Smart Structures and Systems, *Vol. 34, No. 3 (2024) 157-169*
https://doi.org/10.12989/sss.2024.34.3.157

Trajectory monitoring of inland waterway vessels across multiple cameras based on improved one-stage CNN and inverse projection

Yitian Han [1,2a], Dongming Feng [1], Ye Xia [3], Rong Lin [1], Chan Ghee Koh [2] and Gang Wu*[1]

[1] *National and Local Joint Engineering Research Center for Intelligent Construction and Maintenance, Southeast University, Nanjing 211189, China*
[2] *Department of Civil and Environmental Engineering, National University of Singapore,117576, Singapore*
[3] *School of Civil Engineering, Tongji University, Shanghai 200092, China*

(*Received October 13, 2023, Revised September 27, 2024, Accepted September 30, 2024*)

Abstract. Accidents involving inland waterway vessels have raised concerns regarding monitoring their navigation tracks. The economical and convenient deployment of video surveillance equipment and computer vision techniques offer an effective solution for tracking vessel trajectories in narrow inland waterways. However, field applications of video surveillance systems face challenges of small object detection and the limited field of view of cameras. This paper investigates the feasibility of using multiple monocular cameras to monitor long-distance inland vessel trajectories. The one-stage CNN model, YOLOv5, is enhanced for small object detection by incorporating generalized intersection over union loss and a multi-scale fusion attention mechanism. The Bytetrack algorithm is employed to track each detected vessel, ensuring clear distinction in multiple-vessel scenarios. An inverse projection formula is derived and applied to the tracking results from monocular camera videos to estimate vessel world coordinates under potential water level changes in long-term monitoring. Experimental results demonstrate the effectiveness of the improved detection and tracking methods, with consistent trajectory matching for the same vessel across multiple cameras. Utilizing the Savitzky-Golay filter mitigates jitter in the entire final trajectory after timing-alignment merging, leading to a better fit of the dispersed trajectory points.

Keywords: attention mechanism; multiple cameras; multiple object tracking; object detection; vessel trajectory monitoring

1. Introduction

Due to its enormous capacity and eco-friendly characteristics, water transport has been a prominent form of transit for local and international commerce. With the explosive expansion of inland waterway transportation, the loading capacity and transportation velocity have continuously grown. The number of vessels, volume of waterborne trade, and amount of hazardous cargo are constantly on the rise. However, compared to navigation near the shore or at sea, river navigation is often constrained to relatively narrow streams with strong currents. As a result, accidents such as vessel-vessel collisions, vessel-bridge collisions, and vessel groundings are happening with increasing frequency, posing serious threats to the safety of navigation and the ecological preservation of rivers.

Active collision prevention methods can significantly reduce the probability of vessel collisions and groundings by providing warnings to vessels with high collision risks, and vessel object detection is one of the core tasks of these methods. Commonly used technologies for vessel detection include synthetic aperture radar (SAR) (Zhang *et al*. 2019, Raj *et al*. 2022), cameras (Li *et al*. 2019, 2021), and automation identification systems (AIS) (Valsamis *et al*. 2017, Liu *et al*. 2019) are commonly used for vessel detection. With the rapid development of computer vision algorithms in recent years, the camera-based method has gained attention for its potential in inland vessel detection, owing to its advantages of low equipment cost and high resolution and making it a promising option for active collision prevention.

Due to long-distance photography, weather variations, and complex backgrounds, it is challenging to acquire stable detection results based on traditional image processing methods (Szpak and Tapamo 2011). In recent years, artificial intelligence (AI) technologies, particularly deep learning-based methods, have been increasingly employed in various fields, including object detection. Numerous approaches have been proposed to address the generic object detection task. R-CNN (Girshick *et al*. 2014) is a pioneering work in the field of object detection, serving as a prominent two-stage framework. Building upon RCNN, the Fast R-CNN (Girshick 2015) and Faster R-CNN (Ren *et al*. 2017) algorithms have made notable advancements in accelerating the detection speed of two-stage networks. However, You Only Look Once (YOLO) (Redmon *et al*. 2016) and its variants (Redmon and Farhadi 2017, 2018, Bochkovskiy *et al*. 2020, Ultralytics 2020, Ge *et al*. 2021), as a popular single-stage object detector

*Corresponding author, Ph.D., Professor,
E-mail: g.wu@seu.edu.cn
[a] Ph.D. Candidate, E-mail: skye_hanyt@seu.edu.cn

ISSN: 1738-1584 (Print), 1738-1991 (Online)

serious, have attracted wider attention recently due to their remarkably faster detection speed compared to the traditional two-stage algorithms. In the field of vessel detection, Kong and Hu (2019) enhanced images by the horizon line detection and image registration section before using the VGG-based Siamese Network to track the target vessel. Li *et al*. (2021) adopted a self-supervised approach to learn specific visual representations and subsequently fine-tuned their model using supervised learning for accurate vessel detection. Additionally, they collected and labelled a fine-grained vessel dataset called HarborVessels. Lee *et al*. (2021) applied the YOLOv3 model for ship detection using ship-mounted cameras, enhancing situational awareness in maritime settings.

Vessel detection often entails the detection of small objects amidst complex backgrounds. Despite significant advancements in object detection algorithms propelled by deep learning, detecting small objects remains challenging (Mahaur and Mishra 2023). This is primarily due to the limited appearance information available for small objects, making it difficult to distinguish them from the background or similar objects. The attention mechanism is an important data processing method within the realm of machine learning (Tang *et al*. 2023). The principle of the attention mechanism in computer vision is to improve the model's focus by highlighting essential feature information while disregarding less important details. It involves learning the weight distribution of feature layers through structural layers and applying it to the original feature graph for weighted summation. Notably, Yang *et al*. (2021) proposed an attention module named SimAM, which can derive three-dimensional attention weights for feature graphs without requiring additional structures or parameters. This approach draws inspiration from neuroscience principles, assigning higher attention (weight) to neurons with spatial inhibition effects and achieving superior performance in terms of speed and accuracy.

Object detection in three-dimensional (3D) space is an important area of research, but it presents challenges when using monocular cameras due to the absence of depth information. Consequently, stereo vision cameras or binocular cameras are employed to address this issue (Omrani *et al*. 2020, Thombre *et al*. 2022). However, utilizing binocular cameras comes with increased computational resource requirements and setup complexities, particularly during field testing. In response to this challenge, Chabot *et al*. (2017) introduced a multitask network known as Deep MANTA for 3D vehicle localization and orientation estimation from monocular images. Clausse *et al*. (2019) utilized a mask and a 3D bounding box projection to compute the vehicle's center and orientation. However, the difficulty arises from the necessity to label the training dataset with 3D bounding boxes.

