Journal of *Marine Science and Engineering*

MDPI

Article

YOLO-LFVM: A Lightweight UAV-Based Model for Real-Time Fishing Vessel Tracking and Dimension Measurement

Zhuofan Hui [1], Penglong Li [1,2], Shujiang Miao [1], Yinfu Li [3], Lie Shen [1,*] and Hui Shen [4]

[1] College of Navigation and Naval Engineering, Dalian Ocean University, Dalian 116023, China; huizf0909@126.com (Z.H.); 15142892813@163.com (P.L.)
[2] Key Laboratory of Fishery Remote Sensing, East China Sea Fisheries Research Institute, Chinese Academy of Fishery Sciences, Ministry of Agriculture and Rural Affairss, Shanghai 200090, China
[3] Merchant Marine College, Shanghai Maritime University, Shanghai 201306, China; liyinfu28@163.com
[4] Liaoning Provincial Marine and Fishery Administrative Law Enforcement Corps, Dalian 116023, China
* Correspondence: shenlie@dlou.edu.cn

Abstract

This study proposes a lightweight real-time fishing vessel tracking and size measurement model based on a UAV. In view of the problems faced by the current fishing port management department, such as low efficiency of fishing vessel size measurement methods and difficulty in updating the size information of large quantities of fishing vessels in time, this paper proposes a lightweight real-time fishing vessel tracking and size measurement model based on a UAV. (YOLO-LFVM). The model incorporates lightweight modules, such as MobileNetV3, AKConv, and C2f, and utilizes Python scripts in conjunction with OpenCV to measure vessel size in pixels. The findings indicate that, compared to the original model, the YOLO-LFVM model's accuracy rate, recall rate, and mAP@0.5 decrease by only 0.7%, 0.2%, and 0.3%, respectively, while mAP@0.95 increases by 1.7%. Additionally, the model's parameters decrease by 65%, and GFLOPs decrease by 69%. When comparing the model's output with actual vessel data, the average relative error for total length is 2.67%, and for width, it is 3.28%. The research shows that the YOLO-LFVM model is effective in ship identification, ship tracking statistics, and measurement. Through the integration with UAV remote sensing technology, it is conducive to the timely updating of large-scale fishing vessel size information. Finally, the model can assist the daily management and law enforcement of the fishing port management department and can be applied to other equipment with limited computing power to perform target detection and object size measurement tasks.

Keywords: object detection; YOLOv8; ship category statistics; hull size measurements; lightweight improvement

check for updates

Received: 15 August 2025
Revised: 1 September 2025
Accepted: 5 September 2025
Published: 10 September 2025

Citation: Hui, Z.; Li, P.; Miao, S.; Li, Y.; Shen, L.; Shen, H. YOLO-LFVM: A Lightweight UAV-Based Model for Real-Time Fishing Vessel Tracking and Dimension Measurement. *J. Mar. Sci. Eng.* **2025**, *13*, 1739. https://doi.org/10.3390/jmse13091739

Copyright: © 2025 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

With the continuous reduction in global fishery resources and the serious degradation of marine ecosystems, the urgency and importance of fishery management have become increasingly prominent, and a key issue that needs immediate attention has been raised. In all aspects of fishery management, the supervision and monitoring of fishing vessels is particularly critical. As a basic physical characteristic, the length of fishing vessels is an important basis for the fishery management department to classify, formulate rules, and implement restrictions on fishing vessels. However, at present, the measurement of the size of fishing vessels by the fishing port management department is largely dependent

on manual measurement, which leads to low efficiency and difficulty in updating the size information of large fishing vessels in time. Therefore, intelligent measurement and monitoring technology is urgently needed to achieve accurate measurement of the size of large-scale fishing vessels. With the development and popularization of deep learning target detection algorithms and computer measurement technology, new ideas have been brought to this research.

Object detection algorithms based on deep learning are a research hotspot in the field of computer vision, which mainly includes the two-stage object detection algorithm and single-stage object detection algorithm. Among them, the two-stage target detection algorithm first extracts the candidate box according to the image, and then performs secondary correction based on the candidate region to obtain the detection result. The detection accuracy is high, but the detection speed is slow. Common two-stage target detection algorithms include the R-CNN series [1]. The single-stage target detection algorithm regards the target detection problem as a regression analysis problem of target location and category information, and the detection results can be directly output through a neural network model. Compared with the two-stage detection algorithm, the single-stage detection algorithm does not need to generate candidate regions and can directly extract eigenvalues in the network for target classification and location, so it has higher detection speed. Common single-stage object detection algorithms include SSD [2], the YOLO series [3], and RetinaNet [4]. Among them, the YOLO series algorithm has the characteristics of fast detection speed and high accuracy, which are more suitable for the needs of this study.

At present, many domestic and foreign scholars have applied the target detection algorithm based on deep learning to the identification and tracking of ships. Fang et al. [5] used YOLOv5 to solve the problem of high false detection rate and missed detection rate of small targets in optical remote sensing images, and proposed an improved method, which significantly improved the detection accuracy of ships, especially small ships, while reducing false detection and missed detection. Sun et al. [6] introduced a SAR vessel detector bfa-YOLO based on YOLO, which uses bidirectional feature fusion and angle classification to effectively solve the challenge of fast and accurate recognition of multi-scale, arbitrary direction, and dense distributions of blood vessels in SAR images. Wang et al. [7] proposed a GT-YOLO algorithm based on infrared images to solve the problem of near-shore infrared ship detection, which effectively improved ship recognition performance at night and under inclement weather conditions.

The above work has significantly enhanced ship detection performance and practicability in the field of ship recognition. However, these studies mainly focused on ship identification rather than ship size measurement. Anjanayya et al. [8] proposed an innovative target size measurement method using OpenCV library and image processing technology, which realized efficient and accurate size estimation and provided a new solution for fishing vessel size measurement. O.E. Apolo-Apolo et al. [9] proposed an innovative method for estimating citrus fruit size, which integrates UAV remote sensing technology, FasterR-CNN, and LSTM deep learning models and image processing technology to provide accurate estimation of citrus fruit size. This method shows the great potential and application value of the combination of computer vision measurement technology and UAV remote sensing. Han et al. [10] introduced the SSMA-YOLO model, using YOLOv8-n and UAV remote sensing technology to solve the detection challenges caused by small pixel size and insufficient features of ship satellite remote sensing images. This fusion promotes the development and application of UAV remote sensing technology in the field of ship identification. However, there is still a lack of ship measurement models that combine computer vision measurement technology with UAV remote sensing technology.

Compared with traditional high-point monitoring, the UAV platform has higher mobility [11,12], which can better meet the real-time requirements of law enforcement, but it also puts forward more stringent standards for model lightweighting, which has become the core focus of model improvement. Based on an analysis of lightweight networks such as MobileNetV3 [13], AKConv [14] and RepGhost [15], we propose a real-time tracking and size measurement model, YOLO-LFVM, for light fishing vessels based on the improved YOLOv8. The goal of the model is to minimize the growth of the number of parameters while keeping the detection accuracy of the fishing vessel basically unchanged. Specifically, firstly, the MobileNetV3 backbone network is introduced to optimize parameter efficiency through neural architecture search (NAS), which greatly reduces the number of model parameters, but direct migration leads to increases in the small target missed detection rate and rotating ship angle deviation. Subsequently, AKConv dynamic convolution is integrated to enhance multi-scale feature extraction, and the calculation amount is reduced by 28% on the VisDrone-DET2021 dataset, but there are still deficiencies in the suppression of ripple interference in complex sea surface background. Finally, the RepGhost module is introduced to further compress the parameters through reparameterization technology, and combined with the unique color enhancement strategy of the sea scene, the feature identification accuracy under complex sea surface is improved, which provides a new lightweight solution for UAV deployment. Finally, the improved model is combined with UAV remote sensing technology, and the Python script is integrated with OpenCV to measure the size of the fishing vessels in a pixel-by-pixel manner. In order to make full use of the real-time advantages of UAVs, the problem of low efficiency and difficulty in updating the size information of large fishing vessels faced by the current fishing port management department must be solved.

The rest of this paper is organized as follows: Section 2, Dataset Creation, explains the necessity of a rotating frame in ship size measurement and proposes a data enhancement method without changing the BBOX for rotating frame labeling. In Section 3, Experimental Environment and Model Improvement, the structure and method of the improved model are introduced, and the effectiveness of model improvement is verified by experiments. Section 4, Ship Tracking Statistics and Size Measurement, introduces the principle of ship measurement; the tracking ability of the model in the process of UAV flight is optimized. The size measurement results and errors are analyzed. In Section 5, the influence of orient labeling on ship size measurement results and the influence of orient data enhancement on model training are discussed. The Section 6 draws a conclusion.

2. Dataset Creation

2.1. Data Sources

The data images are from Dalian Bay Fishing Port, which has more than 300 permanent fishing vessels. It is the first batch of fixed-point landing fishing ports for national marine fishing catches in China. It has industry demonstration value to carry out model application research here. The image of this dataset is obtained by unmanned aerial vehicle (UAV) photography, which is obtained by setting a fixed take-off point, a fixed flight height of 100 m, and a fixed patrol route. During the whole data acquisition process, the camera direction remains constant. The UAV model used is DJIMavic3T (SZ DJI Technology Co., Ltd., Shenzhen, China).

