high-quality bounding boxes.

To further evaluate the performance of the boxIoU head on the sonar image test set, we kept predicted bounding boxes with boxIoU greater than 0.5 in the testing procedure and visualize their relations with corresponding ground truth when using ResNet-50 as backbone in Figure 7. It can be seen that the predictions of boxIoU head are highly correlated with ground truth. The predictions of boxIoU head are much more accurate when the bounding boxes have higher boxIoU.



**Figure 7.** The predictions of boxIoU head and the corresponding ground truth.

**Table 2.** Main results on our proposed sonar dataset.

Method	Backbone	boxAP	boxAP50	boxAP75	segAP	segAP50	segAP75
M-BS R-CNN	ResNet-50	0.660	0.986	0.817	0.599	0.980	0.705
MS R-CNN	ResNet-50	0.650	0.986	0.806	0.598	0.982	0.703
Mask R-CNN	ResNet-50	0.647	0.983	0.796	0.586	0.979	0.676
M-BS R-CNN	ResNet-101	0.672	0.987	0.844	0.602	0.982	0.713
MS R-CNN	ResNet-101	0.661	0.986	0.823	0.603	0.982	0.713
Mask R-CNN	ResNet-101	0.656	0.985	0.811	0.593	0.979	0.699

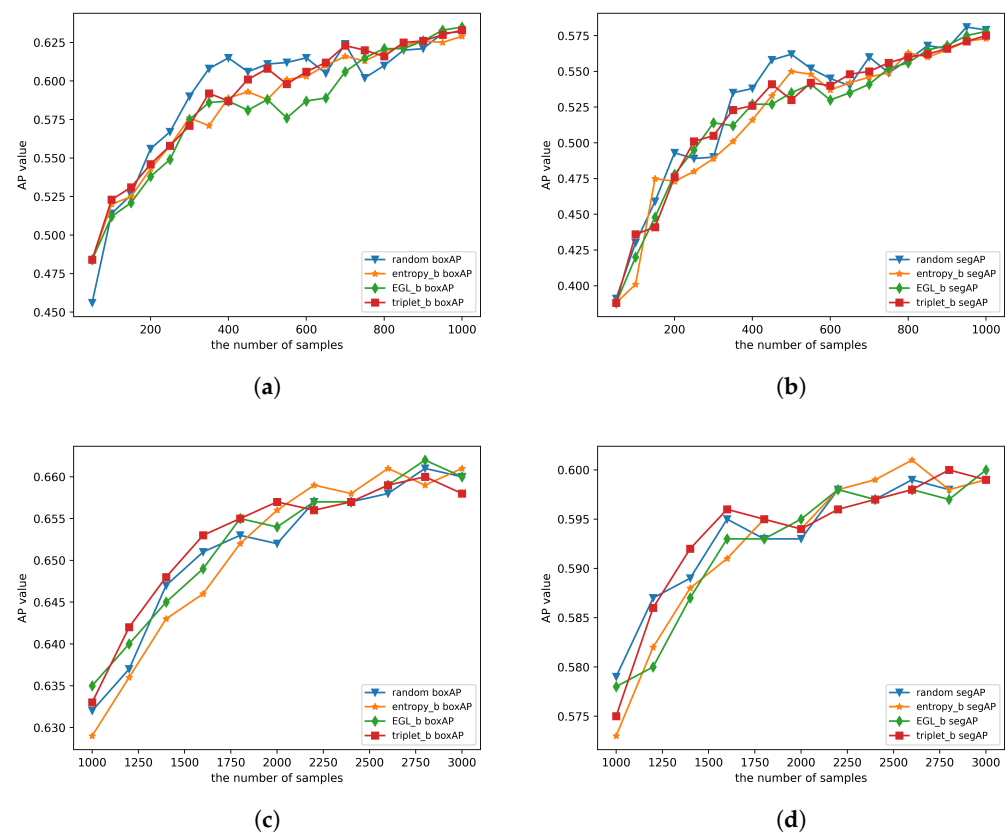
#### 4.4. TBAL and Balanced Sampling Results

The experiments in this subsection verify the effectiveness of the TBAL method and balanced sampling. We also explore the influences of different ratios of samples selected with informativeness and representativeness measure-based sampling methods. We com-

pare our proposed method with another two widely used methods: entropy and expected gradient length (EGL).