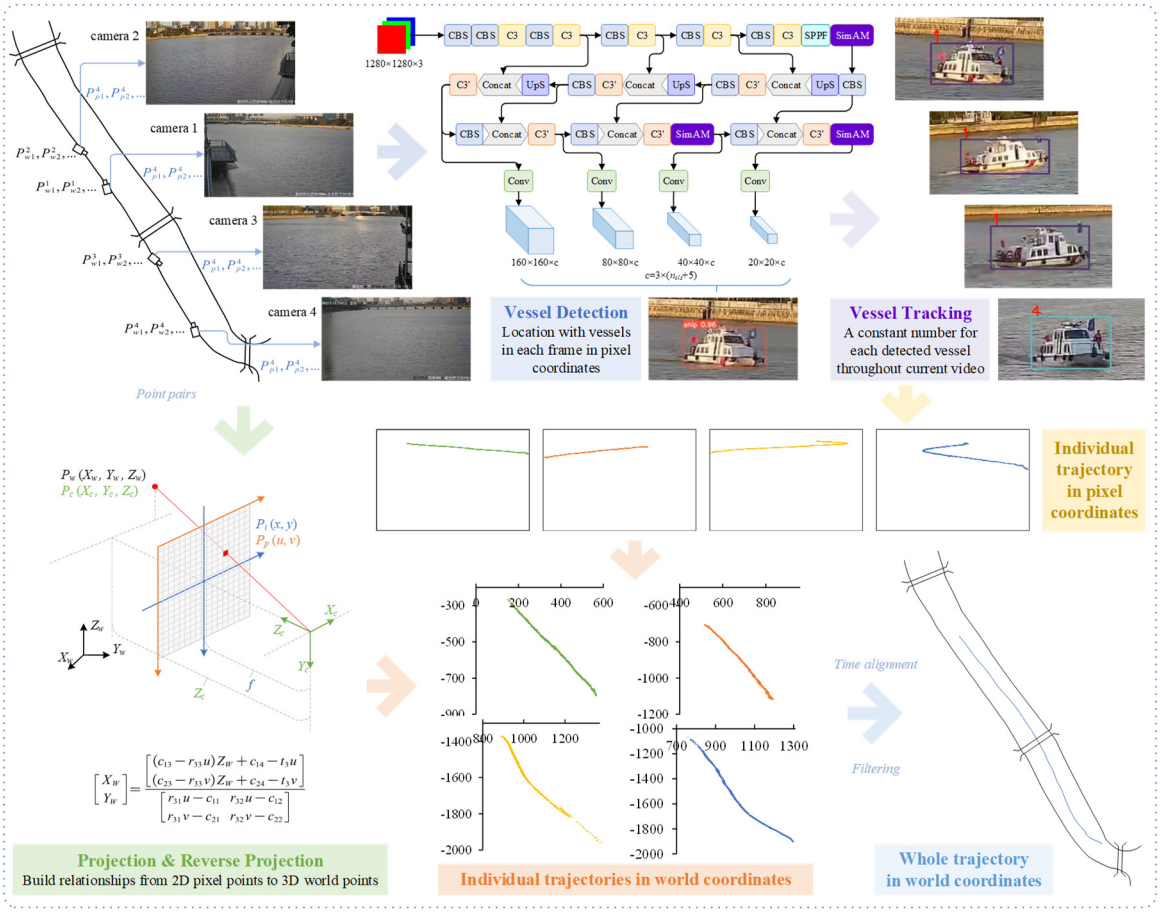


Fig. 1 The framework of the proposed approach

Homography is a mapping between two planes commonly used for perspective transformation and unification between images. Due to the flatness of water/road surfaces, homography transformation was also employed for mapping the trajectories of ships (Zhang *et al*. 2022a) and vehicles (Zhang and Zhang 2021) from the image plane to the road/water plane. The homography matrix can be calculated based on the pairing points coordinates in the image scale and physical scale on the water/road surface plane. However, the water surface elevation varies with the seasons, so the world coordinates of the points on the water surface may shift, and sometimes the world coordinates measurement of the desired points is limited by the inaccessibility of the water surface. Instead of mapping the relationship between two-dimensional (2D) planes, Yoon *et al*. (2018) utilized the projection relationship between 3D and 2D coordinates to restore the 6-DOF pose of the UAV at each frame using background features, and then calculated the displacement of the target structure based on the this pose from the 2D image. This paper presents the inverse projection equation in a more concise form, clearly linking the result to the object's position in the image plane and the water surface elevation. The camera's intrinsic and pose parameters can be calibrated using the checkerboard and point pair information from surrounding facilities or structures within the camera's field of vision. In this way, long-term field measurements can adapt to changes in water level, which can be obtained from hydrological monitoring agencies.

The primary contributions of this paper are summarized as follows: 1) A framework is proposed for vessel trajectory monitoring using multiple monocular cameras, enabling accurate tracking of multiple vessels across different camera views, with successful field testing. 2) The detection performance for small objects is enhanced by incorporating a multi-scale fusion attention mechanism and a loss function based on generalized intersection over union (GIoU). 3) The inverse projection relationship is presented in a more concise form, clearly linking the vessels' positions in the world coordinate system to their positions in the image plane and the water level, making the reconstruction of vessel trajectories more adaptable to seasonal water level changes. Fig. 1 shows the framework of the proposed approach.

2. Improved small object detection

2.1 Architecture of the network

The scale of the vessels in videos is not constant, and the multiple detection heads in the YOLOv5 structure will help the model to detect vehicles of different scales. There are five models of YOLOv5 with different depths and widths from nano, small, middle, and large to extra, namely YOLOv5n, YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. Among these models, YOLOv5n stands out with its minimal parameter count, making it the fastest in terms of detection speed. However, when considering performance aspects, it exhibits the lowest detection accuracy. On the contrary, YOLOv5x boasts the highest detection accuracy but at the cost of employing the most parameters and the slowest processing speed. YOLOv5l6 was selected in this study for its good performance without unnecessarily increasing network size and parameters. The "6" in "YOLOv5l6" signifies that the model downsamples the input size by a factor of at most $1/2^6$.

The structure of the model, as illustrated in Fig. 2, is composed of a backbone, neck, and head. It takes RGB images of size 1280×1280 as input and utilizes multiple-scale feature maps for predictions. The backbone plays a crucial role in receiving the input images and extracting both low-level and high-level features from the data. The CBS module and C3 module are employed successively to extract features from the images. The CBS module encapsulates three functions: Convolution (Conv), Batch Normalization (BN), and activation functions SiLU instead of Leaky ReLU in the older version. Each CBS module performs downsampling on the feature layer, reducing its height and width by half while doubling the number of channels. The SPPF module employs a cascade of multiple small-sized pooling kernels instead of a single large-sized pooling kernel used in the SPP module, raising the speed while preserving the original functionality of fusing feature

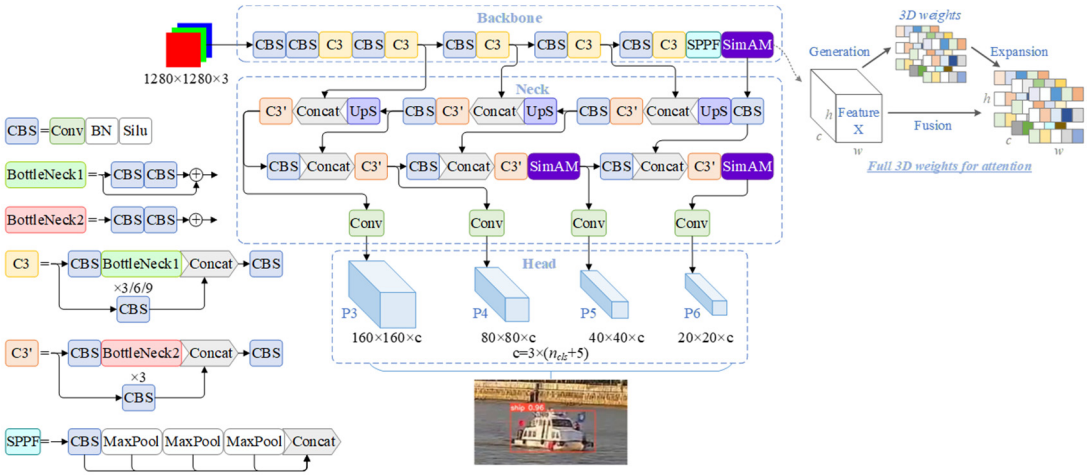


Fig. 2 The architecture of MSA-YOLOv5

maps from different receptive fields. The neck component of the model comprises the Feature Pyramid Network (FPN) (He *et al*. 2014) and Path Aggregation Network (PAN) (Liu *et al*. 2018) structures, constructed using the C3' module and CBS module. Its purpose is to fuse features from multiple levels and scales, allowing the model to capture both fine-grained details and global contextual information. The head of the model is responsible for generating the final output, which entails producing vessel predictions in the image based on four scale feature maps: P3, P4, P5, and P6, containing features ranging from low-level to high-level representations.

2.2 Multi-scale fusion attention mechanism

The definition of small objects in object detection can be categorized into absolute and relative scales. The absolute scale approach treats small objects as those that are smaller than a specified pixel size, while the relative scale approach treats them as those that occupy less than a certain percentage of the original image. It is common practice to consider objects smaller than 32 pixels ×32 pixels as small objects, which is also adopted in this paper. In this paper, the detection performance of YOLOv5 for small objects on the sea surface is improved by adding a simple, parameter-free attention module (SimAM) to make the object detection network pay more attention to small object features.

Unlike existing channel-wise and spatial-wise attention mechanisms, SimAM presents a full 3D weight and parameter-free attention mechanism inspired by well-established neuroscience theories. The spatial suppression theory behind SimAM believes that neurons with the most information exhibit distinctive firing patterns and can inhibit the activity of surrounding neurons. Identifying neurons with significant spatial suppression effects becomes crucial for visual processing, and this can be achieved by calculating the linear separability between a target neuron and others. Therefore, the following energy function is defined

$$e_t(w_t, b_t, y, x_i) = \frac{1}{M-1}\sum_{i=1}^{M-1}\left(-1-(w_t x_i + b_t)\right)^2 + \left(1-(w_t t + b_t)\right)^2 + \lambda w_t^2 \quad (1)$$

where t and x_i denote the target neuron and other neurons in a single channel of the input feature $\mathbf{X} \in \mathbb{R}^{C\times H\times W}$. i is index over spatial dimension and $M = H \times W$ is the number of neurons on that channel. Eq. (1) has a fast closed-form solution for transform weight w_t and bias b_t, which can be obtained as follows

$$w_t = -\frac{2(t-\mu_t)}{(t-\mu_t)^2 + 2\sigma_t^2 + 2\lambda} \quad (2)$$

$$b_t = -\frac{1}{2}(t+\mu_t)w_t \quad (3)$$

$\mu_t = \frac{1}{M-1}\sum_{i=1}^{M-1} x_i$ and $\sigma_t^2 = \frac{1}{M-1}\sum_{i=1}^{M-1}(x_i-\mu_t)^2$ are mean and variance of all neurons except t in that channel. The minimal energy can be computed with the following

$$e_t^* = \frac{4(\hat{\sigma}^2+\lambda)}{(t-\hat{\mu})^2 + 2\hat{\sigma}^2 + 2\lambda} \quad (4)$$

where $\hat{\mu} = \frac{1}{M}\sum_{i=1}^{M-1} x_i$ and $\hat{\sigma}^2 = \frac{1}{M}\sum_{i=1}^{M}(x_i-\hat{\mu})^2$. The importance of each neuron can be represented by $1/e_t^*$, so the SimAM module is finally optimized as

$$\widetilde{\mathbf{X}} = \text{sigmoid}\left(\frac{1}{\mathbf{E}}\right) \odot \mathbf{X} \quad (5)$$