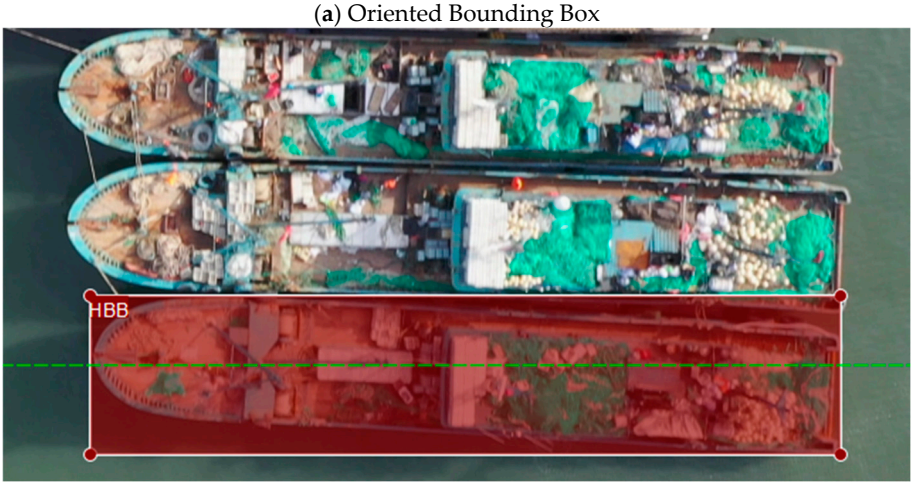
2.2. Oriented Bounding Box Annotation

This research aims to identify and quantify the dimensions of three vessel types: fishing vessels, law enforcement vessels, and tugs. The underlying principle for measuring vessel dimensions involves extrapolating them from predicted bounding box parameters.

Horizontal bounding boxes (HBBs) demonstrate poor adherence to vessel contours, thereby compromising measurement accuracy. Given the widespread adoption of Oriented Bounding Box (OBB) annotation in maritime recognition [16–18], we employed the x-Anylabeling tool for annotation. Figure 1 illustrates the comparison between horizontal and oriented bounding box annotation approaches.



(**a**) Oriented Bounding Box



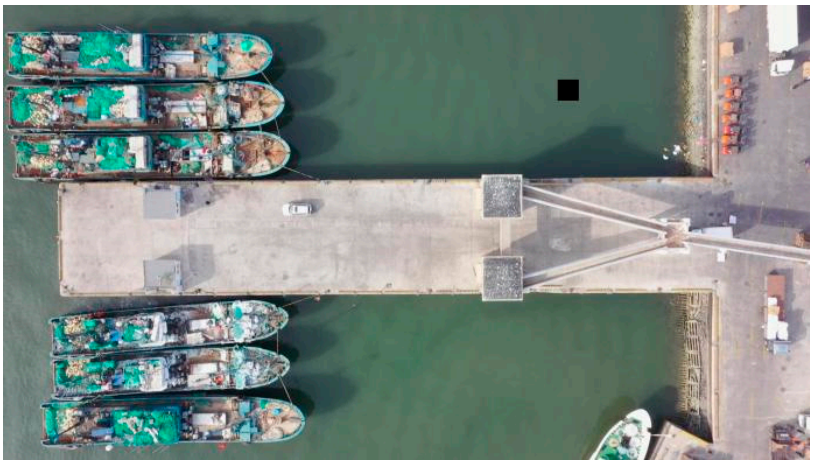
(**b**) Horizontal bounding box annotations

Figure 1. Comparison between the labeling effects of the Oriented Bounding Box and the horizontal bounding box.

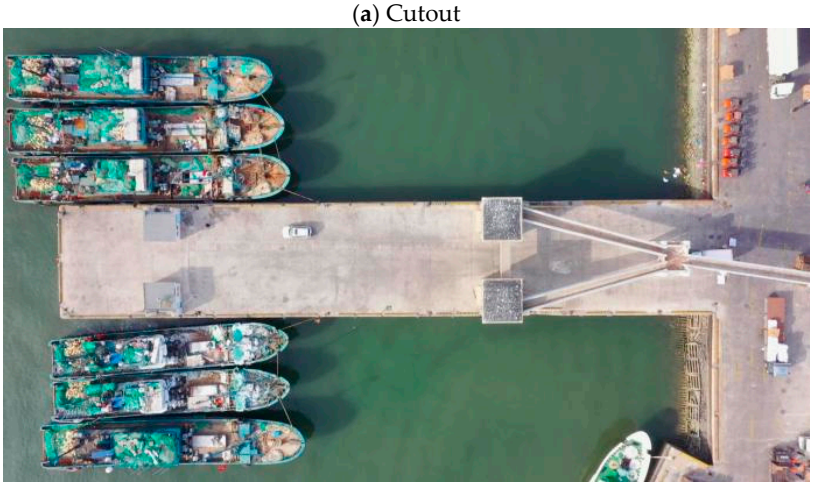
2.3. Data Augmentation for Oriented Bounding Box

Data augmentation is also a necessary technique in deep learning, which can address problems such as data scarcity and data imbalance, enhance generalization capability and robustness, and serve as a shortcut to achieving performance improvement. However, not all data augmentation methods are necessarily applicable to oriented bounding box object detection data. In the process of oriented bounding box object detection, data augmentation is used to enhance the diversity and richness of training samples, improving model generalization performance and detection performance. However, for bounding boxes (BBoxes) in oriented bounding box object detection, directly modifying the position, size, orient angle, and other parameters of a BBox will lead to certain problems. The annotation of BBoxes correctly reflects the actual position, size, and orient angle of the target object. Directly manipulating BBoxes will destroy the correspondence between the annotation and the object, which not only affects the ability of the model to learn the exact positional correspondence of the target object from the image but also increases the complexity of the post-processing steps, resulting in misjudgment results, thus affecting the measurement

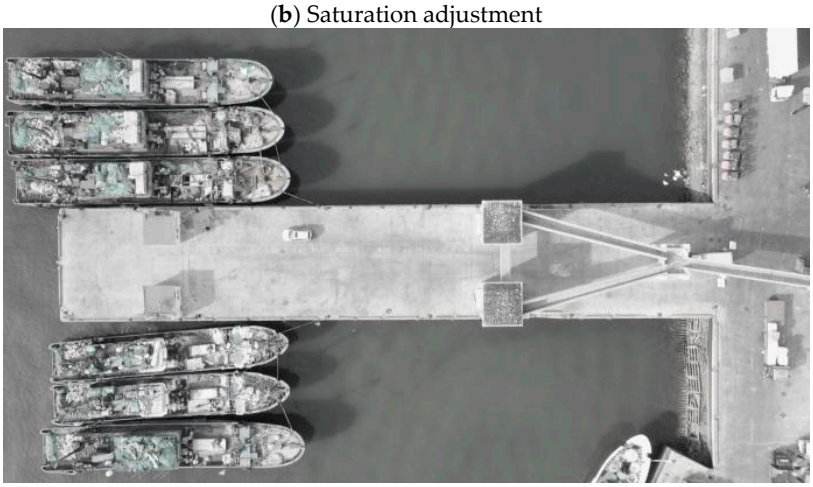
accuracy of the hull size. Therefore, this study adopts data augmentation methods that do not affect the BBox, including brightness adjustment, contrast adjustment, saturation adjustment, cutout, rain line simulation, fog effect generation, and other techniques. Figure 2 shows partial images after data augmentation.



(**a**) Cutout

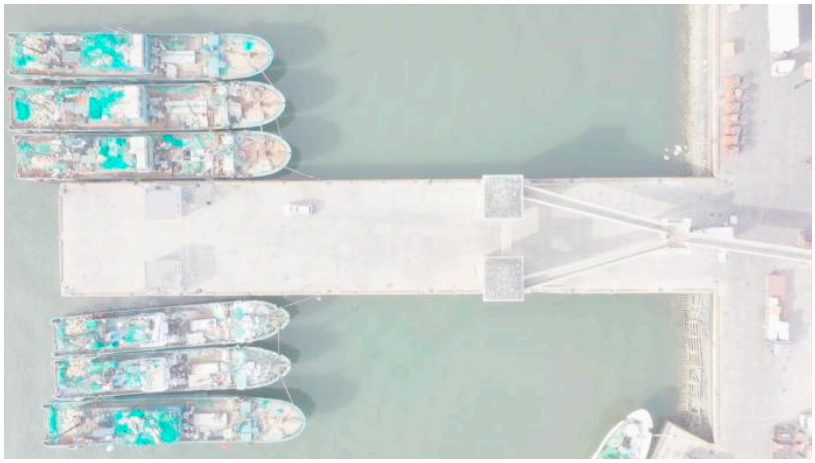


(**b**) Saturation adjustment

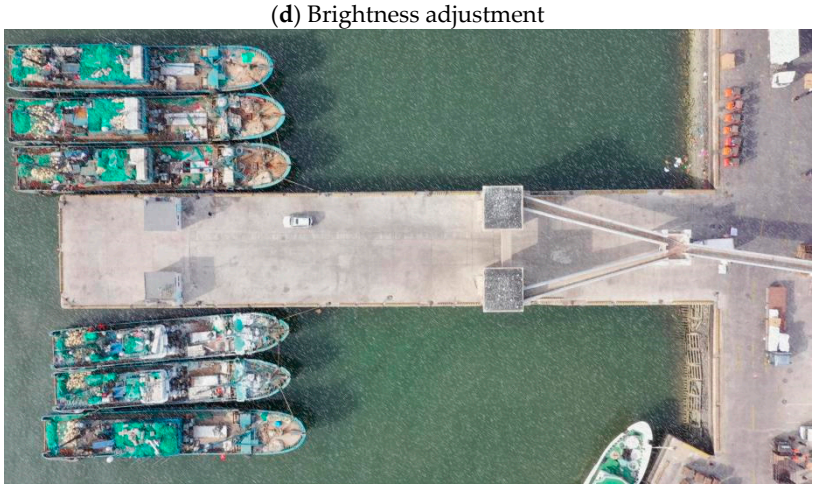


(**c**) Contrast adjustment

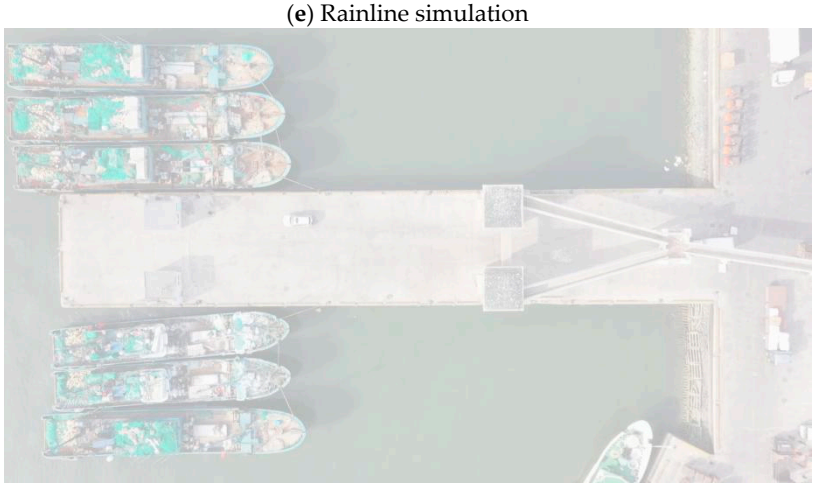
Figure 2. *Cont.*



(**d**) Brightness adjustment



(**e**) Rainline simulation



(**f**) Fog effect generation

Figure 2. Examples of effects after applying partial data augmentation.