#### 4.4.1. Verifying the Effectiveness of the TBAL Method

The dataset, experiment environment, and parameters were the same as the experiment of Section 4.3. We used the M-B Scoring R-CNN to predict the triplets in the unlabeled sample pool, and queried samples based on the method of active learning. The model was initialized with 150 samples selected with representativeness-based sampling. The samples were queried in order of the amount of information. In each iteration, 50 samples were queried at a time when the number of samples was less than 1000, and 200 samples were queried when the number of samples was more than 1000. The results are shown in the Figure 8.

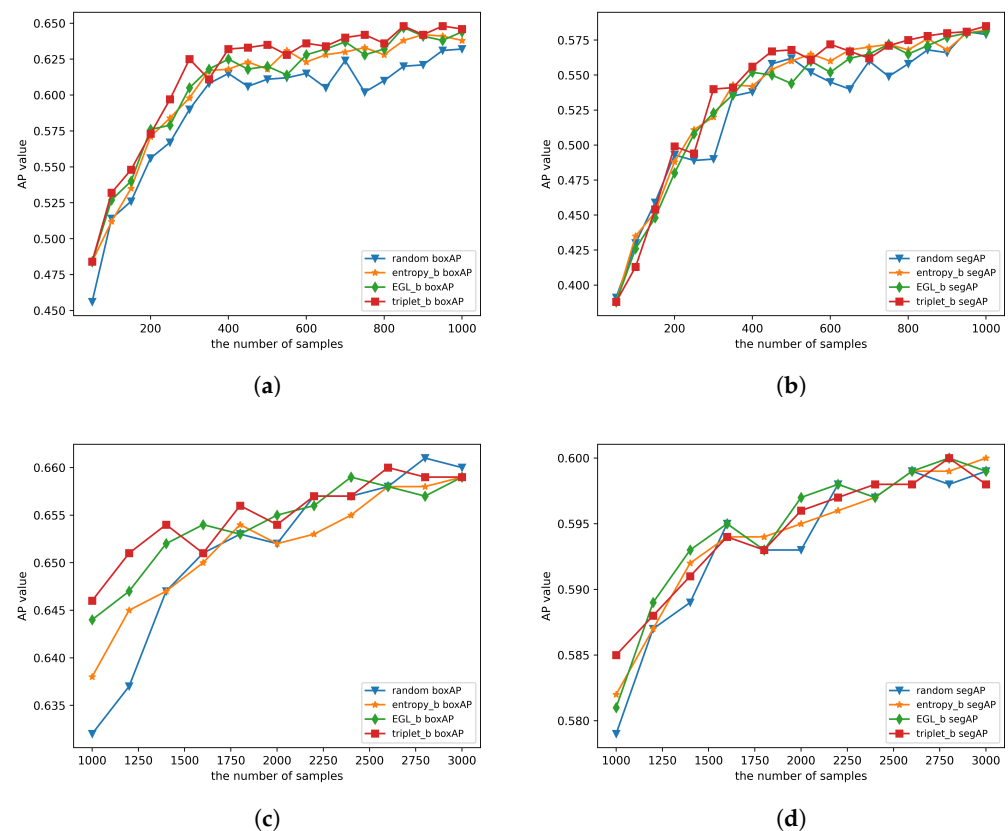


**Figure 8.** AP curve based on the amount of information, where *random*, *entropy*, *EGL*, and *triplet* denote random sampling, information entropy-based method, expected gradient length-based method, and triplets-based method, respectively. (a) BoxAP curves when the number of samples is less than 1000. (b) SegAP curves when the number of samples is less than 1000. (c) BoxAP curves when the number of samples is more than 1000. (d) SegAP curves when the number of samples is more than 1000.