Based on the performance evaluation of SimAM across various vision tasks, it effectively enhances the feature extraction capability of convolutional neural networks (ConvNets) without introducing additional parameters. When analysing the YOLOv5l6 model, head P3 possesses the smallest receptive field, making it well-suited for capturing intricate features in predicting tiny objects. Conversely, head P6 has the largest receptive field, enabling it to capture a broader context but containing coarser feature information primarily suitable for predicting large objects. To further enhance the model's ability to detect small objects, SimAM was incorporated ahead of heads P5 and P6 to compensate for their inherently coarser feature information. Similarly, the SimAM is also applied behind the SPPF module since it has the most fusing features from different receptive fields. The final model, as shown in Fig. 2, named MSA-YOLOv5, enhances attention on multi-level features, especially for small objects, with the help of a multi-scale fusion attention mechanism.

2.3 Loss function

The efficiency of object detection is highly dependent on the definition of the loss function, which typically consists of classification loss (L_{cls}) and bounding box regression loss (L_{obj} and L_{loc}). In single-class classification scenarios, such as in our study, the bounding box regression loss is particularly crucial. Intersection over union (IoU) is a commonly used metric in object detection, which calculates the ratio of the intersection and union of the prediction box (B) and the ground truth box (A). IoU loss treats four coordinate points as a whole for calculation, making the loss insensitive to scale. IoU loss (L_{IoU}) is defined as the negative log of IoU. Therefore, when two boxes perfectly coincide, the IoU is 1, and the loss is 0.

$$IoU = \frac{|A \cap B|}{|A \cup B|} \quad (6)$$

$$L_{IoU} = 1 - IoU \quad (7)$$

However, in cases where the prediction box (B) and the ground truth box (A) do not intersect, the IoU metric cannot reflect the distance between them, and the loss function is not differentiable. This poses a challenge for optimizing the loss function in cases where the boxes do not intersect. To address this issue, some IoUs are compared, and Generalized IoU (GIoU) (Rezatofighi *et al*. 2019) is finally

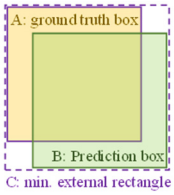


Fig. 3 Illustration of GIoU

selected for the following training according to its performance which will be shown in Section 2.4.4. GIoU introduces the concept of the minimum bounding rectangle (C) that encloses both boxes, allowing for a more accurate measure of their spatial relation. The GIoU and GIoU Loss (L_{GIoU}) are defined as follows

$$GIoU = IoU - \frac{|C - (A \cup B)|}{|C|} \tag{8}$$

$$L_{GIoU} = 1 - GIoU \tag{9}$$

In addition, for the four prediction feature layers P3, P4, P5, and P6, different L_{obj} weights were assigned as 4, 1, 0.25, and 0.06, respectively.

$$L_{obj} = 4.0L_{obj}^{P3} + 1.0L_{obj}^{P4} + 0.25L_{obj}^{P5} + 0.06L_{obj}^{P6} \tag{10}$$

2.4 Experiments

2.4.1 Data set

The dataset used for object detection in the study was provided by the 3rd International Competition for Structural Health Monitoring (ICSHM) (2022) and came from the vessel traffic recorded by the surveillance video for real bridges. A total of 1102 frame images utilized for training were sourced from cameras distinct from those used for testing. These training-set images included various weather conditions, such as sunny, rainy, and night, and images with one or more vessels. The dataset was manually labeled, and the annotation file contained information about each vessel's category label (uniformed as 0) and their normalized bounding box coordinates in individual images. Data augmentation techniques, such as random image flipping and color enhancement, were applied to the dataset. The training set was split into 80% for training the models and 20% for validation. Video sets recorded by four successional surveillance cameras were used as the test set.

Table 1 Description of the test video sets

Video Set No.	Description
1	Good weather, a single big ship
2	Good weather, a single small ship
3	Good weather, two ships
4	Evening, a single small ship
5	Rain, a single big ship
6	Rain, a single small ship
7	Rain, two ships

The description of these video sets can be seen in Table 1. Each set includes four videos that record the same vessel(s) from different views. The approximate installation location of these cameras is indicated in Fig. 1.

2.4.2 Evaluation metrics

The evaluation metrics used in this paper are average precision (AP) and average recall (AR) with different IoU thresholds. The IoU threshold indicates that only detections with an IoU greater than or equal to the threshold are considered correct detections. The calculation formulas of $Precision$ and $Recall$ are as follows

$$Precision = \frac{TP}{TP + FP} \tag{11}$$

$$Recall = \frac{TP}{TP + FN} \tag{12}$$

where *TP*, *FP* and *FN* indicate the number of detections with IoU higher than threshold, detections with IoU lower than threshold, and undetected ground truth instances, respectively.

2.4.3 Implementation details

The training was conducted on an Ubuntu system with 20 CPU cores and an NVIDIA Tesla V100-32GB GPU. The code was implemented in Python 3.8 using the PyTorch framework. The stochastic gradient descent (SGD) optimizer was used, and a cosine learning rate (LR) scheduler was employed to adjust the LR of the optimizer during training. The initial LR was set to 0.01, and the batch size was set to 16.

2.4.4 Detection results

Fig. 4 compares the average precision (AP) and average recall (AR) results for several well-established IoU metrics. The definitions of SIoU (Gevorgyan 2022), WIoU (Tong *et al*. 2023), EIoU (Zhang *et al*. 2022b), CIoU, and DIoU (Zheng *et al*. 2020) can be found in the respective references. GIoU demonstrates better performance in detecting small objects while maintaining comparable performance for medium and large objects. Compared to CIoU, which was used in the original YOLOv5 model, the AP and AR for small object detection improved by 14.64% and 20.69%, respectively, when GIoU was used.

Fig. 5 shows the loss, precision, recall, and average precision at IoU threshold 0.5 (AP-0.5) for both YOLOv5 and MSA-YOLOv5 models when utilizing GIoU. The loss curves indicate that the loss for both models gradually decreases and converges. The MSA-YOLOv5 model exhibits better performance in the evaluation metrics, particularly with an improvement in recall. This higher recall can help reduce missed detections, enhancing tracking continuity and minimizing track ID changes.

Table 2 shows the detection results for two models evaluated using COCO metrics AP@0.50:0.95 and AR@0.50:0.95. Despite the scores for small objects remaining lower than those for larger objects due to challenges such as resolution limitations and background interference, both AP and AR for small objects have