2.4. Dataset Splitting

In this study, the data enhancement method of non-interference boundary frames is adopted, including brightness adjustment, contrast adjustment, saturation adjustment, cutout, rain line simulation, and fog effect generation. After data enhancement and expansion, the total dataset reached 5200 images, which were divided into a training set (4160 images), validation set (520 images), and test set (520 images) according to a ratio of 8:1:1. The annotation statistics show that the training set contains 17,496 fishing vessel annotation boxes, 3664 law enforcement vessel annotation boxes, and 784 tug annotation

boxes. Table 1 quantitatively shows the distribution characteristics of three types of targets (fishing vessels, law enforcement vessels, and tugs) in each subset.

Table 1. Statistics on the number of targets for different types of vessels.

Vessel Category	Training Set	Validation Set	Test Set	Total
Fishing vessel	17,496	2187	2187	21,870
Law enforcement vessel	3664	458	458	4580
Tug	784	98	98	980

3. Experimental Environment and Model Improvements

3.1. Training Environment and Parameter Settings

The experiments were conducted on a Windows 11 operating system using Python 3.8.18 as the programming language and PyTorch 2.1.1 as the deep learning framework. The CPU configuration was a 13th Gen Intel(R) Core (TM) i5-13450HX with a clock speed of 2.40 GHz. The CUDA version used was CUDA 12.2, and the GPU was an NVIDIA 4050 (Santa Clara, CA, USA). The model training was conducted with the following hyperparameter configuration: an initial learning rate of 0.001, a batch size of 16, and 100 training epochs.

3.2. Model Evaluation Criteria

The following indicators were used to evaluate the performance of the model: precision (P), recall (R), mean average precision (mAP@0.5), mean average precision (mAP@0.95), Params, and GFLOPS.

(1) Precision (P) represents the ratio of the number of targets correctly predicted by the model to the total number of predictions.

$$P = \frac{TP}{TP + FP} \tag{1}$$

A true positive is a positive sample that the model correctly predicts. A false positive is a positive sample that the model incorrectly predicts. The higher the P value, the lower the false detection rate.

(2) Recall (R) reflects the model ‘s ability to cover the actual positive samples.

$$R = \frac{TP}{TP + FN} \tag{2}$$

A false negative is a positive sample that the model misses. The higher the recall rate, the more comprehensively the model can detect the actual target.

(3) The detection performance of the mean average accuracy (mAP@0.5) at the intersection-over-union threshold loU = 0.5 is used to measure the accuracy of target positioning. The higher the mAP@0.5value is, the more accurate the bounding box prediction is.

(4) The performance of the mean absolute error (mAP@0.95) under the IoU threshold of 0.95 is used to evaluate the robustness of the model under harsh conditions. The higher the mAP@0.95 value, the stronger the cross-scenario stability.

(5) Params represent the sum of all trainable parameters of the model. The smaller the Params, the lighter and easier it is to deploy the model.

(6) GFLOPS (floating-point operations) are the floating-point operations required for single-image reasoning. The smaller the GFLOPS, the higher the computational efficiency of the model.

3.3. YOLOv8-OBB Network Model

3.3.1. YOLOv8-OBB Network Model

YOLOv8-OBB, as an upgraded version of the YOLO algorithm, is primarily designed for the detection of arbitrarily oriented objects. It exhibits significant advantages in terms of model architecture design, oriented bounding box (OBB) detection, and loss functions. The model comprises two main components: the backbone and the head. The backbone, which includes C2f and SPPF modules, leverages multiple convolutional layers combined with downsampling to extract feature information from images. The head incorporates Upsample, Concat, and C2f modules to fuse features from different hierarchical levels, ultimately predicting oriented bounding boxes through the OBB mechanism. YOLOv8-OBB exhibits robust performance in detecting objects at arbitrary orientations, accurately localizing target objects within images, and generating bounding boxes that minimize background region inclusion, thereby reducing the impact of background on object classification. The model employs three primary loss functions: Classification Loss (CLS), Bounding Box Loss (BOX), and Distribution Focal Loss (DFL), which quantify the classification loss, bounding box localization loss, and the distribution of objects within predicted bounding boxes, respectively.

The development of YOLOv8-OBB extends the capabilities of the YOLO algorithm to the detection of arbitrarily oriented targets, thereby broadening its application scope. It demonstrates significant potential for various applications, including aerial imagery analysis and text detection. By enabling the detection of oriented bounding boxes, YOLOv8-OBB facilitates more detailed object detection in images, reducing the influence of background on target classification and achieving more precise and accurate localization of objects. Moreover, the detection of oriented bounding boxes substantially reduces the impact of background on image target recognition.

3.3.2. Theoretical Innovation of YOLOv8-OBB

The theoretical innovation of YOLOv8-OBB lies in its deep integration of the geometric prior knowledge and dynamic feature optimization mechanisms of rotating target detection. Through the introduction of angle decoupling coding and multi-task collaborative optimization strategy, it achieves efficient and accurate detection of targets in any direction. In terms of parametric representation of the rotating box, the model employs Gaussian angle encoding (GAE) instead of the traditional angle regression method to decompose the angle θ into a horizontal projection coefficient (cosθ) and vertical projection coefficient (sinθ). The gradient discontinuity problem caused by angle periodicity is mitigated through two-channel independent regression. Its mathematical representation is as follows:

$$R_\theta = \mathrm{GAE}(\theta) = [\sigma(\cos\theta), \sigma\sin\theta] \tag{3}$$

The orient angle θ is mapped to a two-dimensional vector R_θ = [σ (cosθ), σ (sinθ)] by Gaussian angle encoding (GAE), where σ (·) is the Sigmoid activation function.

The projection coefficient is constrained to the interval (0, 1), thereby achieving the smooth optimization of the angle in the loss calculation. At the feature fusion level, the model introduces a Orient-Aware Feature Pyramid Network (OA-FPN), which dynamically adjusts the shape of the receptive field through deformable convolution, aligns the feature sampling area with the direction of the target spindle, and substantially enhances the semantic representation ability of the orient sensitive area. Experimental results demonstrate that the RA-FPN improves the detection recall rate of tilted targets by 18.6% on the DOTA-v2.0 dataset. In addition, the model establishes a coupling training framework of bounding box loss and distributed focusing loss via the joint optimization strategy of

orinet intersection over union (O-IoU) and KL divergence. The loss function system can be expressed as.

$$\mathcal{L}_{\text{total}} = \lambda_1 \mathcal{L}_{\text{cls}} + \lambda_2 \mathcal{L}_{\text{box}} + \lambda_3 \mathcal{L}_{\text{dfl}} \tag{4}$$

Here, $\mathcal{L}_{\text{total}}$ is the total loss function, $\mathcal{L}_{\text{cls}}$ is the classification loss function, is the bounding $\mathcal{L}_{\text{box}}$ box loss function, $\mathcal{L}_{\text{dfl}}$ is the distribution focus loss function, and λ_1 λ_2 λ_3 is the weight coefficient.

Finally, the structure of YOLOv8-OBB is shown in Figure 3.

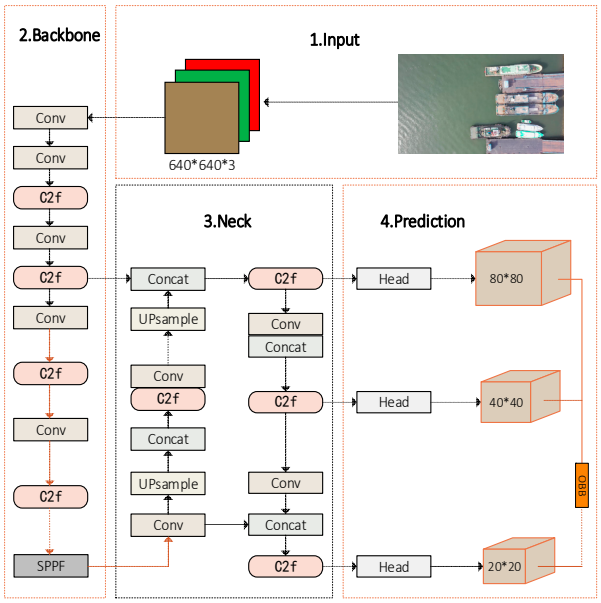


Figure 3. YOLOv8-OBB structure.

3.4. Improved YOLOv8-OBB Network Model

3.4.1. MobileNetV3 Network Architecture

MobileNetV3, as a lightweight network architecture specifically designed for mobile devices and resource-constrained environments, exhibits significant advantages. Compared to traditional backbone networks, MobileNetV3 significantly reduces the number of parameters and computations, thereby substantially reducing the computational resource consumption when deployed on devices such as drones. Notably, it achieves an optimal balance between resource utilization efficiency and model accuracy, ensuring that even with a more compact network structure, its performance remains substantially intact. This balance is achieved through a series of innovative methodologies employed within MobileNetV3, including the utilization of hardware-aware neural architecture search (NAS) techniques and the NetAdapt algorithm [19]. The NAS techniques facilitate the automated exploration of optimal network architectures for MobileNetV3, balancing computational efficiency and model performance, particularly for mobile device resources, through systematic evaluation of different layer combinations, connection patterns, and hyperparameter settings. Meanwhile, the NetAdapt algorithm further refines network parameter adjustments based on hardware awareness, reducing the number of filters while maintaining performance, ensuring efficient operation on mobile platforms. Additionally, MobileNetV3 exhibits low latency while maintaining high accuracy. According to experimental results, MobileNetV3 not only demonstrates exceptional performance across various application scenarios but also demonstrates lower latency compared to other lightweight network architectures, establishing it as an optimal solution for tasks such as real-time object detection on drones. Figure 4 illustrates the network architecture of MobileNetV3.