When the sample size is less than 800, the AP curves of these methods fluctuate greatly. By contrast, the AP curves of the four methods become smoother as the number of samples gets large. Notably, random sampling outperforms the other methods slightly with fewer samples. We speculate that this is due to the fact that sorting samples according to the amount of information and sampling in descending order will incur a decline in generalization ability. Samples with larger amounts of information may be similar in features, in which case the model will have a preference for certain features; and samples with larger amounts of information may include outliers, which will damage the stability of the model.

#### 4.4.2. Verifying the Effectiveness of the Balanced Sampling Algorithm

We then embedded balanced sampling into the above experiments. The results are shown in Figure 9. After the IBAL method was combined with balanced sampling, the performance was obviously improved. When the number of samples was less than 1000, the three IBAL methods showed significant improvement under boxAP and segAP and were stabler, which shows that balanced sampling can alleviate the problems of selecting samples according to the value of the amount of information. This is because balanced sampling no longer focuses on the samples with the largest amounts of information, but chooses samples with different levels of information with different probabilities. Notably, our TBAL method outperformed the other methods when all of them were combined with balanced sampling.



**Figure 9.** AP curve based on the amount of information and balanced sampling. The suffix ‘\_b’ represents using balanced sampling. (a) BoxAP curves when the number of samples is less than 1000. (b) SegAP curves when the number of samples is less than 1000. (c) BoxAP curves when the number of samples is more than 1000. (d) SegAP curves when the number of samples is more than 1000.

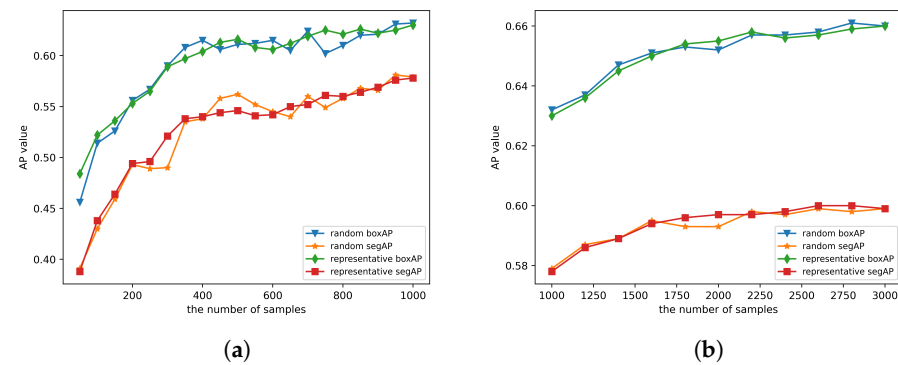
#### 4.4.3. Performance with Different Balance Factors

A common practice for an active learning method is to combine IBAL with RBAL methods, and we followed this practice as well. We used balance factor  $k$  ( $0 \leq k \leq 1$ ) to adjust the ratio of samples selected with different criteria, and  $k = 1$  means not using the RBAL method, whereas  $k = 0$  means not using the IBAL method.