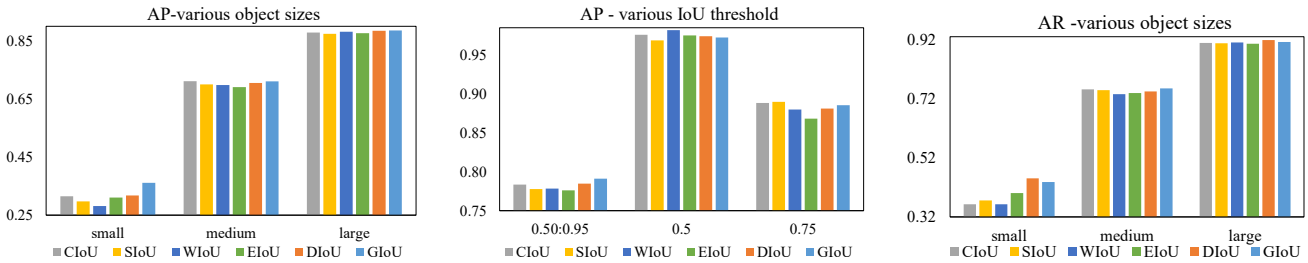
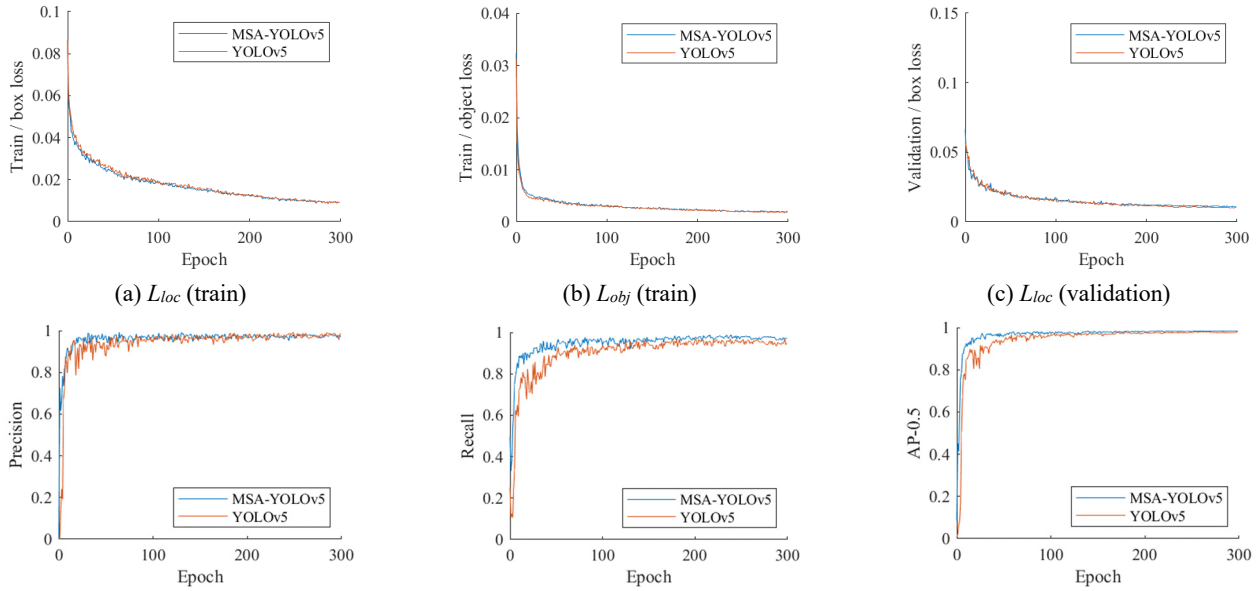


Fig. 4 Comparison of different IoU loss

Note: S = small object (0~32 pixels), M = medium object (32~96 pixels), L = large object (96~105 pixels)



(a) L_{loc} (train) (b) L_{obj} (train) (c) L_{loc} (validation)

(d) Precision (e) Recall (f) AP-0.5

Fig. 5 Results of YOLOv5 and MSA-YOLOv5

Table 2 Detection results for different size objects

Method		YOLOv5	MSA-YOLOv5	Improvement
AP @IoU = 0.50:0.95	all	0.791	0.769	-2.8%
	S*	0.361	0.460	27.5%
	M*	0.71	0.680	-4.3%
	L*	0.885	0.876	-1.0%
AR @IoU = 0.50:0.95	all	0.824	0.590	-0.6%
	S*	0.438	0.475	8.6%
	M*	0.755	0.743	-1.6%
	L*	0.912	0.910	0.3%

*S: small object (0~32 pixels), M: medium object (32~96 pixels), L: large object (96~105 pixels)

significantly improved compared to the original YOLOv5 model.

Lower detection performance for small objects may result in missed or false detections, affecting trajectory continuity or causing difficulties in matching trajectories across multiple cameras. The improved model helps mitigate this issue to some extent, ensuring that fewer small objects are overlooked. Setting a lower IoU threshold may capture more objects while increasing the risk of false positives. Therefore, selecting an appropriate IoU threshold requires consideration of the balance between precision and recall to achieve optimal detection outcomes.

The results indicate a slight decrease in the performance of MSA-YOLOv5 for medium and large objects. This may be attributed to the emphasis of the multi-scale attention mechanism on small object detection, which enhances sensitivity to finer features but leads to a performance trade-off for larger objects. While this focus may slightly diminish the model's generalization to larger objects, the impact on their detection performance is minimal.

Fig. 6 presents the detection outcomes using the two models in select frames. Both models exhibit the capability to detect objects of medium and large sizes. However, the MSA-YOLOv5 model outperforms the original YOLOv5 model in detecting objects of small sizes and frames characterized by vessel occlusion and night conditions.

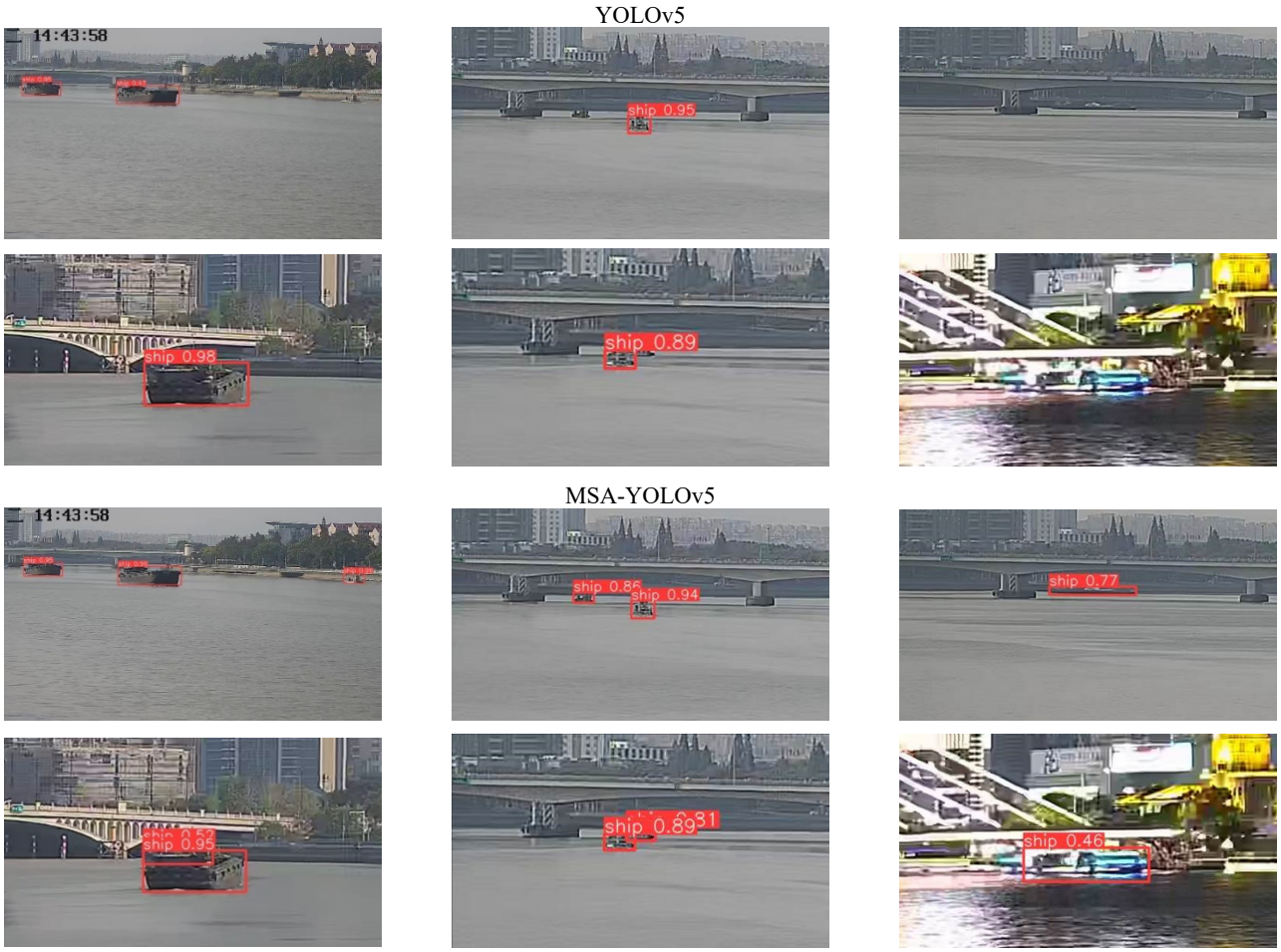


Fig. 6 Sample results of vessel detection

Fig. 7 Sample results of vessels tracking

3. Multi-vessel tracking

3.1 Bytetrack

The Bytetrack algorithm utilizes a Kalman filter and Hungarian algorithm, which differs from current mainstream trackers as it does not use deep learning or re-identification technology. When a video sequence is input, the detection boxes are divided into high-scoring and low-scoring detection boxes based on a threshold, and trajectories are created. The Kalman filter predicts the position and size of the detection boxes in the next frame, and the IOU between the predicted detection boxes and the current high-scoring detection frame is calculated. The trajectories are then matched with the high-scoring detection boxes of the current frame using the Hungarian algorithm based on the IOU. The successfully matched trajectories are updated, and the unmatched trajectories are

matched with the low-scoring detection boxes. New tracks are created for high-scoring boxes not successfully matched in the second match, and low-scoring boxes are deleted.