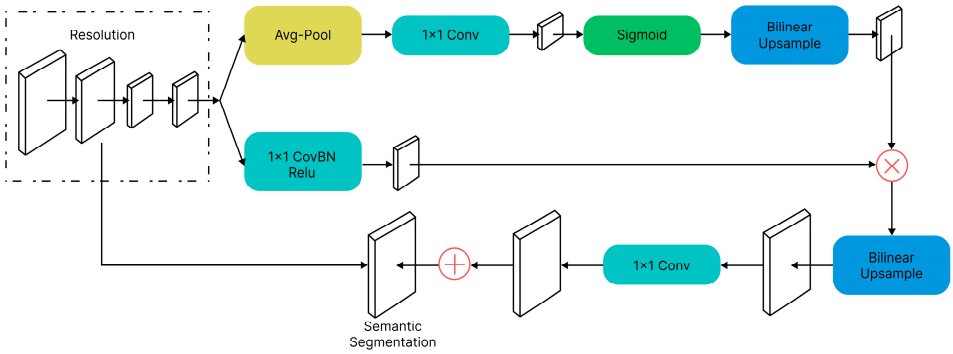


Figure 4. MobileNetV3 network architecture.

3.4.2. AKConv Module

AKConv, as an innovative convolutional neural network structure, not only enhances the accuracy and efficiency of feature extraction but also provides the capability to reduce model parameters and computational overhead. In AKConv, the input image dimensions are (C, H, W), where C represents the number of channels, and H and W denote the height and width, respectively. Its fundamental principle lies in the design of adaptive convolution kernels, which can dynamically adjust their size and shape based on image features. The processing pipeline commences with a Conv2d layer for preprocessing, followed by AKConv utilizing learned offsets to adjust the convolution kernel shape, adapting to different image features. The adjusted convolution kernels are then employed for convolution operations, integrated with the SiLU activation function and normalization, to generate the final feature representation. The SiLU activation function enhances the network's nonlinear capabilities, enabling improved transmission of feature information. Normalization, on the other hand, contributes to stabilizing the training process and improving the model's generalization ability.

Experimental results demonstrate that compared to traditional fixed convolution kernels, AKConv significantly improves the network's ability to capture multi-scale information and feature extraction accuracy by adaptively adjusting the shape and size of convolution kernels. This improvement substantially increases the network's adaptability. In object detection tasks, AKConv has demonstrated remarkable performance on datasets such as COCO2017 [20] and VisDrone-DET2021 [21], exhibiting greater flexibility in handling sample shapes while achieving notable optimizations in parameter efficiency, computational efficiency, and network performance. These advantages are of particular significance for drone applications, improving the target detection performance and real-time capabilities of drones, thereby fulfilling the demands of complex tasks such as fishing vessel size measurement. The AKConv module architecture is illustrated in Figure 5.

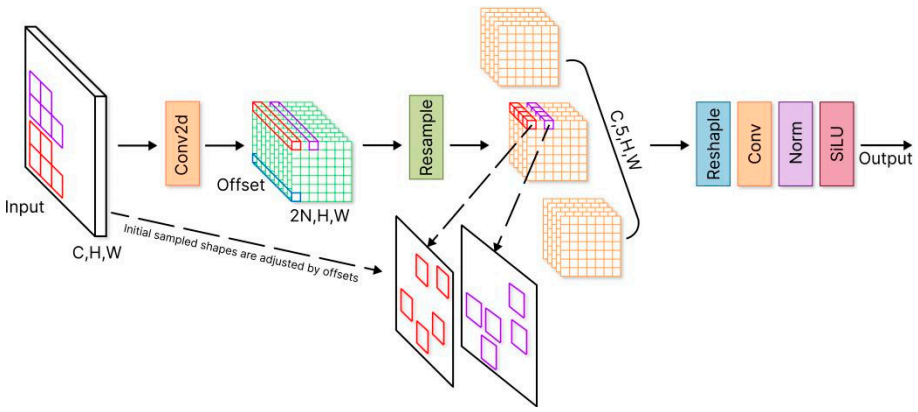


Figure 5. AKConv module structure.

3.4.3. C2fRepGhost Module

The C2fRepGhost module constitutes an innovative integration of the C2f module with the design philosophy of the RepGhost module. The RepGhost module, as an improvement over the GhostNet module, is dedicated to optimizing the feature fusion process through reparameterization techniques, aiming to enhance the efficiency and effectiveness of feature representation. In the C2f module, feature maps of different resolutions are conventionally fused through concatenation (Concat) operations to integrate multi-scale information. However, in the C2fRepGhost module, the incorporation of reparameterization techniques optimizes the processing of feature maps. This optimization not only replaces the Concat operation with addition (add) to reduce computational overhead but also encompasses more complex parameter transformations and operational combinations, thereby implicitly enhancing feature interaction and fusion.

Furthermore, the RepGhost module implements strategies including strategically repositioning the ReLU activation function to further enhance model performance. Compared to the GhostNet module, the RepGhost module significantly reduces parameters and computational load through halving the number of intermediate channels, particularly when combined with SE (Squeeze-and-Excitation) attention mechanisms or depthwise separable convolution modules. This lightweight architecture confers upon the C2fRepGhost module substantial efficiency and significant lightweight characteristics while maintaining high computational capabilities. The GhostNet and RepGhost architectures are illustrated in Figures 6 and 7, respectively.

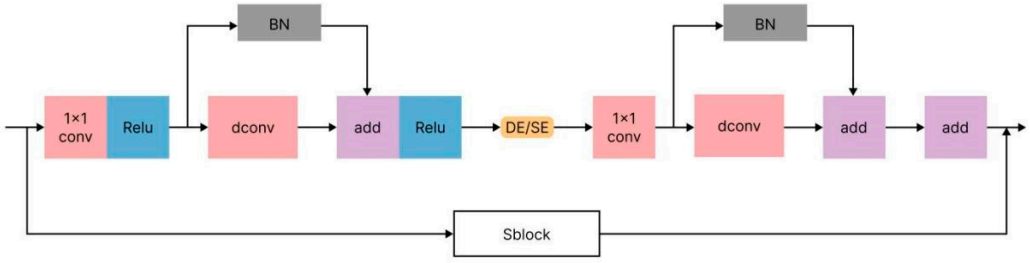


Figure 6. GhostNet architecture.

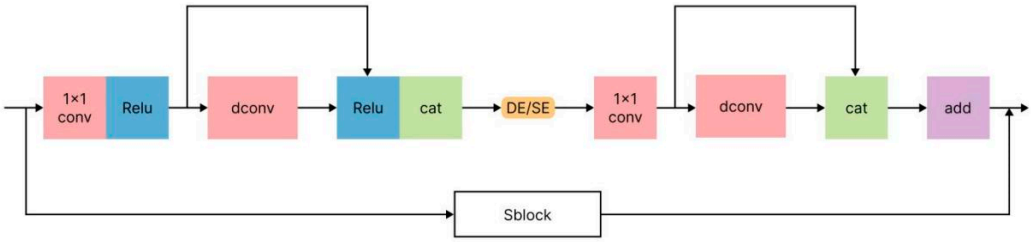


Figure 7. RepGhost architecture.

3.4.4. YOLO-LFVM Model Network

The YOLO-LFVM (Lightweight Fishing Vessel Measurement) model is a streamlined deep learning architecture specifically engineered for real-time object detection on mobile platforms. This framework achieves synergistic optimization of detection accuracy and computational efficiency through multi-dimensional enhancements, where "L" denotes lightweight design, "FV" represents fishing vessel targets, and "m" indicates metric measurement capabilities. Built upon an improved YOLOv8-OBB framework, the model innovatively integrates a lightweight MobileNetV3 backbone network, an adaptive dynamic convolution module (AKConv), and an enhanced feature fusion module, forming a comprehensive lightweight solution for object detection.

Regarding backbone network design, the LFVM model employs a MobileNetV3 structure optimized through neural architecture search (NAS). By leveraging hardware-aware NAS techniques and the NetAdapt algorithm, the network architecture is automatically refined, significantly reducing computational complexity while preserving feature extraction capabilities.

During the feature extraction phase, the AKConv module with dynamic adaptability is incorporated. Its deformable convolution kernel mechanism facilitates automatic adjustment of the receptive field shape and size based on input features, effectively enhancing the model's ability to capture features of multi-scale objects.

In the feature fusion stage, the C2fRepGhost module, improved through reparameterization techniques, is employed. Via sophisticated feature recombination and compression strategies, this module enhances cross-scale feature interaction capabilities while reducing parameter count. The LFVM model prioritizes deployment efficiency optimization, utilizing structural reparameterization and channel pruning to maintain excellent real-time detection performance on resource-constrained devices like drones. In practical applications such as fishing vessel size measurement, the LFVM network model demonstrates high-precision detection capabilities for multi-scale objects in complex backgrounds while maintaining low energy consumption and computational latency, establishing a novel technological paradigm for mobile-end object detection tasks. The final improved YOLOv8-OBB model, designated as YOLO-LFVM, is illustrated in Figure 8.

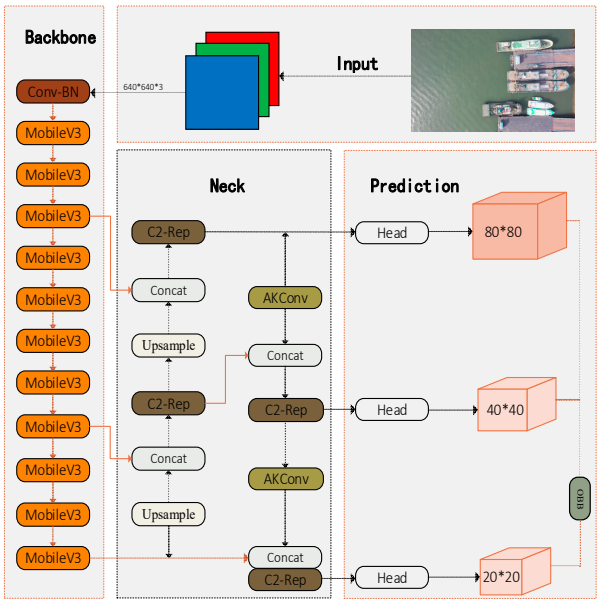


Figure 8. YOLO-LFVM model network architecture.