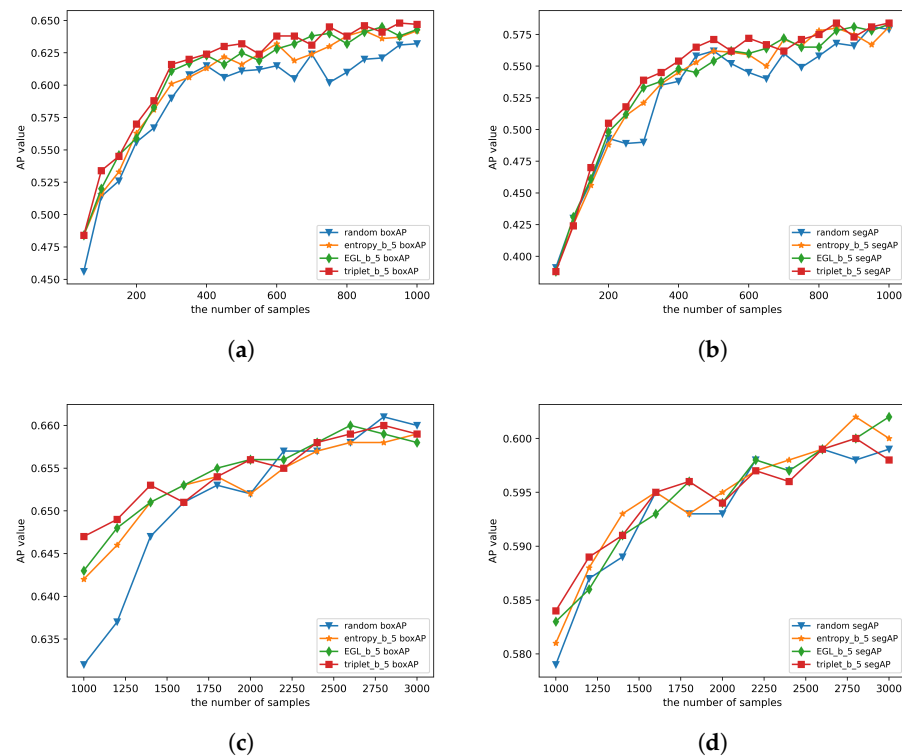
We firstly set  $k$  to 0 to evaluate the performance of the RBAL method, as is shown in Figure 10. When the number of samples was less than 400, the RBAL method slightly outperformed random sampling and is more stable. However, the overall performance of the RBAL method showed no significant advantage. This is due to it being unable to select the most valuable samples, which can improve the performance of models, even though it can make full use of the data structure of all the samples.

For the balance factor  $k$  being 0.5, the results are shown in Figure 11. The performances of the three active learning methods were better than that of random sampling, and the TBAL method was slightly better than other methods. In addition, compared with Figure 9, adding  $0.5 * N$  samples sampled with the RBAL method did not yield an obvious improvement. We assume that the balanced sampling already included parts of structural features which were duplicated with representativeness-based sampling.

We increased the value of  $k$  to 0.8, and the results are shown in Figure 12. Compared with  $k = 0.5$ , the model performed better under a small number of samples when  $k = 0.8$ , suggesting that when the value of  $k$  is larger, that is, the sampling ratio based on the amount of information is higher, the improvement is more noticeable.