3.2 Tracking results

Fig. 7 presents sample results of vessel tracking, where each vessel is assigned a tracking number. These numbers reliably follow the vessels throughout their presence within the camera's field of view, regardless of the number of vessels or weather conditions. The tracking mechanism demonstrates consistent performance from when a vessel enters the camera's field of view until it exits.

The position of detected objects is represented by the coordinates of the top left corner x_1, y_1, and the width (w) and height (h) of the bounding box. When observing the detection results, the camera's observation angle introduces a bias when using the center of the bounding box as the location of the detected vessels. As shown in Fig. 8, this study adopted $\left(x_1+\frac{1}{2}w,\ y_1+\frac{3}{4}h\right)$ as the representative point for vessel objects in pixel coordinates to balance this bias.

Fig. 9 illustrates the vessel trajectories in pixel coordinates of video set No. 2 captured by individual cameras. This visualization provides an example of how the

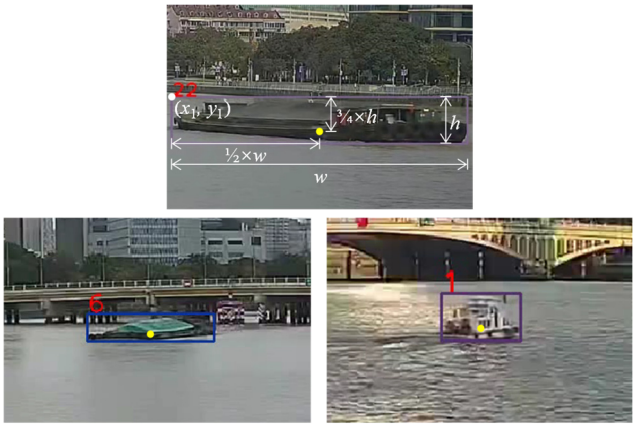
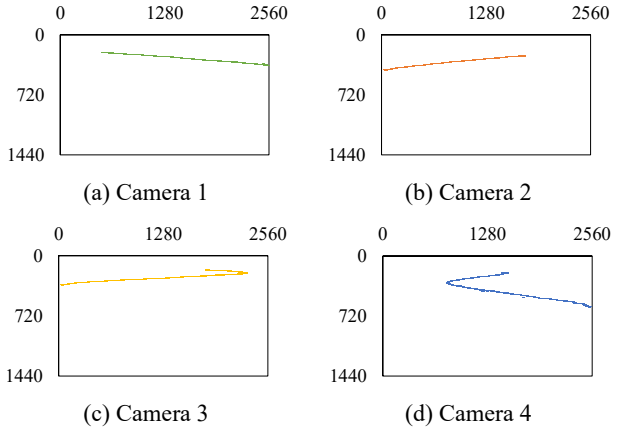


Fig. 8 Representative points for vessel objects



(a) Camera 1 (b) Camera 2

(c) Camera 3 (d) Camera 4

Fig. 9 Vessel trajectories in pixel coordinates in individual cameras (video set No. 2)

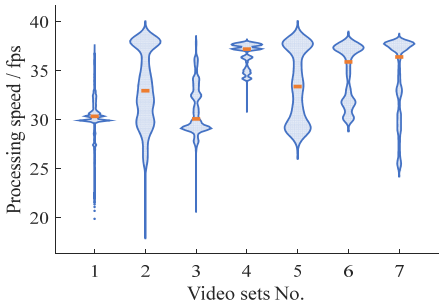


Fig. 10 Distribution of processing speed

vessels move within the frame of each camera throughout recorded videos.

The test videos were recorded at a frame rate of 25 frames per second (fps). Fig. 10 illustrates the distribution (in blue) and the average (in orange) of the processing speed for detection and tracking in each frame in every video set. Except for a small portion of frames, the detection and tracking processes can be executed at a rate exceeding 25 fps, indicating that they can effectively operate in real time.

4. Inverse projection from 2D pixel coordinates

4.1 Theoretical derivation

Fig. 11 shows the relationship among world coordinates, camera coordinates, image coordinates, and pixel coordinates, with $P_w(X_w, Y_w, Z_w)$, $P_c(X_c, Y_c, Z_c)$, $P_i(x, y)$, and $P_p(u, v)$ representing the respective point positions in these coordinate systems. This transformation relation has been introduced in numerous previous works; thus, it is not derived here. The final transformation relation is as follows

$$Z_c\begin{bmatrix}u\\v\\1\end{bmatrix}=\begin{bmatrix}f_x & 0 & c_x\\0 & f_y & c_y\\0 & 0 & 1\end{bmatrix}\begin{bmatrix}r_{11} & r_{12} & r_{13} & t_1\\r_{21} & r_{22} & r_{23} & t_2\\r_{31} & r_{32} & r_{33} & t_3\end{bmatrix}\begin{bmatrix}X_w\\Y_w\\Z_w\\1\end{bmatrix}$$
$$=\boldsymbol{K}[\boldsymbol{R}\quad \boldsymbol{T}]\begin{bmatrix}X_w\\Y_w\\Z_w\\1\end{bmatrix}^{\mathrm{T}} \tag{13}$$

Here, $\boldsymbol{K}=\begin{bmatrix}f_x & 0 & c_x\\0 & f_y & c_y\\0 & 0 & 1\end{bmatrix}$ is the intrinsic matrix of the camera, which can be obtained by checkerboard calibration, $\boldsymbol{R}=\begin{bmatrix}r_{11} & r_{12} & r_{13}\\r_{21} & r_{22} & r_{23}\\r_{31} & r_{32} & r_{33}\end{bmatrix}$ is the rotation matrix, $\boldsymbol{T}=[t_1\quad t_2\quad t_3]^{\mathrm{T}}$ is the translation matrix, and $[\boldsymbol{R}\quad \boldsymbol{T}]$ constitutes the extrinsic matrix of the camera.

Perspective-n-Point (PnP) is the problem of estimating the pose of a calibrated camera based on a set of 3D points in the world and their corresponding 2D projections in the image. The least square method is commonly used to find the optimal solution when multiple point pairings are available. However, some measurements may deviate from

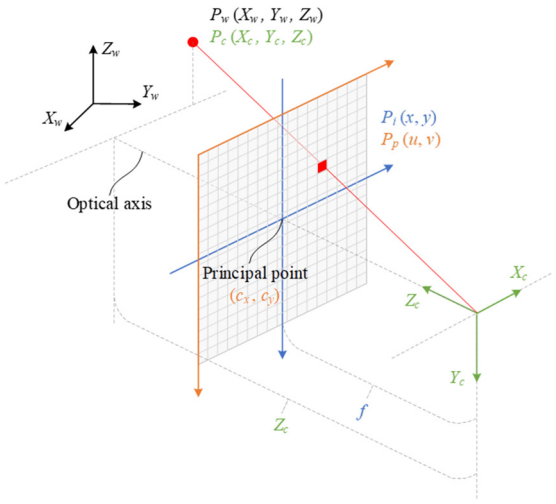


Fig. 11 The relation between multiple coordinate systems

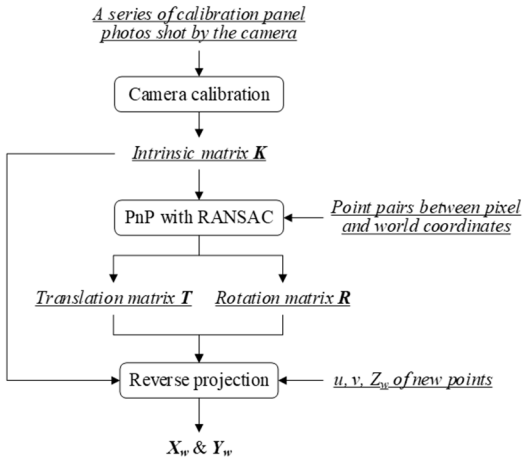


Fig. 12 Flowchart of inverse projection procedure

the actual values in real-world scenarios. To address this, the random sample consensus (RANSAC) algorithm is employed to remove outliers and improve the camera's extrinsic matrix estimation. The extrinsic matrices can be solved by the PnP method with RANSAC.