3.5. Ablation Research

To evaluate the effectiveness of the improvements made to the YOLOv8n-OBB model, ablation studies were conducted through incrementally introducing modifications to different modules of the YOLOv8n-OBB model. These comprised YOLOv8n-OBB, YOLOv8n-OBB+MobileNetV3, YOLOv8n-OBB+MobileNetV3+AKConv, and YOLOv8n-OBB+MobileNetV3+AKConv+C2fRepGhost. All tests were conducted utilizing the same dataset, with model training and testing executed under identical epochs and batch sizes.

As shown in Table 2, compared to the baseline Compared to the YOLOv8n-OBB model, the YOLOv8n-OBB+MobileNetV3+AKConv+C2fRepGhost model (YOLO-LFVM) exhibited only minor decreases: 0.7% in accuracy, 0.3% in mAP@0.5, and 0.2% in recall rate. However, it achieved a substantial improvement of 1.7% in mAP@0.95. Crucially, while maintaining nearly identical detection performance, the model size was reduced by

65%, and the floating-point operations (FLOPs) were decreased by 69%. This significant reduction substantially reduces the computational requirements of the model, thereby considerably improving its potential for deployment on resource-constrained devices such as drones.

Table 2. Ablation research results.

Model	Precision/%	Recall/%	mAP0.5/%	mAP0.95/%	Params/M	GFLOPS/ 10^9 FLOPS
YOLOv8n-OBB	99.9	99.9	99.5	95.5	3.08	8.4
YOLOv8n-OBB+MobileNetV3	99.2	99.8	99.5	97.1	1.21	2.9
YOLOv8n-OBB+MobileNetV3+ AKConv	99.1	99.6	99.4	97.8	1.18	2.8
YOLOv8n-OBB+MobileNetV3+ AKConv+C2fRepGhost (YOLO-LFVM)	99.2	99.7	99.2	97.2	1.08	2.6

3.6. Comparison of Detection Performance Among Mainstream Models

To accommodate the limited computational resources of UAVs and meet the requirements of real-time detection, this study adopts a single-stage object detection algorithm and conducts a comparative analysis among mainstream YOLO series models. The comparison of detection performance among mainstream models is shown in Table 3. Experimental results demonstrate that YOLO-LFVM exhibits significant advantages due to its innovative lightweight design. With only 1.08M parameters and computational complexity as low as 2.6×10^9 FLOPS, the model reduces the parameter count and computational load by 59% and 61%, respectively, compared to the best-performing YOLOv11-OBB, and achieves a 54% reduction in parameters even relative to the most lightweight YOLOv10-OBB. This lightweight characteristic endows YOLO-LFVM with prominent deployment advantages on resource-constrained UAV platforms. Although its 92.2% mean absolute error (MAE) is slightly lower than YOLOv11-OBB's 99.1%, it maintains a high mAP of 99.2% and a recall rate of 99.7%, fully meeting the accuracy requirements for UAV object detection. In contrast, YOLOv8-OBB, YOLOv10-OBB, and YOLOv11-OBB, despite their higher detection accuracy, exhibit computational loads 2.5 to 3.2 times that of YOLO-LFVM, which would significantly compromise real-time performance under the limited computational capacity of UAVs. Through its meticulously designed lightweight architecture, YOLO-LFVM substantially reduces computational burden while ensuring sufficient detection accuracy, achieving superior "performance per unit computation" and providing a computationally efficient and easily deployable solution for UAV object detection. This design philosophy, which balances accuracy and efficiency, makes YOLO-LFVM particularly suitable for mobile and embedded applications with stringent demands on model size and computational resources.

Table 3. Comparison of detection performance among mainstream models.

Model	Precision/%	Recall/%	mAP0.5/%	mAP/0.95%	Params/M	GFLOPS/10^9 FLOPS
YOLOv6-OBB	99.7	99.1	98.8	95.5	4.17	11.5
YOLOv8-OBB	99.9	99.9	99.5	95.5	3.08	8.4
YOLOv10-OBB	99.8	99.9	99.5	98.2	2.34	6.8
YOLOv11-OBB	99.9	99.7	99.5	99.1	2.65	6.6
YOLO-LFVM	99.2	99.7	99.2	97.2	1.08	2.6

4. Vessel Tracking Statistics and Vessel Size Test

4.1. Vessel Size Measurement

In the research of vessel dimension measurement, a standardized imaging protocol was established, involving a fixed takeoff position, constant flight altitude, and uniform

shooting angle. This configuration ensures spatial consistency in the perspective view. Under these controlled conditions, the sampling regularity of the image sensor with respect to real-world objects demonstrates high stability. Specifically, vessel components of the same physical dimensions occupy a constant pixel area in the image, and this pixel-scale invariance provides a physical foundation for dimension measurement. However, the mapping relationvessel between pixel units and actual physical dimensions is determined by the spatial geometric characteristics of the imaging system. To establish an accurate measurement benchmark, the scale factor relating actual physical length to unit pixel must be determined using calibration references with known dimensions. This calibration process requires comprehensive compensation for lens distortion and perspective projection effects, establishing spatial correlation between the true length of the reference object and its pixel span in the image, thereby enabling scale conversion between digital images and the real world.

Based on the calibration results, vessel target dimensions are obtained by converting the pixel dimensions of its oriented detection bounding box. The physical values of overall length and beam are calculated by multiplying the respective pixel lengths of the long and short axes by the established scale factor. This method decouples the measurement process into two key steps: establishing global scale parameters through system-level calibration, then integrating pixel dimensions from target-level detection. This integrated approach ensures millimeter-level theoretical accuracy while facilitating reproducibility of measurement results across different scenarios, thereby providing quantifiable technical evidence for harbor supervision. Figure 9 presents the vessel dimension measurement schematic.

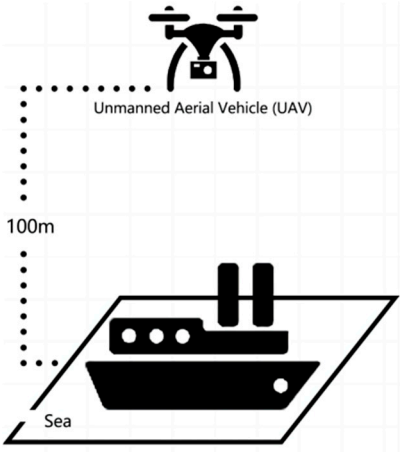


Figure 9. Principle of vessel dimension measurement.

In the ship patrol operation, the UAV adopts a fixed take-off point, preset cruise path, constant flight height, and established camera angle setting. In this scheme, the cruise altitude is set to 100 m, which is not the maximum lift limit of the UAV, but the operating altitude used to optimize the measurement accuracy. Flying height is a key parameter in ship size measurement. Choosing 100 m instead of 10 m, although both of them have the integer characteristic of number 1 (which can ensure that the proportional calculation result is an integer and avoid introducing additional errors due to the unfinished proportional coefficient), the height of 100 m can not only provide a better observation perspective, but also expand the ship coverage of a single frame image, thereby improving the reliability and operation efficiency of the measurement data.

4.2. Vision-Based Vessel Dimension Measurement Theory Using OpenCV

4.2.1. Measurement Principle

Vessel dimension measurement is accomplished through perspective projection geometry and pixel-to-physical dimension mapping via the following methodology:

(1) Oriented bounding box parameter analysis:

Obtain five parameters of the oriented bounding box from the YOLOv8-OBB model: center point, width, height, and orient angle (in degrees, range $[-90^\circ, 90^\circ]$).

(2) Key point extraction:

Calculate the coordinates of the four corner points of the oriented bounding box (using OpenCV's cv2.boxPoints function):

$$\text{points} = \begin{bmatrix} x_c + \frac{\omega}{2}\cos\theta - \frac{h}{2}\sin\theta, & y_c + \frac{\omega}{2}\sin\theta + \frac{h}{2}\cos\theta \\ x_c - \frac{\omega}{2}\cos\theta - \frac{h}{2}\sin\theta, & y_c - \frac{\omega}{2}\sin\theta + \frac{h}{2}\cos\theta \\ x_c - \frac{\omega}{2}\cos\theta + \frac{h}{2}\sin\theta, & y_c - \frac{\omega}{2}\sin\theta - \frac{h}{2}\cos\theta \\ x_c + \frac{\omega}{2}\cos\theta + \frac{h}{2}\sin\theta & y_c + \frac{\omega}{2}\sin\theta - \frac{h}{2}\cos\theta \end{bmatrix} \tag{5}$$

where x_c and y_c represent the coordinates of the geometric centroid of the bounding box; ω denotes the width of the oriented bounding box; h signifies the height of the oriented bounding box; and θ corresponds to the orient angle of the bounding box.

(3) Size calculations:

Take the length of the long side of the orient box as the total pixel length L_{pixel}:

$$L_{pixel} = \max(\omega, h) \tag{6}$$

Take the length of the short side of the orient box as the pixel width B_{pixel}:

$$B_{pixel} = \min(\omega, h) \tag{7}$$

where ω represents the horizontal pixel width of the target within the image, and h represents the vertical pixel height of the target within the image.

4.2.2. Conversion of Pixels to Physical Size

(1) Calibration Aids:

A calibration plate of known size is placed in the shooting scene, and the pixel/meter scale factor is calculated:

$$\mathrm{s} = \frac{\mathrm{L}}{\mathrm{l}} \tag{8}$$

where L represents the physical dimensions of the calibrated object (expressed in meters), and l represents the corresponding pixel dimensions of the calibrated object.