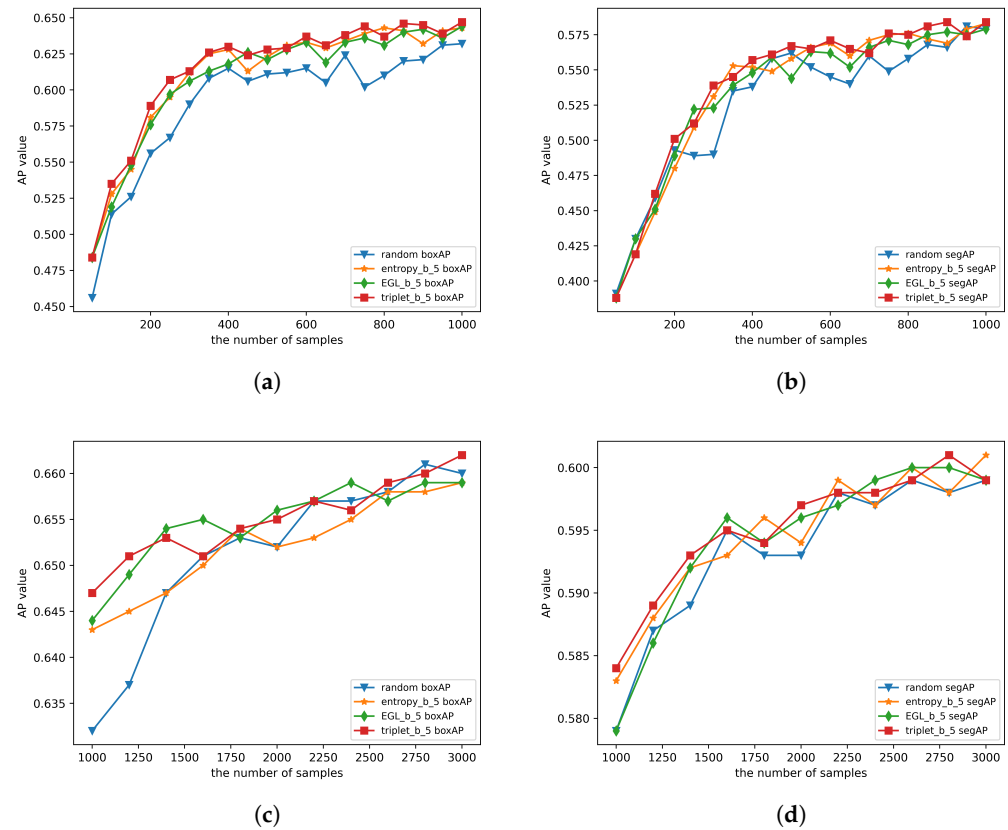


**Figure 10.** AP curve of RBAL method. (a) BoxAP and segAP curves when the number of samples is less than 1000. (b) BoxAP and segAP curves when the number of samples is more than 1000.



**Figure 11.** The AP curves comparing random sampling with the three sampling algorithms, all of which combine the informativeness-based method with the representativeness-based algorithm. Suffix ‘\_b’ represents for using balanced sampling, and suffix ‘\_5’ represents the balance factor  $k = 0.5$ . (a) BoxAP curves when the number of samples is less than 1000. (b) SegAP curves when the number of samples is less than 1000. (c) BoxAP curves when the number of samples is more than 1000. (d) SegAP curves when the number of samples is more than 1000.





**Figure 12.** The AP curves comparing random sampling with the three sampling algorithms which combine the informativeness-based method with the representativeness-based algorithm. Suffix ‘\_b’ represents using balanced sampling, and suffix ‘\_8’ represents the balance factor  $k$  being 0.8. (a) BoxAP curves when the number of samples is less than 1000. (b) SegAP curves when the number of samples is less than 1000. (c) BoxAP curves when the number of samples is more than 1000. (d) SegAP curves when the number of samples is more than 1000.

## 5. Conclusions

In this paper, we proposed a two-stage instance segmentation model called Mask-Box Scoring R-CNN which outperforms Mask R-CNN in both detection and segmentation. The proposed boxIoU head corrects the classification score in the NMS process and obtains better positioning accuracy than Mask Scoring R-CNN. Meanwhile, we proposed a triplets-measure-based active learning method to evaluate samples more comprehensively, and a balanced sampling method was designed to select samples that are neither too simple nor extremely hard for training. Finally, we presented a deep active learning framework for sonar image instance segmentation called Active Mask-Box Scoring R-CNN. Experiments showed that our methods perform well on the sonar image dataset. In the future, we plan to explore one-stage instance segmentation models in this work.

**Author Contributions:** Conceptualization, F.X. and L.J.; methodology, F.X.; software, F.X.; validation, J.H. and J.W.; writing—original draft preparation, J.H. and J.W.; writing—review and editing, F.X., L.J., J.H. and J.W.; visualization, J.H. and J.W.; supervision, L.J.; funding acquisition, L.J. All authors have read and agreed to the published version of the manuscript.

**Funding:** The Project was Supported by the National Natural Science Foundation of China (No: 61871124 and 61876037), by the fund of China ship development and design center (No: JJ-2021-702-05), by the fund of national key Laboratory of science and technology on underwater acoustic antagonizing (No: 2021-JCJQ-LB-033-09).

**Data Availability Statement:** The data presented in this study are available on request from the corresponding author. The data are not publicly available now due to deficient maintenance capacity.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