Fig. 11 demonstrates that if the intrinsic and extrinsic matrices are determined, the 2D pixel projection from 3D world coordinates is certain, whereas the projection from 2D to 3D is not. That is why binocular vision is usually used to locate the coordinate trajectories of objects in 3D space. In this study, vessels are observed on an inland water surface where the elevation remains relatively constant within a limited stretch of the waterway. Therefore, through the 2D pixel coordinates (u, v) and one of the 3D world coordinates (X_w, Y_w, Z_w), such as Z_w chosen in this paper, the remaining 2D coordinate values can be solved. Eq. (13) can be reformulated as follows to obtain the calculation equation for this relationship

$$\begin{aligned}&[r_{31}\ r_{32}\ r_{33}\ t_3]\begin{bmatrix}X_w\\Y_w\\Z_w\\1\end{bmatrix}\begin{bmatrix}u\\v\end{bmatrix}\\&=\begin{bmatrix}f_x & 0 & c_x\\0 & f_y & c_y\end{bmatrix}\begin{bmatrix}r_{11} & r_{12} & r_{13} & t_1\\r_{21} & r_{22} & r_{23} & t_2\\r_{31} & r_{32} & r_{33} & t_3\end{bmatrix}\begin{bmatrix}X_w\\Y_w\\Z_w\\1\end{bmatrix}\end{aligned} \tag{14}$$

Defining $C=\begin{bmatrix}f_x & 0 & c_x\\0 & f_y & c_y\end{bmatrix}\begin{bmatrix}r_{11} & r_{12} & r_{13} & t_1\\r_{21} & r_{22} & r_{23} & t_2\\r_{31} & r_{32} & r_{33} & t_3\end{bmatrix}=\begin{bmatrix}c_{11} & c_{12} & c_{13} & c_{14}\\c_{21} & c_{22} & c_{23} & c_{24}\end{bmatrix}$ for convenience of notation, Eq. (14) can be organized as follows

$$\begin{bmatrix}X_w\\Y_w\end{bmatrix}=\frac{\begin{bmatrix}(c_{13}-r_{33}u)Z_w+c_{14}-t_3u\\(c_{23}-r_{33}v)Z_w+c_{24}-t_3v\end{bmatrix}}{\begin{bmatrix}r_{31}u-c_{11} & r_{32}u-c_{12}\\r_{31}u-c_{21} & r_{32}u-c_{22}\end{bmatrix}} \tag{15}$$

where all parameters on the right-hand side were obtained beforehand except for u, v, and Z_w. Fig. 12 depicts the entire reverse projection procedure from 2D pixel to 3D world coordinates.

4.2 Results of projection and inverse projection

Fig. 13 shows the projection results of four cameras in pixel coordinates and the inliers' mean absolute error (MAE) compared to the original given data. Take the coordinates of each group of point pairs at the upper and lower boundaries of x and y coordinate axes as boundary boxes, and take their diagonal length to normalize the error as

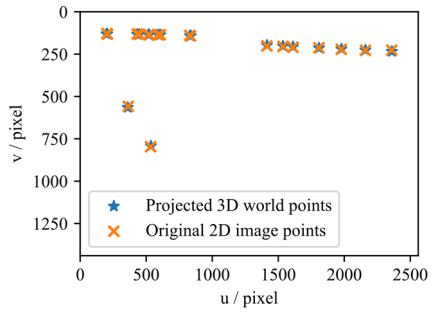
$$NMAE=\frac{MAE}{\sqrt{\left(P_{\max}^{i,x}-P_{\min}^{i,x}\right)^2-\left(P_{\max}^{i,y}-P_{\min}^{i,y}\right)^2}} \tag{16}$$

where p is the coordinates of the point pairs and i is the camera number. The points exhibit a strong alignment except for a few outliers discarded by the RANSAC method.

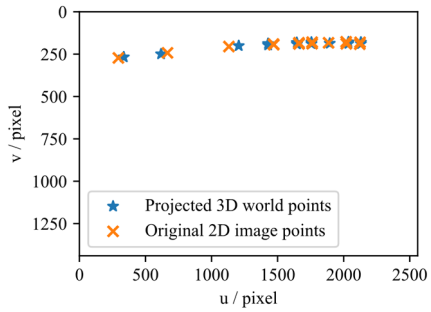
To provide a more intuitive understanding of the results, a scaled river representation was first created based on Baidu Map as Fig. 14(a), enabling a better visualization of the positions of all the points along the river. Fig. 14(b) compares all point pairs on the same graph, in which the circles represent the positions of the original marked points in world coordinates, while the crosses represent the positions in world coordinates obtained by performing inverse projection calculations using the pixel coordinates of the point pairs. According to Eq. (16), the NMAEs of the inverse projection result from the four cameras are 5.44%, 9.05%, 8.88%, and 10.61%, respectively.

It can be observed that their distributions are generally consistent, but the accuracy is not as high as in the forward projection from 3D to 2D. This is because inverse projection requires higher accuracy in marking the point pairs than forward projection. At positions farther away from the camera, even a difference of one pixel can correspond to tens or even hundreds of meters in distance. The error can be reduced by increasing the accuracy of the measurement.

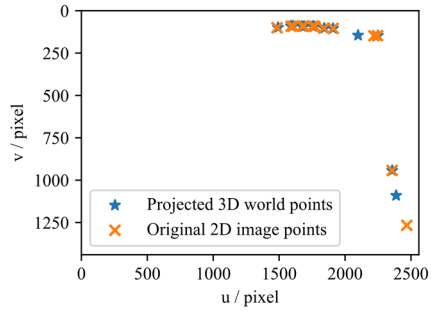
Since the competition prompt did not provide the height of the water surface in the world coordinate system, it is



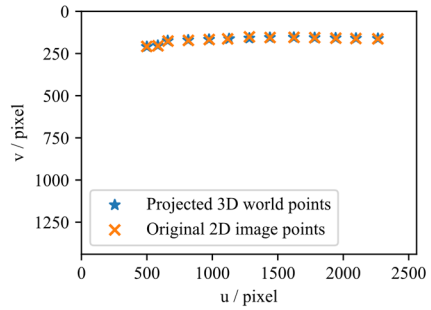
(a) Camera 1: MAE = 2.66 pixels, NMAE = 0.12%



(b) Camera 2: MAE = 25.56 pixels, NMAE = 1.40%

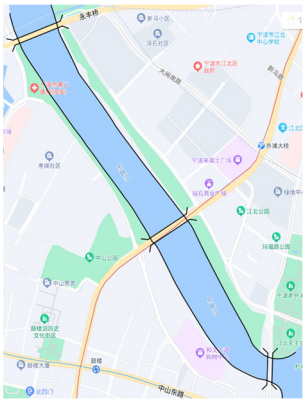


(c) Camera 3: MAE = 2.95 pixels, NMAE = 0.19%

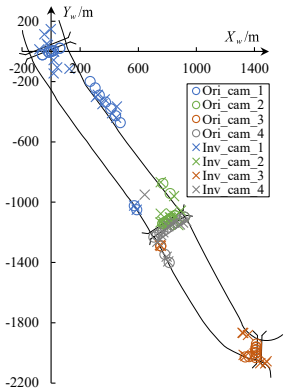


(d) Camera 4: MAE = 2.92 pixels, NMAE = 0.17%

Fig. 13 Projection comparison in 2D pixel coordinates



(a) Scaled sketch map



(b) Inversed projection in 3D world coordinates

Fig. 14 Inversed projection comparison in 3D world coordinates (XY plane), Ori: Original, Inv: Inverse

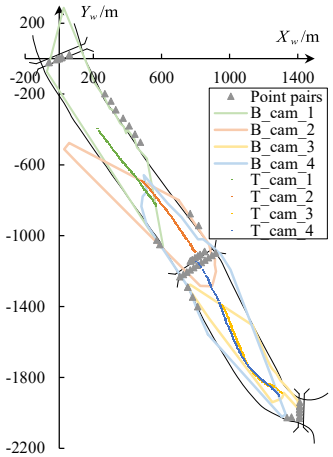
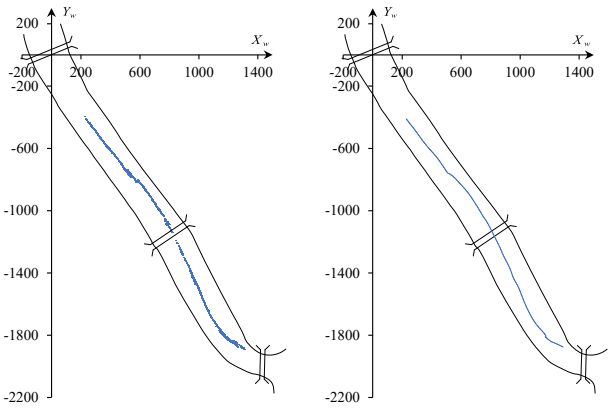


Fig. 15 Vessel's trajectory of individual cameras (video set No. 2), B: Boundary, T: Trajectory

inferred to be approximately 0 m ($Z_w = 0$) based on other points and is used for subsequent result presentation. The pixel coordinates of the water surface boundaries are extracted from a video frame and projected back to the world coordinate system to explicit the range each camera can observe in the world coordinates. The region enclosed by the light-colored lines in Fig. 15 represents the observation range of each camera in the world coordinate system, and the dark-colored points show the original tracking result of the target vessel from test video set No. 2. The boundaries of the observed waters almost coincide with the map, and the adjacent observation regions and the trajectories in individual cameras are both connected. Cameras 3 and 4 have a high degree of overlap in the observation area. Within this overlapping region, these two cameras captured the trajectory of the same vessel from different perspectives. The consistency between the yellow dots (Camera 3) and blue dots (Camera 2) in Fig. 15 provides strong evidence for the validity of this method.