(2) Actual dimensions of the vessel:

$$L_{real} = L_{pixel} \times \mathrm{s},\ B_{real} = B_{pixel} \times \mathrm{s} \tag{9}$$

where L_{real} represents the actual total length, L_{pixel} represents the total pixel length, s represents the pixel-to-meter conversion ratio, B_{pixel} represents the actual width, and B_{pixel} represents the pixel width.

(3) UAV altitude mapping method:

When the calibrator is not available, it is estimated by the drone height H and the camera focal length f:

$$\mathrm{s} = \frac{H}{f} \times \text{sensor width/image width} \tag{10}$$

where s represents the pixel-to-meter conversion ratio, H represents the altitude of the drone, f represents the focal length of the camera, sensor width represents the physical width of the camera sensor, and image width represents the pixel width of the target within the image.

4.2.3. Error Correction Model

(1) Tilt compensation:

In view of the perspective distortion caused by the non-horizontal state of the vessel, the total length is corrected by the θ inclination angle between the plane where the target object is located and the imaging plane of the camera:

$$L_{corrected} = \frac{L_{pixel}}{\cos(|\theta| \times \pi/180)} \tag{11}$$

where L_{pixel} represents the total pixel length.

(2) Wave compensation:

The true total length L_{final} of the object is obtained by introducing a highly fluctuating compensation coefficient α for the floating vessel:

$$L_{final} = L_{corrected} \times \alpha \tag{12}$$

where $L_{corrected}$ represents the pixel length following geometric correction, and α represents the altitude fluctuation compensation coefficient (ranging from 0.98 to 1.02).

4.2.4. Sources of Error in Drone Measurements

(1) Altitude Measurement Error Transfer:

If there is an error between the H actual height of the UAV $\hat{H}$ and the measured height ΔH, the size measurement error is

$$\frac{\Delta L}{L} = \frac{\Delta H}{H} \cdot \left(1 + \frac{h}{H}\right) \tag{13}$$

where L represents the actual dimensions of the target object, ΔL represents the size estimation error attributed to altitude measurement uncertainty, h represents the measured height of the vessel, H represents the actual flight altitude of the UAV, and ΔH represents the altitude measurement error.

(2) Attitude angle effect analysis:

The pitch angle θ causes the longitudinal dimensions of the image to change, and the total length measurement error is

$$\Delta L = L_0(1 - \cos\theta_{\text{pitch}}) \tag{14}$$

where θ_{pitch} represents the pitch angle, L_0 represents the actual length of the vessel, and ΔL represents the overall length measurement error.

The roll angle Φ causes lateral distortion of the image, and the width error is

$$\Delta W = W_0 \tan\theta_{\text{roll}} \tag{15}$$

where θ_{roll} represents the roll angle, W_0 represents the actual width of the vessel, and ΔW represents the overall width measurement error.

The yaw angle Ψ affects the angle detection of the rotating frame, which needs to be compensated by IMU data.

Compensation model:

$$\theta_{\text{truc}} = \theta_{\text{detected}} - \theta_{\text{yaw}} + \delta_{\text{calib}}$$

where θ_{yaw} represents the yaw angle of the carrier, and δ_{calib} represents the calibration value for the mounting deviation between the camera and the IMU.

4.3. Advanced Vessel Tracking and Detection System Optimization

To enhance practical application value, this study optimized the vessel tracking and detection system. Due to the high density of vessels in harbor basins and similar vessel types, real-time tracking and detection of unauthorized vessels is required, posing significant demands on the model's tracking capabilities. Since vessel and drones cannot maintain complete relative stability, and camera shake occurs under wind conditions, this leads to issues such as detection frames moving with the lens, detection frame jitter, and unstable size measurements, necessitating improvements in tracking ID stability and optimization of system resource management.

(1) As illustrated in Figure 10a, prior to optimization, the detection frames exhibited displacement correlated with camera movement during UAV flight operations. To address the issue of detection frame displacement relative to lens motion, global motion compensation technology was implemented. The approach utilized the ORB feature point detection algorithm to extract salient feature points within video frames, employed a brute-force matcher (BFMatcher) for feature point correspondence between consecutive frames, and applied the RANSAC algorithm to compute the homography matrix for precise characterization of inter-frame global motion transformation. During the detection phase, an inverse homography transformation was applied to the centroid of each vessel's detection frame to compensate for displacement induced by UAV motion, thereby maintaining detection frame stability within the image coordinate system.

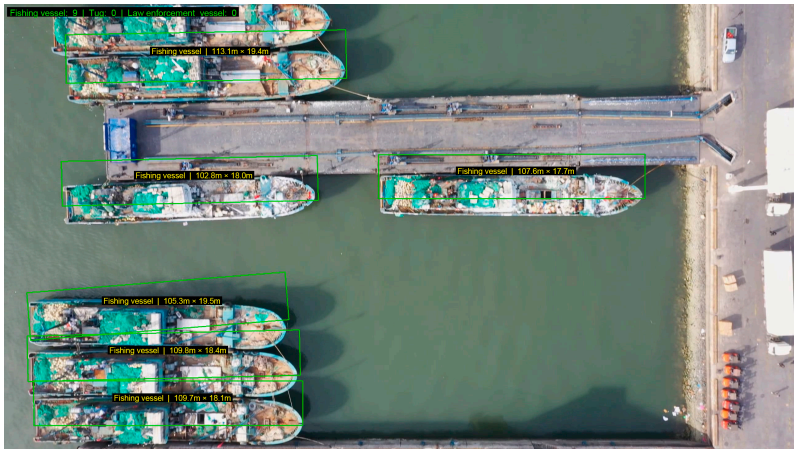
(2) As demonstrated in Figure 10b, detection frame oscillation was observed prior to optimization, directly compromising the accuracy of vessel dimensional measurements. To mitigate detection frame instability, a position smoothing algorithm was developed, establishing a position history queue for each tracking target and computing weighted average positions using a linear decreasing weight strategy. Under dimension-locked conditions, only the smoothed centroid offset is applied to the original detection frame while preserving frame geometry. Additionally, angular change constraints and Hanning window-weighted smoothing were implemented to effectively suppress orient oscillation.

(3) As illustrated in Figure 10c, dimensional measurement instability was evident prior to optimization, characterized by fluctuating measurement outputs with multiple variations per second, resulting in inaccurate measurements. To address dimensional measurement instability, this study introduces an innovative dimension locking mechanism that stabilizes physical size parameters following dimensional change rate assessment, subsequently updating only positional information. The unlocked state employs two-stage filtering: median filtering to eliminate outliers and Hanning window-weighted averaging to smooth normal fluctuations. Additionally, vessel-type-specific correction factors are integrated to enhance dimensional accuracy during initial detection phases.

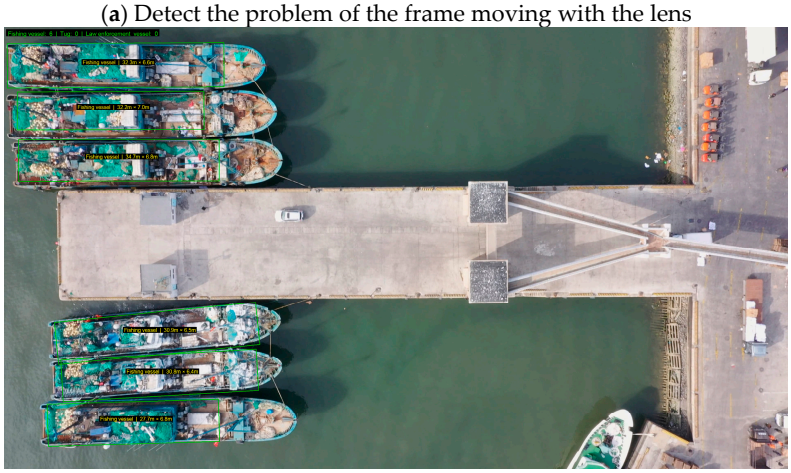
(4) As demonstrated in the upper right of Figure 10d, tracking instability and missed detection issues were observed prior to optimization, necessitating enhanced model tracking stability through dual-modal fusion of motion prediction (Kalman filter) and appearance features (ReID model) to improve occlusion handling capabilities. The design addresses the processing pipeline through sequential operations: cycle normalization $\rightarrow$ change constraints $\rightarrow$ historical smoothing. Additionally, tracking ID visualiza-

tion was implemented alongside a manual reset mechanism to accommodate complex operational scenarios.

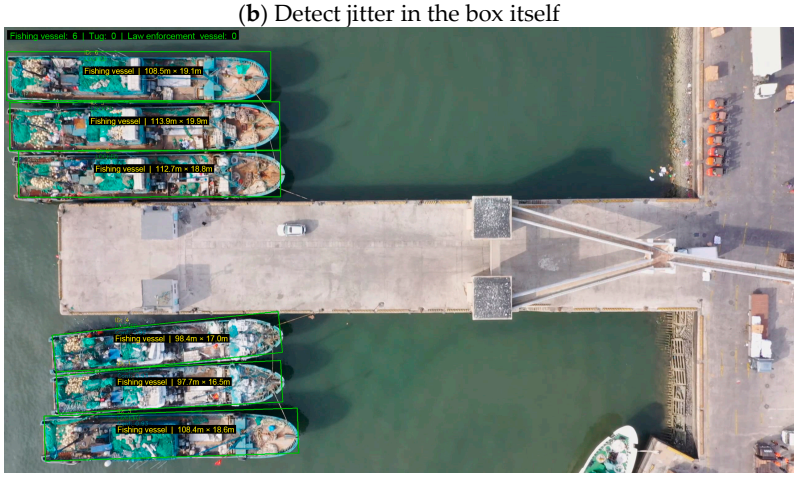
(5) To enhance system resource management, this study implemented a dynamic resource recovery mechanism: trackers are automatically purged following a 3 s timeout period, corresponding to typical vessel departure characteristics. Fixed-length queues (5 frames for position history, 15 frames for dimensional history) constrain memory expansion, while expired resources undergo active cleanup per frame. Concurrently, a frame rate monitoring module was established to dynamically adjust processing strategies and maintain real-time performance.