- Otsu, N. A threshold selection method from gray-level histograms. *IEEE Trans. Syst. Man Cybern.* **1979**, *9*, 62–66. [\[CrossRef\]](#)
- Torre, V.; Poggio, T.A. On edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **1986**, *8*, 147–163. [\[CrossRef\]](#) [\[PubMed\]](#)
- Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [\[CrossRef\]](#)
- He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
- Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
- Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 91–99. [\[CrossRef\]](#) [\[PubMed\]](#)
- Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 21–37.
- Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
- Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv* **2014**, arXiv:1412.7062.
- Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
- Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 834–848. [\[CrossRef\]](#)
- Mozaffari, M.H.; Lee, W.S. Semantic Segmentation with Peripheral Vision. In Proceedings of the International Symposium on Visual Computing, San Diego, CA, USA, 5–7 October 2020; Springer: Cham, Switzerland, 2020; pp. 421–429.
- Mozaffari, M.H.; Lee, W.S. Dilated convolutional neural network for Tongue Segmentation in Real-time Ultrasound Video Data. In Proceedings of the 2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Houston, TX, USA, 9–12 December 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1765–1772.
- He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
- Bolya, D.; Zhou, C.; Xiao, F.; Lee, Y.J. Yolact: Real-time instance segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27–28 October 2019; pp. 9157–9166.
- Bai, M.; Urtasun, R. Deep watershed transform for instance segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5221–5229.
- Kirillov, A.; Levinkov, E.; Andres, B.; Savchynskyy, B.; Rother, C. Instancecut: From edges to instances with multicut. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5008–5017.
- Arnab, A.; Torr, P.H. Pixelwise instance segmentation with a dynamically instantiated network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 441–450.
- Settles, B. Active Learning Literature Survey. In *Computer Sciences Technical Report 1648*; University of Wisconsin: Madison, NJ, USA, 2009.
- Gal, Y.; Islam, R.; Ghahramani, Z. Deep bayesian active learning with image data. In Proceedings of the International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; pp. 1183–1192.
- Sener, O.; Savarese, S. Active learning for convolutional neural networks: A core-set approach. *arXiv* **2017**, arXiv:1708.00489.
- Yang, L.; Zhang, Y.; Chen, J.; Zhang, S.; Chen, D.Z. Suggestive annotation: A deep active learning framework for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Quebec City, QC, Canada, 11–13 September 2017; Springer: Berlin/Heidelberg, Germany, 2017; pp. 399–407.
- Jain, S.D.; Grauman, K. Active image segmentation propagation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2864–2873.
- Huang, Z.; Huang, L.; Gong, Y.; Huang, C.; Wang, X. Mask scoring r-cnn. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 6409–6418.
- Zhao, Y.; Shi, Z.; Zhang, J.; Chen, D.; Gu, L. A novel active learning framework for classification: Using weighted rank aggregation to achieve multiple query criteria. *Pattern Recognit.* **2019**, *93*, 581–602. [\[CrossRef\]](#)

26. Lewis, D.D.; Gale, W.A. A sequential algorithm for training text classifiers. In Proceedings of the SIGIR'94, Dublin, Ireland, 3–6 July 1994; Springer: Berlin/Heidelberg, Germany, 1994; pp. 3–12.
27. Shannon, C.E. A mathematical theory of communication. *Bell Syst. Tech. J.* **1948**, *27*, 379–423. [[CrossRef](#)]
28. Settles, B.; Craven, M.; Ray, S. Multiple-instance active learning. *Adv. Neural Inf. Process. Syst.* **2007**, *20*, 1289–1296.
29. Nguyen, H.T.; Smeulders, A. Active learning using pre-clustering. In Proceedings of the Twenty-First International Conference on Machine Learning, Banff, AB, Canada, 4–8 July 2004; p. 79.
30. Liu, Y.; Wang, Y.; Sowmya, A. Batch mode active learning for object detection based on maximum mean discrepancy. In Proceedings of the 2015 International Conference on Digital Image Computing: Techniques and Applications (DICTA), Adelaide, Australia, 23–25 November 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 1–7.
31. Bodla, N.; Singh, B.; Chellappa, R.; Davis, L.S. Soft-NMS—improving object detection with one line of code. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 5561–5569.
32. Lowe, D.G. Object recognition from local scale-invariant features. In Proceedings of the Seventh IEEE International Conference on Computer Vision, Corfu, Greece, 20–27 September 1999; IEEE: Piscataway, NJ, USA, 1999; Volume 2, pp. 1150–1157.
33. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; Springer: Berlin/Heidelberg, Germany, 2014; pp. 740–755.