Nevertheless, the boundary of the west bank designated in camera 2 deviates from the expected position due to the absence of control points (point pairs) in that region. This deviation can be rectified by supplementing point pairs on the west bank, preferably within the deviated zone. This situation highlights the significant influence of camera calibration errors on the results. It is recommended to select point pairs on both sides of the river banks or scattered around the surrounding environment as much as possible to balance the result of the external parameter calibration.



(a) Original trajectory points (b) Filtered trajectory

Fig. 16 Whole trajectories of the vessel (video set No. 2)

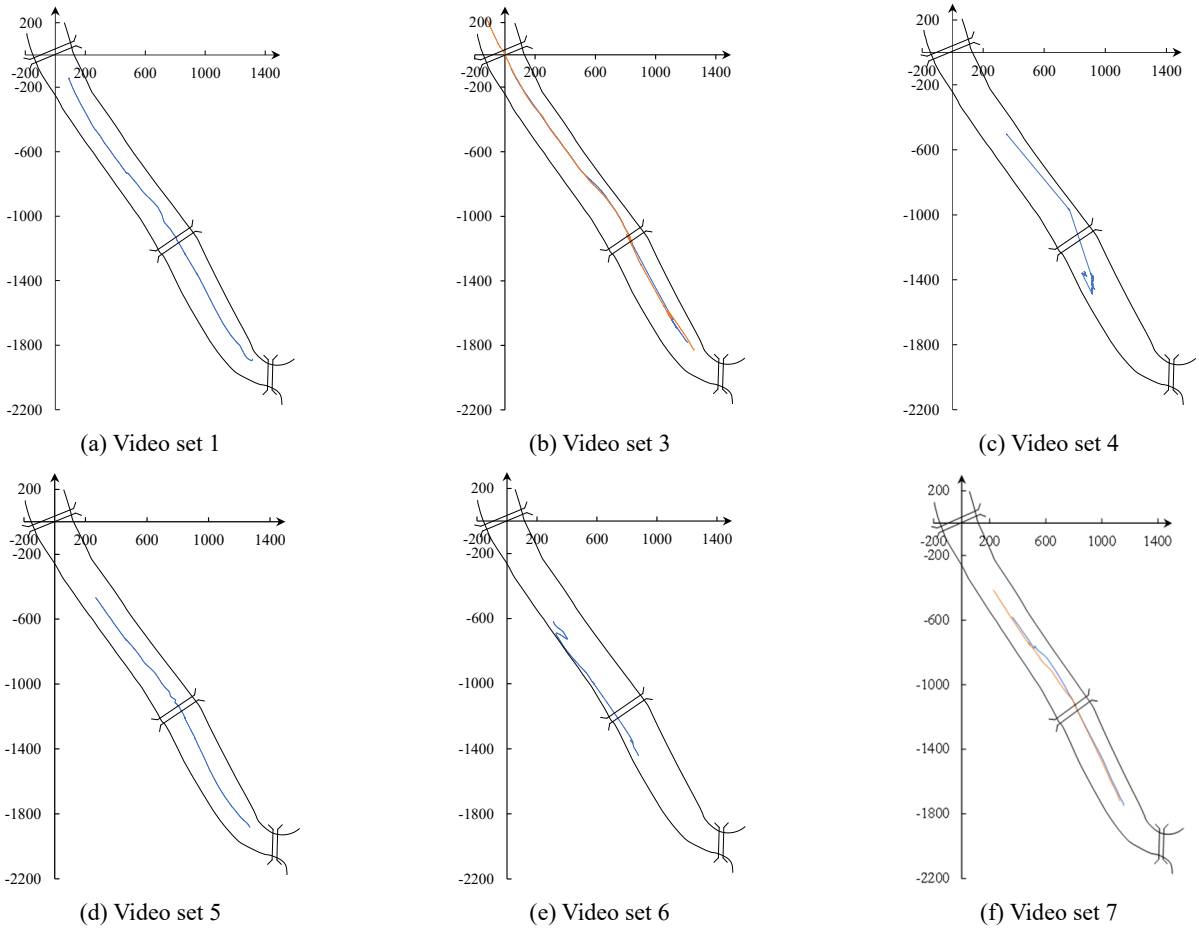
4.3 Results of the whole trajectory

To establish continuity among the vessel tracks recorded by individual cameras and mitigate the impact of varying shooting angles, the positions of identical vessel track points identified by different cameras at a given time were averaged after timestamp alignment. The results of this averaging process are displayed in Fig. 16(a). The results illustrate that the vessel's track covers the whole path, keeps within the river's boundaries, and is generally consistent with the videos. Savitzky–Golay filter (Savitzky and Golay 1964) is applied to the tracking results to smooth them without distorting their tendency. The final smoothed trajectory of test video set No. 2 and other video sets are shown in Figs. 16(b) and 17.

However, due to the difficulty in locating the ship at night, the vessel's trajectories are incomplete, as shown in Fig. 17(c). The vessel fails to be detected or continuously tracked, making it difficult to completely reconstruct its trajectory throughout the entire cruise. This can be attributed to the nighttime vessel features that blend in with the dark background and are difficult to detect. At the same time, the training set contains only a small number of low-quality nighttime frame images. As a result, the tracking will terminate, and the monitored vessel will receive a new number after a set number of intermediate frames miss the detecting object.

5. Conclusions

This paper proposes a relay tracking method using multiple monocular cameras, which enables the reconstruction of the vessel's whole trajectories in real-world coordinates. To address the challenge of detecting small objects for inland waterway vessels, various IoU loss



(a) Video set 1 (b) Video set 3 (c) Video set 4

(d) Video set 5 (e) Video set 6 (f) Video set 7

Fig. 17 Final vessel trajectories of other video sets

functions were compared, and the GIoU loss was ultimately selected due to its higher AP and AR scores. Furthermore, the multi-scale fusion attention mechanism further improves the one-stage model YOLOv5l6 in small object detection, including in occlusion cases and the night environment. To track the detected objects and ensure consistent individual IDs, the Bytetrack algorithm was employed, thereby avoiding the confusion of vessel trajectories in multiple-vessel cases. Field test results demonstrate the effectiveness of the proposed method, achieving favorable performance while maintaining frame rates above 25 fps during detection and tracking tasks on the test dataset.

The mapping of vessel trajectory from image to actual world coordinates is estimated by deducing the inverse projection relationship between pixel coordinates and world coordinates. Thus, the mapping relationship can be obtained based on the point pair information on surrounding facilities or structures, overcoming the limitations of field measurement and being flexible to the changing water surface elevation. According to the videos' context and the map, the results of vessel trajectory and water surface boundary in world coordinates are consistent with the actual observation. The consistency of the same vessel's trajectory in different cameras demonstrates the method's effectiveness. The complete vessel trajectory in the whole section of the river can be merged by aligning the time and well-fitted by the Savitzky–Golay filter.

Although this study presented promising results, some limitations require further investigation. First, the detection results in the night environment were not ideal, and the vessels could not be identified from some video perspectives. Second, although uncommon, the occlusion of vessels from each other can also lead to vessels being renumbered or swapped numbers, leading to an incomplete final trajectory.

Acknowledgments

The authors would like to acknowledge the committee of the 3rd International Competition for Structural Health Monitoring (IC-SHM 2022) for organization and data sharing. This research was funded by the National Natural Science Foundation of China (52127813) and the Fundamental Research Funds for the Central Universities (2242023K5006).

References

Bochkovskiy, A., Wang, C.-Y. and Liao, H.-Y.M. (2020), "YOLOv4: Optimal speed and accuracy of object detection", arXiv, 2004.10934. https://doi.org/10.48550/arXiv.2004.10934

Chabot, F., Chaouch, M., Rabarisoa, J., Teuliere, C. and Chateau, T. (2017), "Deep MANTA: A coarse-to-fine many-task network for joint 2D and 3D vehicle analysis from monocular image", *Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*), pp. 1827-1836. https://doi.org/10.1109/CVPR.2017.198

Clausse, A., Benslimane, S. and de La Fortelle, A. (2019), "Large-scale extraction of accurate vehicle trajectories for driving behavior learning", *Proceedings of the 30th IEEE Intelligent Vehicles Symposium* (*IV19*), pp. 2391-2396. https://doi.org/10.1109/IVS.2019.8814095

Ge, Z., Liu, S., Wang, F., Li, Z. and Sun, J. (2021), "YOLOX: Exceeding YOLO series in 2021", arXiv, 2107.08430. https://doi.org/10.48550/arXiv.2107.08430

Gevorgyan, Z. (2022), "SIoU loss: More powerful learning for bounding box regression".