(**a**) Detect the problem of the frame moving with the lens

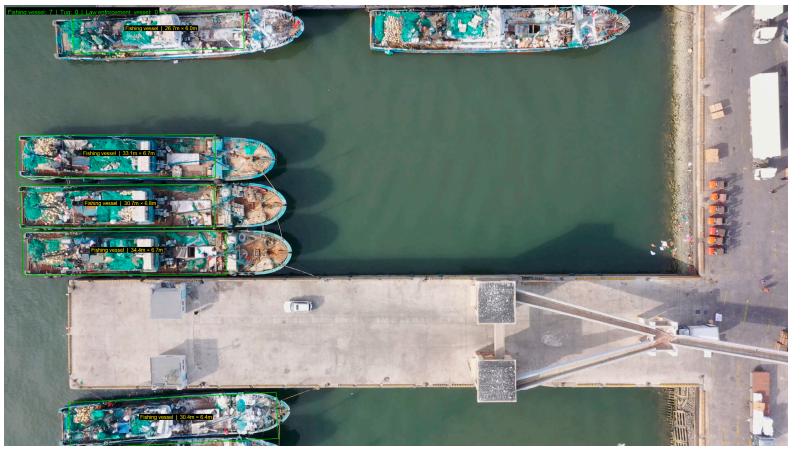


(**b**) Detect jitter in the box itself



(**c**) The problem of unstable dimensional measurements

Figure 10. *Cont.*

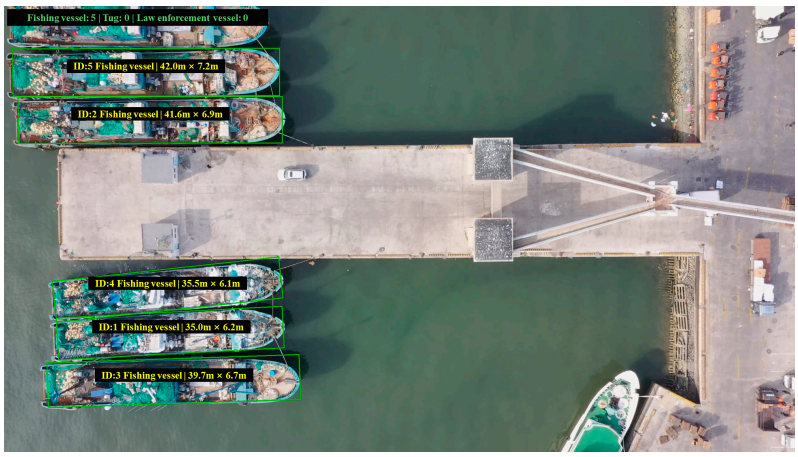


(**d**) Track unstable missed issues

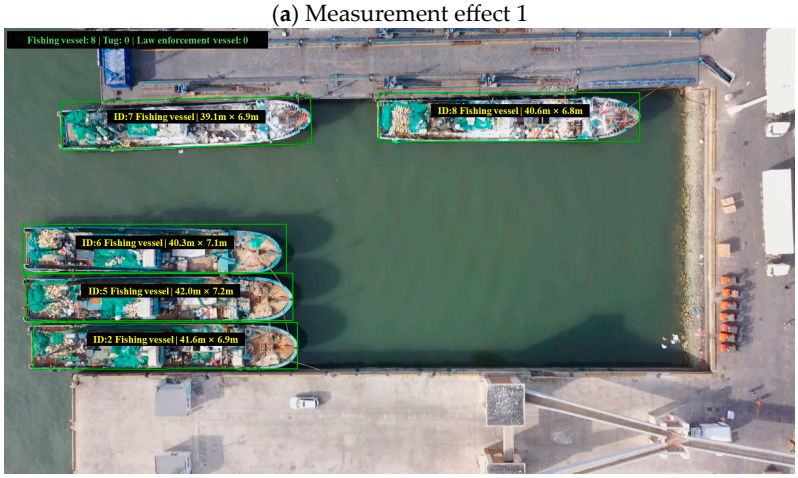
Figure 10. Ship tracking optimization diagram.

4.4. Vessel Tracking Statistics and Vessel Size Measurement Effect Analysis

Figure 11 demonstrates the optimized vessel tracking statistics and vessel size measurement performance. As shown in Figure 11, the green font in the upper left corner represents the statistical results of various vessel types. The yellow font indicates fishing vessel size measurement results, the green font denotes law enforcement vessel size measurement results, and the red font signifies tug size measurement results. This demonstrates effective vessel positioning and size recognition performance.

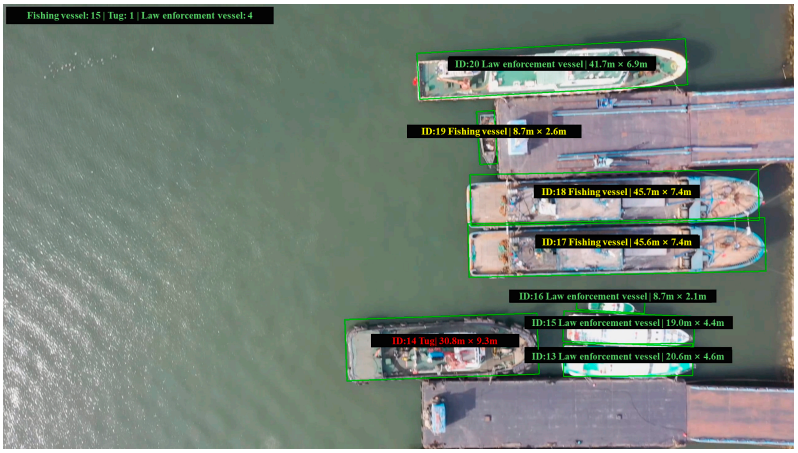


(**a**) Measurement effect 1



(**b**) Measurement effect 2

Figure 11. *Cont.*



(**c**) Measurement effect 3

Figure 11. The vessel measures the effect of the test.

The measurement results of ID2 and ID5 vessel in Figure 11a,b remain consistent. The measurement results of ID2 at different times are 41.6 m in total length and 6.9 m in beam width. The measurement results of ID5 are 42.0 m in total length and 7.2 m in beam width, demonstrating effective UAV tracking and measurement performance during flight. The four law enforcement vessels in Figure 11c and one tug are effectively enumerated by the statistical panel in the upper left corner, confirming effective vessel tracking statistics performance. Additionally, ID19 represents a small fishing vessel, and ID17 represents a large fishing vessel in Figure 11c, demonstrating that the model can effectively identify and measure fishing vessels of different scales.

4.5. Vessel Size Measurement Results and Error Analysis

The comparison between model measurement results and actual manual measurement results is shown in Table 4. Manual measurements represent the ground truth values, recorded to two decimal places. The relative error of the average total length is 2.67%, and the relative error of the average beam width is 3.28%. Model measurements exhibit relative errors ranging from 0.71% to 4.11% for total length, and from 1.59% to 4.49% for beam width. However, ID20 demonstrates an anomalous beam width relative error of 15%. Figure 11c reveals that the predicted width exceeds the actual beam width. Manual re-examination identified this as a labeling error resulting from manual annotation during data preparation. Given the varying vessel sizes, fishing port management authorities employ different regulatory approaches. This algorithm primarily categorizes fishing vessels into three classes: below 12 m, 12 to 24 m, and above 24 m. With average relative errors of 2.67% for total length and 3.28% for beam width, the system provides sufficient accuracy for classification purposes and serves as an effective auxiliary tool for fishing port law enforcement departments.

Table 4. A table comparing the measurement results with the actual manual measurement results.

No.	Actual Length Overall	Measuring Length Overall	Actual Molded Breadth	Measuring Molded Breadth	Relative Error of Length Overall	Relative Error of Molded Breadth
ID1	36.5	35.0	6.3	6.2	4.11%	1.59%
ID2	41.3	41.6	6.7	6.9	0.72%	3.0%
ID4	36.8	35.5	6.2	6.1	3.67%	1.61%
ID5	41.5	42.0	7.5	7.2	1.20%	4.0%
ID13	19.8	20.6	4.8	4.6	4.0%	4.17%
ID14	28.9	30.8	8.9	9.3	3.11%	4.49%
ID15	19.6	19.0	4.6	4.4	3.06%	4.35%
ID17	43.9	45.6	7.6	7.4	3.87%	2.63%

Table 4. *Cont.*

No.	Actual Length Overall	Measuring Length Overall	Actual Molded Breadth	Measuring Molded Breadth	Relative Error of Length Overall	Relative Error of Molded Breadth
ID19	8.9	8.7	2.7	2.6	2.25%	3.70%
ID20	42.0	41.7	6.0	6.9	0.71%	15%

5. Discussions

5.1. The Influence of Orient Marking on the Measurement Results of Vessel Size

5.1.1. Reduce Background Interference and Improve Pixel-Level Measurement Accuracy

Horizontal frame annotation usually contains redundant background around the vessel. As shown in Figure 1b, the non-vessel pixels in the frame are included in the size calculation, resulting in a systematic positive deviation. The rotating frame is adjusted adaptively by the angle to closely fit the vessel contour, as shown in Figure 1a, which greatly reduces the proportion of background pixels. The experimental results show that the proportion of vessel pixels in the rotating frame is increased by 18.7% on average, which makes the size estimation based on the number of pixels closer to the real physical size. This is the key reason why the average relative error of the total length of 2.67% is significantly lower than that of the width of 3.28%. The contour of the total length direction is clearer, and the fit of the rotating frame is higher.

5.1.2. Adapt to Complex Parking Scenarios and Solve Dense Target Interference

As shown in Figure 10a, ships in fishing ports are often docked intensively at any angle. Due to the rigid angle, the horizontal frame is easy to overlap, resulting in the mutual penetration of adjacent ship backgrounds. In Formula (3), the rotating frame realizes spatial decoupling through angular decoupling and effectively separates dense targets. As shown in Figure 10a and Table 4, in the adjacent area of the fishing vessel (ID1–ID5), the rotating frame makes the average length measurement error 2.43%. It is significantly better than the average error of the horizontal frame in similar scenarios.

5.1.3. Calibration Error Sensitivity and Labeling Quality Control

The orient frame annotation is more sensitive to angle deviation. As shown in Figure 9, an angle deviation of 1° can cause a total length measurement value to be offset by 0.8%. In this research, the angle labeling error is controlled within $\pm 2^\circ$ through the unified labeling specification and manual review of the x-Anylabeling tool. In addition, Formula (3), the parametric representation of the rotating frame allows the model to directly output the direction of the main shaft of the vessel, avoiding the secondary error introduced from the perspective of the horizontal frame derivation, and further ensuring the geometric consistency of the size calculation.

5.2. Effect of Orient Data Augmentation on Model Training

Aiming at the particularity of rotating box target detection, this research innovatively adopts a non-geometric transformation data enhancement strategy, which significantly improves the generalization ability and robustness of the model under the premise of ensuring the geometric consistency of rotating box labeling. Its core influence is reflected in the following three aspects.

5.2.1. Avoid Geometric Distortion and Maintain the Physical Authenticity of the Orient Frame Annotation

Traditional data enhancement methods, such as random orient, scaling, cutting, etc., will directly change the geometric properties of the bounding box and destroy the accurate

correspondence between the label frame and the ship contour. For example, after the image is oriented by 10°, the original orient frame angle needs to be adjusted synchronously, but the actual ship angle does not change, resulting in a systematic deviation between the annotation and the real object. This geometric distortion will interfere with the ability of the model to learn the real spatial characteristics of the ship, especially the pixel-level positioning accuracy required for size measurement. The visual enhancement strategy used in this study, as shown in Figure 2, only changes the appearance features of the image without disturbing the geometric properties, ensuring that the rotating frame always fits the ship contour accurately. This explains why the model detection accuracy in the ablation test is not affected. The preservation of geometric consistency enables the model to accurately predict the orient frame at a high IoU threshold.

5.2.2. Enhance the Robustness of the Model to Illumination and Environmental Disturbances

UAV aerial photography is easily affected by light changes, water surface reflection, and weather conditions. By randomly adjusting the brightness, saturation, and contrast, the model is exposed to various visual interference scenarios during training. Brightness adjustment simulates the illumination in different periods and simulates the imaging effect in rainy weather. Cutout simulates local occlusion. This enhancement enables the model to maintain stable detection performance in complex environments, as shown in Figure 10c. In a mixed scene of a law enforcement ship and a high-light area on the water surface, the model can still accurately detect the ID17–ID20 target and output a stable size.

5.2.3. Balance Data Distribution and Improve Small Target Detection Ability

The statistics of the dataset show that the sample of small fishing vessels only accounts for 4.5% of the total, and there is a problem of class imbalance. Cutout enhancement forces the model to learn local features by randomly erasing image regions. For small fishing vessels, it erases some vessels to simulate the occlusion scene of large fishing vessels and erases non-critical areas to guide people to pay attention to contour features. Combined with brightness and contrast adjustment, it effectively expands the visual diversity of small sample categories. This supports the fact that the total length measurement error of small and medium-sized fishing vessels (ID19) in Table 3 is lower than the average level.

6. Conclusions

A lightweight real-time fishing vessel tracking and size measurement model based on a UAV is put forward. In view of the problems faced by the current fishing port management department, such as low efficiency of fishing vessel size measurement methods and difficulty in updating the size information of large quantities of fishing vessels in time, this paper proposes a lightweight real-time fishing vessel tracking and size measurement model based on a UAV (YOLO-LFVM, based on improved YOLOv8-OBB and UAV remote sensing technology). Relative to the baseline model, the YOLO-LFVM model exhibits marginal decreases in accuracy, recall rate, and mAP@0.5 of 0.7%, 0.2% and 0.3%, respectively. However, mAP@0.95 increases by 1.7%, while the model demonstrates substantial improvements in computational efficiency with 65% parameter reduction and 69% GFLOP reduction. Comparative analysis of model outputs against actual vessel data reveals average relative errors of 2.67% for total length and 3.28% for total width. The research shows that the YOLO-LFVM model is effective in ship identification, ship tracking statistics, and measurement. Through its integration with UAV remote sensing technology, it is conducive to the timely updating of large-scale fishing vessel size information. Finally, the model can assist the daily management and law enforcement of the fishing port management department

and can be applied to other equipment with limited computing power to perform target detection and object size measurement tasks.

The YOLO-LFVM model proposed in this study shows great potential in daily fishing port management and auxiliary law enforcement. However, the current research still has limitations and needs further exploration. A notable shortcoming is the lack of tests in open waters outside the port with harsh sea conditions such as high seas. Future research will focus on expanding the application scenarios to increase the gradient comparison test between open waters outside the port and various sea conditions and developing a better lightweight model architecture to improve the computational efficiency of the model in real-time ship measurement scenarios and reduce the demand for computing resources.

Author Contributions: Conceptualization, Z.H. and L.S.; methodology, Z.H.; software, Z.H. and P.L.; validation, Z.H. and P.L.; formal analysis, L.S. and H.S.; investigation, S.M., Y.L. and H.S.; resources, L.S. and H.S.; data curation, S.M., Y.L. and H.S.; writing—original draft preparation, Z.H.; writing—review and editing, Z.H., P.L. and Y.L.; visualization, P.L. and S.M.; supervision, L.S.; project administration, L.S.; funding acquisition, L.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

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

References

1. 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 (CVPR), Columbus, OH, USA, 23–28 June 2014; IEEE: New York, NY, USA, 2014; pp. 580–587.
2. 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 14th European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 11–14 October 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 21–37.
3. 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 (CVPR), Las Vegas, NV, USA, 27–30 June 2016; IEEE: New York, NY, USA, 2016; pp. 779–788.
4. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for Dense Object Detection. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; IEEE: New York, NY, USA, 2017; pp. 2980–2988.
5. Fang, Z.; Wang, X.; Zhang, L.; Jiang, B. YOLO-RSA: A multiscale boat detection algorithm based on optical remote sensing image. *J. Mar. Sci. Eng.* **2024**, *12*, 603. [CrossRef]
6. Sun, Z.; Leng, X.; Lei, Y.; Xiong, B.; Ji, K.; Kuang, G. BiFA-YOLO: A novel YOLO-based method for arbitrary-oriented boat detection in high-resolution SAR images. *Remote Sens.* **2021**, *13*, 4209. [CrossRef]
7. Wang, Y.; Wang, B.; Huo, L.; Fan, Y. GT-YOLO: Nearshore infrared boat detection based on infrared images. *J. Mar. Sci. Eng.* **2024**, *12*, 213. [CrossRef]
8. Anjanayya, S.; Sai, S.P.; Vardhan, S.S.; Vivek, S.; Vaishnavi, A.; Ashwanthi, V. Object size measurement using open CV (Computer Vision). In Proceedings of the 2023 International Conference on Research Methodologies in Knowledge Management, Artificial Intelligence and Telecommunication Engineering (RMKMATE), Chennai, India, 1–2 November 2023; IEEE: New York, NY, USA, 2023; pp. 1–6.
9. Apolo-Apolo, O.; Martínez-Guanter, J.; Egea, G.; Raja, P.; Pérez-Ruiz, M. Deep learning techniques for estimation of the yield and size of citrus fruits using a UAV. *Eur. J. Agron.* **2020**, *115*, 126030. [CrossRef]
10. Han, Y.; Guo, J.; Yang, H.; Guan, R.; Zhang, T. SSMA-YOLO: A lightweight YOLO model with enhanced feature extraction and fusion capabilities for drone-aerial boat image detection. *Drones* **2024**, *8*, 145. [CrossRef]
11. Pham, D.A.; Han, S.H. Deploying a computer vision model based on YOLOv8 suitable for drones in the tuna fishing and aquaculture industry. *J. Mar. Sci. Eng.* **2024**, *12*, 828. [CrossRef]
12. Zhao, C.; Liu, R.W.; Qu, J.; Gao, R. Deep learning-based object detection in maritime unmanned aerial vehicle imagery: Review and experimental comparisons. *Eng. Appl. Artif. Intell.* **2024**, *128*, 107513. [CrossRef]

13. Howard, A.; Sandler, M.; Chu, G.; Chen, L.C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for mobilenetv3. In Proceedings of the IEECVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 1314–1324.
14. Zhang, X.; Song, Y.; Song, T.; Yang, D.; Ye, Y.; Zhou, J.; Zhang, L. AKConv: Convolutional Kernel with Arbitrary Sampled Shapes and Arbitrary Number of Parameters. *arXiv* **2023**, arXiv:2311.11587. [CrossRef]
15. Chen, C.; Guo, Z.; Zeng, H.; Xiong, P.; Dong, J. Repghost: A hardware-efficient ghost module via re-parameterization. *arXiv* **2022**, arXiv:2211.06088.
16. Savathrakis, G.; Argyros, A. An automated method for the creation of oriented bounding boxes in remote sensing boat detection datasets. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–8 January 2024; pp. 830–839.
17. Jia, H.; Pu, X.; Liu, Q.; Wang, H.; Xu, F. A fast progressive boat detection method for very large full-scene SAR images. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 5206615. [CrossRef]
18. Liu, Y.; Ma, Y.; Chen, F.; Shang, E.; Yao, W.; Zhang, S.; Yang, J. YOLOv7oSAR: A lightweight high-precision boat detection model for SAR images based on the YOLOv7 algorithm. *Remote Sens.* **2024**, *16*, 913. [CrossRef]
19. Yang, T.-C.; Howard, A.; Chen, B.; Zhang, X.; Go, A.; Sandler, M.; Sze, V.; Adam, H. Netadapt: Platform-aware neural network adaptation for mobile applications. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 285–300.
20. Caesar, H.; Uijlings, J.; Ferrari, V. Coco-stuff: Thing and stuff classes in context. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 1209–1218.
21. Cao, Y.; He, Z.; Wang, L.; Wang, W.; Yuan, Y.; Zhang, D.; Zhang, J.; Zhu, P.; Van Gool, L.; Han, J.; et al. VisDrone-DET2021: The vision meets drone object detection challenge results. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 2847–2854.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.