Girshick, R. (2015), "Fast R-CNN", *Proceedings of 2015 IEEE International Conference on Computer Vision* (*ICCV*), pp. 1440-1448. https://doi.org/10.1109/ICCV.2015.169

Girshick, R., Donahue, J., Darrell, T. and Malik, J. (2014), "Rich feature hierarchies for accurate object detection and semantic segmentation", *Proceedings of 2014 IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*), pp. 580-587. https://doi.org/10.1109/CVPR.2014.81

He, K.M., Zhang, X.Y., Ren, S.Q. and Sun, J. (2014), "Spatial pyramid pooling in deep convolutional networks for visual recognition", *Proceedings of the 13th European Conference on Computer Vision* (*ECCV*), Zurich, Switzerland, September. https://doi.org/10.1007/978-3-319-10578-9_23

Kong, W. and Hu, T. (2019), "A deep neural network method for detection and tracking ship for unmanned surface vehicle", *Proceedings of 2019 IEEE 8th Data Driven Control and Learning Systems Conference* (*DDCLS*), May. https://doi.org/10.1109/DDCLS.2019.8908899

Lee, W.J., Roh, M.I., Lee, H.W., Ha, J., Cho, Y.M., Lee, S.J. and Son, N.S. (2021), "Detection and tracking for the awareness of surroundings of a ship based on deep learning", *J. Comput. Des. Eng.*, **8**(5), 1407-1430. https://doi.org/10.1093/jcde/qwab053

Li, S.L., Guo, Y.P., Xu, Y. and Li, Z.L. (2019), "Real-time geometry identification of moving ships by computer vision techniques in bridge area", *Smart. Struct. Syst.*, *Int. J.*, **23**(4), 359-371. https://doi.org/10.12989/sss.2019.23.4.359

Li, G., Lei, Y., Si, L. and Zheng, C. (2021), "Self-supervised visual representation learning for fine-grained ship detection", *Proceedings of 2021 IEEE 4th International Conference on Information Systems and Computer Aided Education* (*ICISCAE*), September. https://doi.org/10.1109/ICISCAE52414.2021.9590709

Liu, S., Qi, L., Qin, H.F., Shi, J.P. and Jia, J.Y. (2018), "Path aggregation network for instance segmentation", *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), pp. 8759-8768. https://doi.org/10.1109/CVPR.2018.00913

Liu, J., Shi, G. and Zhu, K. (2019), "Vessel trajectory prediction model based on AIS sensor data and adaptive chaos differential evolution support vector regression (ACDE-SVR)", *Appl. Sci.*, **9**(15), 2983. https://doi.org/10.3390/app9152983

Mahaur, B. and Mishra, K.K. (2023), "Small-object detection based on YOLOv5 in autonomous driving systems", *Pattern Recogn. Lett.*, **168**, 115-122. https://doi.org/10.1016/j.patrec.2023.03.009

Omrani, E., Mousazadeh, H., Omid, M., Masouleh, M.T., Jafarbiglu, H., Salmani-Zakaria, Y., Makhsoos, A., Monhaseri, F. and Kiape, A. (2020), "Dynamic and static object detection and tracking in an autonomous surface vehicle", *Ships Offshore Struct.*, **15**(7), 711-721. https://doi.org/10.1080/17445302.2019.1668642

Raj, J.A., Idicula, S.M. and Paul, B. (2022), "Lightweight SAR Ship detection and 16 Class Classification using Novel Deep Learning Algorithm with a Hybrid Preprocessing Technique", *Int. J. Remote Sens.*, **43**(15-16), 5820-5847. https://doi.org/10.1080/01431161.2021.2008544

Redmon, J. and Farhadi, A. (2017), "YOLO9000: Better, Faster, Stronger", *Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*), July. https://doi.org/10.1109/CVPR.2017.690

Redmon, J. and Farhadi, A. (2018), "YOLOv3: An incremental improvement", arXiv, 1804.02767. https://doi.org/10.48550/arXiv.1804.02767

Redmon, J., Divvala, S., Girshick, R. and Farhadi, A. (2016), "You Only Look Once: Unified, Real-Time Object Detection", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*). https://doi.org/10.1109/CVPR.2016.91

Ren, S.Q., He, K.M., Girshick, R. and Sun, J. (2017), "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", *IEEE Trans. Pattern Anal. Mach. Intell.*, **39**(6), 1137-1149. https://doi.org/10.1109/TPAMI.2016.2577031

Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I. and Savarese, S. (2019), "Generalized intersection over union: A metric and a loss for bounding box regression", *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), June. https://doi.org/10.1109/CVPR.2019.00075

Savitzky, A. and Golay, M.J.E. (1964), "Smoothing + differentiation of data by simplified least squares procedures", *Anal. Chem.*, **36**(8), 1627-1639. https://doi.org/10.1021/ac60214a047

Szpak, Z.L. and Tapamo, J.R. (2011), "Maritime surveillance: Tracking ships inside a dynamic background using a fast level-set", *Expert Syst. Appl.*, **38**(6), 6669-6680. https://doi.org/doi.org/10.1016/j.eswa.2010.11.068

Tang, W., Mondal, T.G., Wu, R.T., Subedi, A. and Jahanshahi, M.R. (2023), "Deep learning-based post-disaster building inspection with channel-wise attention and semi-supervised learning", *Smart. Struct. Syst., Int. J.*, **31**(4), 365-381. https://doi.org/10.12989/sss.2023.31.4.365

The 3rd International Competition for Structural Health Monitoring (IC-SHM 2022). https://shmc.tongji.edu.cn/ICSHM2022/main.htm

Thombre, S., Zhao, Z., Ramm-Schmidt, H., García, J.M.V., Malkamäki, T., Nikolskiy, S., Hammarberg, T., Nuortie, H., Bhuiyan, M.Z.H., Särkkä, S. and Lehtola, V.V. (2022), "Sensors and AI techniques for situational awareness in autonomous ships: a review", *IEEE Trans. Intell. Transp. Syst.*, **23**(1), 64-83. https://doi.org/10.1109/TITS.2020.3023957

Tong, Z., Chen, Y., Xu, Z. and Yu, R. (2023), "Wise-IoU: bounding box regression loss with dynamic focusing mechanism", arXiv preprint arXiv:2301.10051.

Ultralytics (2020), YOLOv5: Open source neural networks in Python, accessed 9 June 2020. https://github.com/ultralytics/yolov5/

Valsamis, A., Tserpes, K., Zissis, D., Anagnostopoulos, D. and Varvarigou, T. (2017), "Employing traditional machine learning algorithms for big data streams analysis: The case of object trajectory prediction", *J. Syst. Softw.*, **127**, 249-257. https://doi.org/10.1016/j.jss.2016.06.016

Yang, L., Zhang, R.-Y., Li, L. and Xie, X. (2021), "SimAM: A simple, parameter-free attention module for convolutional neural networks", *Proceedings of the 38th International Conference on Machine Learning*, **139**, 11863-11874.

Yoon, H., Shin, J. and Spencer Jr., B.F. (2018), "Structural displacement measurement using an unmanned aerial system", **33**(3), 183-192. https://doi.org/10.1111/mice.12338

Zhang, B. and Zhang, J. (2021), "A traffic surveillance system for obtaining comprehensive information of the passing vehicles based on instance segmentation", *IEEE Trans. Intell. Transp. Syst.*, **22**(11), 7040-7055. https://doi.org/10.1109/TITS.2020.3001154

Zhang, T., Jiang, L., Xiang, D., Ban, Y., Pei, L. and Xiong, H. (2019), "Ship detection from PolSAR imagery using the ambiguity removal polarimetric notch filter", *ISPRS J. Photogramm. Remote Sens.*, **157**, 41-58. https://doi.org/10.1016/j.isprsjprs.2019.08.009

Zhang, B., Xu, Z., Zhang, J. and Wu, G. (2022a), "A warning framework for avoiding vessel-bridge and vessel-vessel collisions based on generative adversarial and dual-task networks", *Comput.-Aided Civil Infrastr. Eng.*, **37**(5), 629-649. https://doi.org/doi.org/10.1111/mice.12757

Zhang, Y.F., Ren, W.Q., Zhang, Z., Jia, Z., Wang, L. and Tan, T.N. (2022b), "Focal and efficient IOU loss for accurate bounding box regression", *Neurocomputing*, **506**, 146-157. https://doi.org/10.1016/j.neucom.2022.07.042

Zheng, Z.H., Wang, P., Liu, W., Li, J.Z., Ye, R.G. and Ren, D.W. (2020), "Distance-IoU loss: faster and better learning for bounding box regression", *Proceedings of the AAAI Conference on Artificial Intelligence*, **34**(07), 12993-13000. https://doi.org/10.1609/aaai.v34i07.6999